Non-Parametric Measure Approximations for Constrained Multi-Objective Optimisation under Uncertainty

Mickael Rivier, Nassim Razaaly, Pietro Marco Congedo

Non-Parametric Measure Approximations for Constrained Multi-Objective Optimisation under Uncertainty

M. Rivier[a,b], N. Razaaly[a,c], P.M. Congedo[a]

[a]*Inria, Centre des Mathématiques Appliquées, Ecole Polytechnique, IPP, 91120 Palaiseau, France*
[b]*ArianeGroup, Le Haillan, 33185 Cedex, France*
[c]*Institut Pprime, ISAE-ENSMA, 2 Bd des Frères Lumière, 86360 Chasseneuil-du-Poitou, France*

Abstract

In this paper, we propose non-parametric estimations of robustness and reliability measures approximation error, employed in the context of constrained multi-objective optimisation under uncertainty. These approximations with tunable accuracy permit to capture the Pareto front in a parsimonious way, and can be exploited within an adaptive refinement strategy.

First, we illustrate an efficient approach for obtaining joint representations of the robustness and reliability measures, allowing sharper discrimination of Pareto-optimal designs. A specific surrogate model of these objectives and constraints is then proposed to accelerate the optimisation process.

Secondly, we propose an adaptive refinement strategy, using these tunable accuracy approximations to drive the computational effort towards the computation of the optimal area. To this extent, an adapted Pareto dominance rule and Pareto optimal probability computation are formulated.

The performance of the proposed strategy is assessed on several analytical test-cases against classical approaches. We also illustrate the method on an engineering application, performing shape optimisation under uncertainty of an Organic Rankine Cycle turbine.

Keywords:
Optimisation under uncertainty, Gaussian processes, Robustness and reliability, Organic Rankine cycle

1. Introduction

There has been a growing interest in optimisation with uncertainty in recent years [1, 2, 3, 4]. We focus here specifically on cost-efficient and derivative-free strategies for tackling multiple objectives and constraints in this context. Optimisation Under Uncertainty (OUU) can be roughly classified under two categories, Reliability-Based Design Optimisation (RBDO), which deals with probabilistic and worst-case feasibility constraints, and Robust Design Optimisation (RDO), where the deterministic objectives are replaced with averaged or worst-case ones, possibly in a multi-objective context (*i.e.* Taguchi optimisation).

In the context of multi-objective optimisation [5, 6], a popular derivative-free optimisation strategy for constrained multi-objective problems are Evolutionary Algorithms (EA). A comparison of multi-objective EA strategies is proposed in [7], in the bi- or tri-objective setting. When dealing with more objectives (≥ 4), one should refer to the many-objective optimisation literature, notably reviewed in [8]. For more details, [9] gives a thorough description of both the theoretical foundations and practical metaheuristics for application to engineering test cases.

In this work, we focus specifically on performing optimisation in the presence of uncertainties. Such a setting has received significant attention from the optimisation community, as can be seen in several reviews [10, 11, 12]. Notably, [10] proposed four types of uncertainties: (i) noise, which is an intrinsic bias on the evaluation of the fitness function, coming, for example, from imperfect sensors; (ii) robustness, which refers to specific parameters that may vary from their nominal value between the optimisation procedure and the real-world application; (iii) fitness approximation, when the true fitness value is not computable or very expensive, and less accurate but cheaper representation is used instead; (iv) time-varying fitness function, meaning that the optimum changes through time and that the optimisation algorithm should tackle this variability.

Much work has been devoted to tackling noise within the optimisation process. This field is usually referred to as Noisy Optimisation (NO). In this context, some parsimonious techniques rely on the assumption that there exist either different levels of accuracy for computing the Quantities of Interest. Namely, multi-level techniques [13, 14] usually involve several mesh refinements for numerical simulation, and multi-fidelity approaches [15, 16] use different modeling strategies. Recent works have also proposed adaptive approaches for tackling problems where accuracy can be tuned online. This approach is particularly adapted when the fitness function is the limit of intermediate approximations of increasing fidelities, such as iterative solvers or Monte Carlo estimations. This concept has been proposed in [17] with online computation time allocation in a Bayesian Optimisation (BO) context. In this work, a similar online refinement strategy is employed to achieve high parsimony.

In a multi-objective setting, the presence of noise and cohabitation of different approximation accuracies raises the need for extending the classical Pareto dominance to an uncertain environment. The EA community has proposed techniques for problems featuring uniform [18] or Gaussian noises [19, 20, 21]. Interval and worst-case approaches are tackled in [22, 23, 24], within the classical NSGA-II and MOEA frameworks. Recent developments [25, 26, 27, 28] extends these computations to non-parametric noise distributions through the construction of histogram-based approximate densities, which permits fast computations of the probability of dominance. A straightforward extension of the Pareto dominance rule to interval uncertainty in the presence of constraints is proposed in [29] under the name of Bounding-Boxes. This extension has been exploited in several papers [30, 31] within an adaptive strategy for performing multi-objective optimisation on objective functions computed with tunable accuracy.

In this paper, the core idea of the approach is to tune the accuracy of statistical objectives and constraints approximations to focus computational power on the most promising

designs. The first contribution is the computation of non-parametric representation of the robustness and reliability measures error, permitting to estimate the Pareto front in a parsimonious way. Specifically, we aim to avoid the restriction to independent uniform [31] or Gaussian [32, 33, 34, 35] distributions, which often rely on the so-called projected process technique. We propose to use empirical sampling-based distribution approximations, that should allow for more representative error estimations given the current information, and hopefully quicker convergence. An original approach is introduced for constructing these measure samples, which relies on a coupled space Gaussian Process (GP) surrogate model for drawing joint realisations of objectives and constraints over the whole design space. These approximated measures are treated as a random vector and are compared and ranked using the concept of probabilistic Pareto dominance [36, 26, 27]. The second main contribution of this work is an adaptive refinement strategy, where the accuracy is improved only on the most promising individuals. We assess the proposed approach on several (mono and bi-objective) optimisation problems to estimate its computational cost and accuracy in terms of average distance to the exact Pareto front. Finally, we tackle the problem of an Organic Rankine turbine shape optimisation with random operating conditions.

Section 2 illustrates the formulation of multi-objective optimisation problems under uncertainties treated here within a probabilistic setting. In Section 3, we introduce the numerical ingredients permitting to compute non-parametric measure approximations. Then, Section 4 illustrates the proposed general framework to solve the OUU problem. In Section 5, we analyse the performance of the proposed approach on some algebraic test cases by comparing it with known techniques in the literature. Finally, Section 6 is devoted to showing the applicability of the framework to complex engineering test cases with the shape optimisation of Organic Rankine cycles turbines.

2. Uncertainty-based optimisation problem

In this section, the context of this work is made explicit. The general formulation of the tackled uncertainty-based optimisation problem is given, and classical choices in terms of robustness and reliability measures are introduced. The proposed probabilistic setting is then given, which allows for comparing aleatory values in a constrained multi-objective context.

2.1. Optimisation

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

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

with objectives $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^{m_1}$ and constraints $\boldsymbol{g}(\boldsymbol{x}) \in \mathbb{R}^{m_2}$, and where the design vector $\boldsymbol{x}$ lives in $\mathcal{X} \subset \mathbb{R}^n$. Note that $\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}$ stands for $\forall i, g_i(\boldsymbol{x}) \leq 0$.

The computation of $\boldsymbol{f}$ and $\boldsymbol{g}$ can also depend on a multitude of uncontrollable parameters (*e.g.* environmental, material, geometrical, ...), here denoted by $\boldsymbol{\xi} \in \Xi$.

The most straightforward setting is to consider these uncertain parameters as aleatoric. This work is limited to this scope and does not deal with epistemic uncertainties. In this context, $\boldsymbol{\xi}$, $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ are random variables and Equation (1) is not suitable anymore.

It is classical to reformulate this problem into a measure-based Optimisation Under Uncertainty (OUU) problem, where specific statistics of the random outputs $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ are plugged as objective and constraint functions.

2.2. Robustness and reliability measures

Writing $\boldsymbol{\rho_f}$ and $\boldsymbol{\rho_g}$ some statistical measures on the objective function $\boldsymbol{f}$ and constraint $\boldsymbol{g}$, the OUU problem can be formulated as follows:

$$\begin{aligned} \text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) \leq \boldsymbol{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}. \end{aligned} \tag{2}$$

As raised in the introduction, using statistical measures as objectives is usually referred to as Robust Design Optimisation (RDO), and these statistics ($\boldsymbol{\rho_f}$) are called robustness measures. Similarly, Reliability-Based Design Optimisation (RBDO) deals with so-called reliability measures $\boldsymbol{\rho_g}$ as constraints.

In the following, the original functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are gathered in a vector $\boldsymbol{q} \in \mathbb{R}^{m_1+m_2}$, that refers to Quantities of Interest (QoI). Namely,

$$\forall (\boldsymbol{x}, \boldsymbol{\xi}) \in \mathcal{X} \times \Xi, \ \boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi}) = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}) \\ \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}) \end{pmatrix}. \tag{3}$$

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

$$\forall \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{\rho}(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{\rho_f}(\boldsymbol{x}) \\ \boldsymbol{\rho_g}(\boldsymbol{x}) \end{pmatrix}. \tag{4}$$

The most classical robustness and reliability measures for problem (2) are the following:

- Expectation: $\rho(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[q(\boldsymbol{x}, \boldsymbol{\xi})]$,
- Variance: $\rho(\boldsymbol{x}) = \mathbb{V}_{\boldsymbol{\xi}}[q(\boldsymbol{x}, \boldsymbol{\xi})]$,
- Worst case: $\rho(\boldsymbol{x}) = \max_{\boldsymbol{\xi}}[q(\boldsymbol{x}, \boldsymbol{\xi})]$
- Quantile associated with probability p: $\rho(\boldsymbol{x}) = \mathrm{q}^p_{\boldsymbol{\xi}}[q(\boldsymbol{x}, \boldsymbol{\xi})]$ (also called Value at Risk, VaR) ,
- Superquantile: $\rho(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}\big[q(\boldsymbol{x}, \boldsymbol{\xi}) \mid q(\boldsymbol{x}, \boldsymbol{\xi}) \geq \mathrm{q}^p_{\boldsymbol{\xi}}[q(\boldsymbol{x}, \boldsymbol{\xi})]\big]$ (also called Expected Value at Risk, EVaR).

Note that any combination of these may be used, and one has to choose the most relevant for a given application.

In practice, robust objectives usually are either expectation or worst-case measures. The latest can also be relaxed to quantile or superquantile measures.

As for reliability measures, they can impose hard constraint, by using worst-case measures, but are usually relaxed into probabilistic constraints, that ensures $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}) \leq \mathbf{0}$ with a given probability p. This can be readily obtained by using the p-quantile as a reliability measure.

Given the distribution of the random input variable $\boldsymbol{\xi}$, and specific measures $\boldsymbol{\rho}$, problem (2) can be solved with classical techniques. However, if the unitary evaluation of $\boldsymbol{q}$ is expensive, $\boldsymbol{\rho}$ becomes very tedious to compute. A compromise between the accuracy and computational cost of $\boldsymbol{\rho}$ must be chosen.

Specifically, we propose to consider these estimated measures as random variables, with a given Probability Density Function (PDF) representing the associated inaccuracy. Next section presents this probabilistic setting in details.

2.3. Probabilistic setting

In this work, we aim to tackle multi-objective optimisation problems (≤ 3 objectives) where the statistical measures $\boldsymbol{\rho}$ used as objectives and constraint functions are not exactly known and regarded as random fields $\boldsymbol{P}$ over $\mathcal{X}$. In this context, the classical Pareto dominance does not hold and a probabilistic one must be defined. Several propositions exist in the literature [36, 26, 27], notably differing by the assumed distribution shape.

The classical Pareto dominance is described in the case of minimisation. For two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^m$,

$$\begin{aligned}
\boldsymbol{a} \succ \boldsymbol{b} \ (\boldsymbol{a} \text{ dominates } \boldsymbol{b}) &\iff \forall j \in [\![1, m]\!],\ a_j \leq b_j \quad \text{and} \\
&\qquad\ \exists j \in [\![1, m]\!],\ a_j < b_j, \\
\boldsymbol{a} \succcurlyeq \boldsymbol{b} \ (\boldsymbol{a} \text{ strictly dominates } \boldsymbol{b}) &\iff \forall j \in [\![1, m]\!],\ a_j < b_j, \\
\boldsymbol{a} \sim \boldsymbol{b} \ (\boldsymbol{a} \text{ is indifferent to } \boldsymbol{b}) &\iff \boldsymbol{a} \not\succ \boldsymbol{b} \text{ and } \boldsymbol{b} \not\succ \boldsymbol{a}.
\end{aligned} \tag{5}$$

In the presence of constraints, we follow the constrained Pareto dominance proposed in [31]. For two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^m$, with $\boldsymbol{a_f}$ and $\boldsymbol{b_f}$ the objective values and $\boldsymbol{a_g}$ and $\boldsymbol{b_g}$ the constraint values,

$$\begin{aligned}
\boldsymbol{a} \succ_c \boldsymbol{b} &\iff \boldsymbol{a} \in \mathcal{A} \ \text{ and } \ \boldsymbol{a_f} \succ \boldsymbol{b_f} \quad \text{or} \quad \boldsymbol{b} \in \mathcal{F}, \\
\boldsymbol{a} \succcurlyeq_c \boldsymbol{b} &\iff \boldsymbol{a} \in \mathcal{A} \ \text{ and } \ \boldsymbol{a_f} \succcurlyeq \boldsymbol{b_f} \quad \text{or} \quad \boldsymbol{b} \in \mathcal{F}, \\
\boldsymbol{a} \sim_c \boldsymbol{b} &\iff \boldsymbol{a} \not\succ_c \boldsymbol{b} \ \text{ and } \ \boldsymbol{b} \not\succ_c \boldsymbol{a},
\end{aligned} \tag{6}$$

where $\mathcal{A}$ is the admissible set, where all constraints are satisfied, and $\mathcal{F}$ is the failure set, where at least one constraint is not verified. Using this definition, two individuals that both violate the constraints will mutually dominate each other. This allows to discard inefficient individuals even when no satisfying individual has yet been found.

Here, a probabilistic Pareto dominance between two designs $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ hence consists in computing the probability $\mathbb{P}_{\boldsymbol{P}}\big[\boldsymbol{P}(\boldsymbol{x}_1) \succ_c \boldsymbol{P}(\boldsymbol{x}_2)\big]$ of a design to dominate the other. Recall that $\boldsymbol{P}$ is a random field, meaning that $\boldsymbol{P}(\boldsymbol{x}_1)$ and $\boldsymbol{P}(\boldsymbol{x}_2)$ are random variables. For better generality and readability, these are denoted $\mathbf{A}$ and $\mathbf{B}$ in the following.

Formally, the unconstrained probability is defined as:

$$\mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ \mathbf{B}\big] = \int_{\mathbb{R}^m_-} \phi_{\mathbf{A}-\mathbf{B}}(\boldsymbol{y})\mathrm{d}\boldsymbol{y} = \Phi_{\mathbf{A}-\mathbf{B}}(\mathbf{0}), \tag{7}$$

where $\phi_{\mathbf{A}-\mathbf{B}}$ is the Probability Density Function (PDF) of $\mathbf{A}-\mathbf{B}$ and $\Phi_{\mathbf{A}-\mathbf{B}}$ is the associated Cumulative Density Function (CDF). An equivalent formula can be written using the joint PDF $\phi_{\mathbf{A},\mathbf{B}}$ of $\mathbf{A}$ and $\mathbf{B}$:

$$\mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ \mathbf{B}\big] = \iint_{\boldsymbol{a}\succ\boldsymbol{b}} \phi_{\mathbf{A},\mathbf{B}}(\boldsymbol{a},\boldsymbol{b})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{b}. \tag{8}$$

As for the constrained case, definition (6) shows that for $\boldsymbol{a}$ to dominate $\boldsymbol{b}$, either $\boldsymbol{b}$ must be in the failure domain, or $\boldsymbol{a}$ must be in the admissible domain and dominate $\boldsymbol{b}$ according to the classical Pareto dominance rule. The integral in (8) can be split up depending on $\boldsymbol{b}$ belonging to the failure or admissible domain, and an additional dominance constraint can be added in the latest case. Hence, the domination probability reads:

$$\mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ_c \mathbf{B}\big] = \int_{\boldsymbol{b}\in\mathcal{F}} \phi_{\mathbf{B}}(\boldsymbol{b})\mathrm{d}\boldsymbol{b} + \iint_{\substack{\boldsymbol{a}\in\mathcal{A}\\ \boldsymbol{b}\in\mathcal{A}\\ \boldsymbol{a}\succ\boldsymbol{b}}} \phi_{\mathbf{A},\mathbf{B}}(\boldsymbol{a},\boldsymbol{b})\mathrm{d}\boldsymbol{a}\mathrm{d}\boldsymbol{b}. \tag{9}$$

Note that this formulation naturally tackles the existence of dependency between the random variables $\mathbf{A}$ and $\mathbf{B}$, which greatly differs from the assumptions made in [26, 27].

We finally propose a *probabilistic constrained Pareto dominance* rule between two random vectors:

$$\begin{aligned}
\mathbf{A} \underset{0}{\succ}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ_c \mathbf{B}\big] = 1,\\
\mathbf{A} \underset{0}{\succcurlyeq}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succcurlyeq_c \mathbf{B}\big] = 1,\\
\mathbf{A} \underset{0}{\sim}_c \mathbf{B} &\iff \mathbf{A} \underset{0}{\not\succ}_c \mathbf{B} \ \text{ and } \ \mathbf{B} \underset{0}{\not\succ}_c \mathbf{A}.
\end{aligned} \tag{10}$$

This dominance rule generalises the *constrained boxed Pareto dominance* introduced in [31].

However, in the context of random variables with infinite support (*e.g.* Gaussian variable), the dominance probability cannot reach exactly 1. It has been proposed in [26] to relax this threshold. The *ϵ-relaxed probabilistic constrained Pareto dominance* finally reads:

$$\begin{aligned}
\text{Domination: } \mathbf{A} \underset{\epsilon}{\succ}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succ_c \mathbf{B}\big] \geq 1-\epsilon,\\
\text{Strict domination: } \mathbf{A} \underset{\epsilon}{\succcurlyeq}_c \mathbf{B} &\iff \mathbb{P}_{\mathbf{A},\mathbf{B}}\big[\mathbf{A} \succcurlyeq_c \mathbf{B}\big] \geq 1-\epsilon,\\
\text{Indifference: } \mathbf{A} \underset{\epsilon}{\sim}_c \mathbf{B} &\iff \mathbf{A} \underset{\epsilon}{\not\succ}_c \mathbf{B} \ \text{ and } \ \mathbf{B} \underset{\epsilon}{\not\succ}_c \mathbf{A}.
\end{aligned} \tag{11}$$

The above formulation gives a general setting for comparing aleatory vectors, regardless of their distribution shape. In practice, ϵ is usually taken at 10^{-2}. Next section proposes a sampling-based technique to get an accurate representation of the random field $\boldsymbol{P}$ over the design space $\mathcal{X}$.

3. Non-parametric measure approximation

We aim to make as few assumptions as possible on the distribution of $\boldsymbol{P}(\boldsymbol{x})$, for $\boldsymbol{x}$ in $\mathcal{X}$. To this extent, we only assume that realisations $\boldsymbol{P}^{(i)}(\boldsymbol{x})$ can be drawn and exploit this empirical set of samples to characterise $\boldsymbol{P}(\boldsymbol{x})$.

More precisely, because $\boldsymbol{P}$ is a random field over the design space $\mathcal{X}$, we propose to draw realisations $\boldsymbol{P}^{(i)}$ of the whole random field. This consists in jointly drawing $\boldsymbol{P}^{(i)}(\boldsymbol{x})$ on a set of locations $\boldsymbol{x}$.

In this setting, dominance must only be verified between these joint samples. With a number $N_{samples}$ of realisations, the probability of dominance between two designs can be empirically estimated as follows:

$$\mathbb{P}_{\boldsymbol{P}}\big[\boldsymbol{P}(\boldsymbol{x}_j) \not\succ_c \boldsymbol{P}(\boldsymbol{x}_k)\big] \simeq \frac{1}{N_{samples}} \sum_{i=1}^{N_{samples}} \mathbb{1}_{\{\boldsymbol{P}^{(i)}(\boldsymbol{x}_j)\not\succ_c \boldsymbol{P}^{(i)}(\boldsymbol{x}_k)\}}. \tag{12}$$

In practice, we tend to take $N_{samples}$ between 10^3 and 10^4.

The major task is to obtain realisations $\boldsymbol{P}^{(i)}(\boldsymbol{x})$ of the robustness and reliability measures. To this extent, we assume that a probabilistic surrogate can be constructed on the original Quantities of Interest $\boldsymbol{q}$ and draw realisations $\boldsymbol{P}^{(i)}$ through measure estimations on functional realisations $\boldsymbol{q}^{(i)}(\boldsymbol{x}, \cdot)$, simultaneously for every studied location $\boldsymbol{x} \in \mathcal{X}$. In this work, we use Gaussian process surrogate models, that are built using the GPy python package [37].

We first recall the main characteristics and formulas of Gaussian Process surrogate models. Then, an approximated approach for drawing these realisations of $\boldsymbol{P}$ from a GP surrogate of $\boldsymbol{q}$ in the coupled space $\mathcal{X} \times \Xi$ is proposed.

3.1. Gaussian processes

This section gives a quick reminder of the GP setting, its interpretability and the main formulas that will be used in the following. Gaussian process surrogate modelling, also called Kriging, is a Bayesian kernel-based regression method [38].

The kernel setting, with the famous so-called kernel trick, allows to transpose most of the classical parametric (*e.g.* linear) approaches to the non-parametric world. In practice, basing on data points location $\boldsymbol{x}_i$, parametric features can be incorporated by replacing scalar product between the different parameters $\boldsymbol{x}_1^\intercal \boldsymbol{x}_2$ by a scalar product between chosen features $\boldsymbol{\phi}(\boldsymbol{x}_1)^\intercal \boldsymbol{\phi}(\boldsymbol{x}_2)$. Then, the kernel trick consists in choosing a kernel function k, and to use $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ in place of the previous scalar product. In this way, the features are not explicitly defined and can be of infinite number. Some kernels are called universal and engender a space of functions that is dense in the space of continuous function.

The Bayesian setting, in addition, tracks the distribution of all parameters (of infinite number with the kernel trick), and returns not only a predictive model of the Quantity of Interest, but its whole distribution at each location $\boldsymbol{x}$. In the context of Gaussian process, this distribution is fully determined by a predictive mean and a predictive variance for the QoI.

In practice, Gaussian processes can also simply be viewed as an infinite dimensional Gaussian distribution, which reduction to a finite set of points is also Gaussian. In this setting, the kernel plays the role of covariance function between points and the predictive mean and variance are the conditional mean and variances at each point $\boldsymbol{x}$. Given the training data $\{\boldsymbol{x}_{train}, \boldsymbol{y}_{train}\}$ where $\boldsymbol{y}_{train} = f(\boldsymbol{x}_{train}) + \epsilon$ and ϵ is a random Gaussian noise, they read:

$$\begin{aligned} \boldsymbol{\mu}_{GP}(\boldsymbol{x}) &= \boldsymbol{k}_*^\intercal (K + \Delta)^{-1} \boldsymbol{y}_{train}, \\ \Sigma_{GP}(\boldsymbol{x}) &= \boldsymbol{k}_{**} - \boldsymbol{k}_*^\intercal (K + \Delta)^{-1} \boldsymbol{k}_*, \end{aligned} \tag{13}$$

where K is the covariance matrix between training points $\boldsymbol{x}_{train}$, $K = k(\boldsymbol{x}_{train}, \boldsymbol{x}_{train})$, $\boldsymbol{k}_*$ is the covariance $k(\boldsymbol{x}_{train}, \boldsymbol{x})$ between the new points $\boldsymbol{x}$ and the training points and $\boldsymbol{k}_{**}$ is the covariance between the new points, $k(\boldsymbol{x}, \boldsymbol{x})$. Matrix Δ refers to the so-called input noise or nugget effect. Usually, it regularises the problem with the same noise on each training point, $\Delta = \sigma^2 I_N$, with N the number of training data. However, in the heteroscedastic case, one can readily assign a different noise to each training point, meaning $\Delta = diag\big(\{\sigma_i^2\}_i\big)$.

In this paper, hyperparameters of the kernel function are optimised by Maximum Likelihood. Because Automatic Relevance Determination (ARD) is exploited, a lengthscale parameter is assigned to each dimension. This heavily limits the use of these models when the problem features more than tens of dimensions.

Finally, one can draw functional realisations of a GP surrogate model, discretised on a finite set of points. A GP is a Gaussian random vector, so a realisation $\boldsymbol{f}^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}_{GP}, \Sigma_{GP})$ is drawn as follows:

$$\boldsymbol{f}^{(i)} = \boldsymbol{\mu}_{GP} + L_{GP} Y^{(i)}, \quad Y \sim \mathcal{N}(\boldsymbol{0}, I_N), \tag{14}$$

with L_{GP} referring to the Cholesky decomposition of the covariance matrix, so that $\Sigma_{GP} = L_{GP} L_{GP}^\intercal$.

3.2. Direct measure sampling in coupled space

We assume that a GP surrogate model can be built on $\boldsymbol{q}$ in the coupled design/uncertainty space. We here simply build a scalar GP for each output dimension. Based on training data $\big\{\boldsymbol{q}(\boldsymbol{x}_i, \boldsymbol{\xi}_i)\big\}_i$, the surrogate $\hat{\boldsymbol{q}}$ is constructed and yields an estimation of $\boldsymbol{q}$ at any location $(\boldsymbol{x}, \boldsymbol{\xi})$ in the coupled space $\mathcal{X} \times \Xi$.

This surrogate model can be exploited to obtain an estimation of the robustness and reliability measure through Monte Carlo integration. This integration is directly performed on the predictive mean $\hat{\boldsymbol{q}}$ of the GP. For example, with the expectation measure $\boldsymbol{\rho}(\boldsymbol{x}) =$

$\mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})]$, one can use the estimator :

$$\widetilde{\boldsymbol{\rho}}(\boldsymbol{x}) = \frac{1}{N_\xi} \sum_{j=1}^{N_\xi} \hat{\boldsymbol{q}}(\boldsymbol{x}, \boldsymbol{\xi}_j). \tag{15}$$

More interestingly, one can draw realisations of the GP built on $\boldsymbol{q}$, which directly translates into realisations of $\boldsymbol{P}$ by integrating over the uncertain dimension.

For example, in the case of the expectation measure $\boldsymbol{\rho}(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})]$, by evaluating $\boldsymbol{q}$ with N_ξ locations in the uncertain space at N_x different designs, the joint realisations of $\boldsymbol{P}$ over all these designs write as follows:

$$\forall k \in [\![1, N_x]\!], \quad \boldsymbol{P}^{(i)}(\boldsymbol{x}_k) = \frac{1}{N_\xi} \sum_{j=1}^{N_\xi} \boldsymbol{q}^{(i)}(\boldsymbol{x}_k, \boldsymbol{\xi}_j), \tag{16}$$

drawn simultaneously on a realisation $\boldsymbol{q}^{(i)}$ at all designs.

The same can be carried out with empirical estimators of other robustness and reliability measures.

However, one can notice that $\boldsymbol{q}^{(i)}$ is drawn at $N_x \times N_\xi$ locations, which can quickly reach a value of more than 10^5. Looking back at Equation (14), it reveals that Cholesky decomposition should hence be performed on a matrix of size $10^5 \times 10^5$. The computational cost associated with such a task is prohibitive. This problem is quite common in the literature, and several approximations techniques have been proposed, such as a circulant embedding of the covariance matrix [39], regularisation [40] or the use of a Gibbs sampler for Gaussian Markov Random Field [41]. However, some of these approaches require equispaced points, and the others remain too expensive to allow substantial sampling of the $\boldsymbol{q}$ random field.

As raised in [42], the complexity is particularly high when the domain is much larger than the characteristic correlation length of the random field. Intuitively, when a value is drawn, it gives information in its surrounding at a distance proportional to the correlation length.

Based on this idea of local information, an inducing point strategy is proposed in the next section for drawing realisations $\boldsymbol{q}^{(i)}$ at an extremely high number of locations simultaneously. This strategy is presented in Section 3.2.1, and a quantitative quality indicator is proposed in Section 3.2.2 for use in an iterative strategy in Section 3.2.3.

3.2.1. Inducing points strategy

The central idea is to draw realisations $\boldsymbol{q}^{(i)}$ of the Gaussian random field on few well-spread points, from which a smooth functional realisation can be extended to the whole input space. These well-chosen points will be referred to as inducing points.

In the following, for conciseness, coordinates $(\boldsymbol{x}, \boldsymbol{\xi})$ in the coupled space are denoted with $\boldsymbol{z}$. With a set of training designs $\boldsymbol{z}_{train} = (\boldsymbol{x}_{train}, \boldsymbol{\xi}_{train})$ and a set of inducing points $\boldsymbol{z}_{ind} = (\boldsymbol{x}_{ind}, \boldsymbol{\xi}_{ind})$, we denote $\boldsymbol{z}_{tot} = \boldsymbol{z}_{train} \cup \boldsymbol{z}_{ind}$.

We assume that a GP has been constructed on $\boldsymbol{z}_{train}$, as plotted in Figure 1, on values $\boldsymbol{q}_{train}$ and with an optimised kernel k_{train}. Its predictive mean and covariance are written

$\boldsymbol{\mu}_{train}$ and Σ_{train}. This surrogate model is the current estimation of $\boldsymbol{q}$ over the whole coupled space.

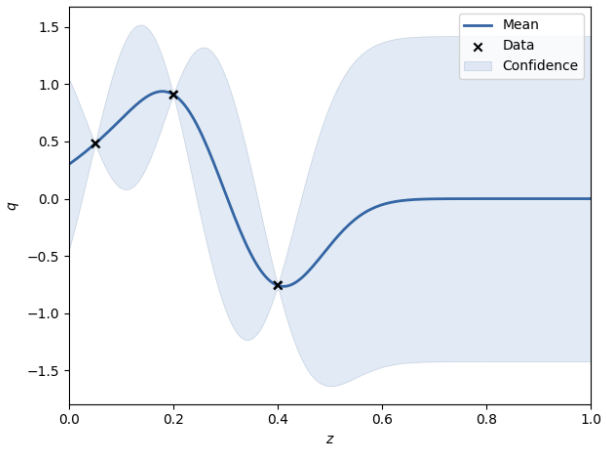


Figure 1: Gaussian process model built on the training data.

We aim to generate random realisations of this Gaussian field on an extensive set of points $\boldsymbol{z}_{draw}$. Direct Gaussian sampling would perform Cholesky decomposition on the matrix $K_{draw} = \Sigma_{train}(\boldsymbol{z}_{draw})$ of size N^2_{draw}, which quickly becomes unfeasible.

We propose to only draw realisations on the inducing points:

$$\boldsymbol{q}^{(i)}_{ind} = \boldsymbol{\mu}_{train}(\boldsymbol{z}_{ind}) + L_{ind}\boldsymbol{Y}^{(i)}, \quad \boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{0}, I_{N_{ind}}), \tag{17}$$

with L_{ind} the Cholesky decomposition of matrix $K_{ind} = \Sigma_{train}(\boldsymbol{z}_{ind})$ of much smaller size N^2_{ind}. Figure 2 gives an example of such drawn values at the inducing points.

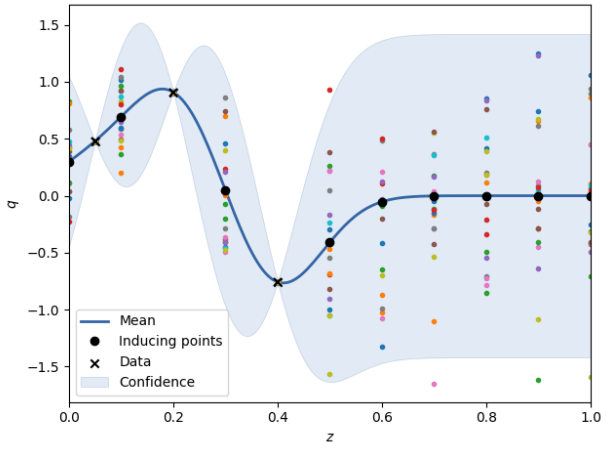


Figure 2: Realisations drawn at the inducing points.

Similarly to the locations $\boldsymbol{z}_{tot}$, we gather training values $\boldsymbol{q}_{train}$ and drawn ones $\boldsymbol{q}^{(i)}_{ind}$ in $\boldsymbol{q}^{(i)}_{tot}$. The realisations drawn at the inducing points are then extended to all locations $\boldsymbol{z}_{draw}$

by constructing a new GP on the $(\boldsymbol{z}_{tot}, \boldsymbol{q}_{tot}^{(i)})$ database. Note that the new GP kernel is not optimised as k_{train} is directly used. Final realisations estimates $\boldsymbol{q}_{draw}^{(i)}$ are obtained as follows:

$$\boldsymbol{q}_{draw}^{(i)} = \boldsymbol{\mu}_{tot}^{(i)}(\boldsymbol{z}_{draw}) = \boldsymbol{k}_*^\intercal K_{tot}^{-1} \boldsymbol{q}_{tot}^{(i)}, \tag{18}$$

where $\boldsymbol{k}_* = k_{train}(\boldsymbol{z}_{tot}, \boldsymbol{z}_{draw})$ and K_{tot}^{-1} only need to be computed once as they do not depend on i. These realisations are represented in Figure 3, and the empirical 2-σ band of the random field is computed on 10^5 samples. One can note that they fit well the real confidence band of the underlying GP.

Intuitively, this approach corresponds to parametrising the random field as a sum of kernels located at all training and inducing points. This is noticeable in the above Equation (18). By denoting $K_{tot}^{-1}\boldsymbol{q}_{tot}^{(i)}$ as a vector $\boldsymbol{\beta}^{(i)}$ of coefficients, the proposed realisations $\boldsymbol{q}_{draw}^{(i)}$ are linear combinations of the kernel function k_{train} located at each training and inducing point. Coefficients $\boldsymbol{\beta}^{(i)}$ manage the randomness and kernel functions the spatial characteristic. The functional realisations write $k_{train}(\boldsymbol{z}_{tot}, \cdot)^\intercal \boldsymbol{\beta}^{(i)}$.

In practice, the proposed approach has a lot of similarities with using a Karhunen-Loève Expansion (KLE) of a Gaussian random field with Nyström approximation of the basis functions. Computational details are given in Appendix A.

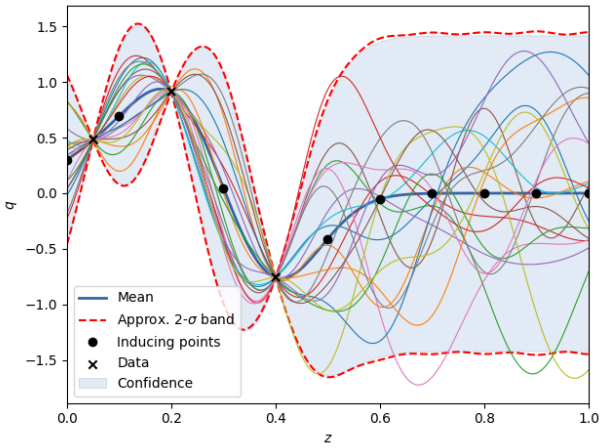


Figure 3: Functional realisations from draws in Figure 2. Approximated 2-σ band of the stochastic process in dashed red.

3.2.2. Quality indicator

To attain a good accuracy for this random field approximation, a key element is the number of inducing points. We propose in this section an indicator to assess the approximation quality. Intuitively, if the density of inducing points is high enough with respect to the lengthscales, the predictive means $\boldsymbol{q}_{draw}^{(i)}$ (Eq. (18)) will be able to represent all scales of the random process. In such cases, the associated predictive variance remains very low over the whole design space $\mathcal{X}$. The GP covariance reads:

$$\Sigma_{tot}(\boldsymbol{z}_{draw}) = K_{**} - \boldsymbol{k}_*^\intercal K_{tot}^{-1} \boldsymbol{k}_*, \tag{19}$$

with $K_{**} = k_{train}(\boldsymbol{z}_{draw}, \boldsymbol{z}_{draw})$. The predictive variance vector $\sigma^2_{tot}(\boldsymbol{z}_{draw})$ is the diagonal of $\Sigma_{tot}(\boldsymbol{z}_{draw})$. We propose to base the quality indicator on the mean predictive variance over all points $\boldsymbol{z}_{draw}$. More precisely, as K_{tot}^{-1} is positive definite, $\boldsymbol{k}_*^\intercal K_{tot}^{-1}\boldsymbol{k}_* > 0$ and thus $\forall i, \sigma^2_{toti}(\boldsymbol{z}_{draw}) \in [0, K_{**ii}]$, we propose the following normalised indicator:

$$\alpha = 1 - \frac{1}{N_{draw}} \sum_{i=1}^{N_{draw}} \frac{\sigma^2_{toti}(\boldsymbol{z}_{draw})}{K_{**ii}}, \tag{20}$$

where $\alpha \in [0, 1]$ and $\alpha \simeq 1$ denotes a low predictive variance, meaning a good approximation quality.

Note that for most of the usual kernel functions, the diagonal of K_{**} is filled with σ^2, the variance hyperparameter of the chosen kernel. This is notably the case with the exponential, squared exponential and Matern kernels. In these cases, using $\sigma^2 N_{draw} = \mathrm{Tr}\big(K_{**}\big)$ and linearity of the trace, the indicator reads:

$$\alpha = \frac{\mathrm{Tr}\big(\boldsymbol{k}_*^\intercal K_{tot}^{-1}\boldsymbol{k}_*\big)}{\sigma^2 N_{draw}}. \tag{21}$$

One may recognise a Frobenius norm at the numerator of the proposed indicator:

$$\begin{aligned} \mathrm{Tr}\big(\boldsymbol{k}_*^\intercal K_{tot}^{-1}\boldsymbol{k}_*\big) &= \mathrm{Tr}\big(\boldsymbol{k}_*^\intercal (L_{tot}L_{tot}^\intercal)^{-1}\boldsymbol{k}_*\big), \\ &= \mathrm{Tr}\big((L_{tot}^{-1}\boldsymbol{k}_*)^\intercal L_{tot}^{-1}\boldsymbol{k}_*\big), \\ &= ||L_{tot}^{-1}\boldsymbol{k}_*||_F^2, \end{aligned} \tag{22}$$

where L_{tot} refers again to the Cholesky decomposition of K_{tot}.

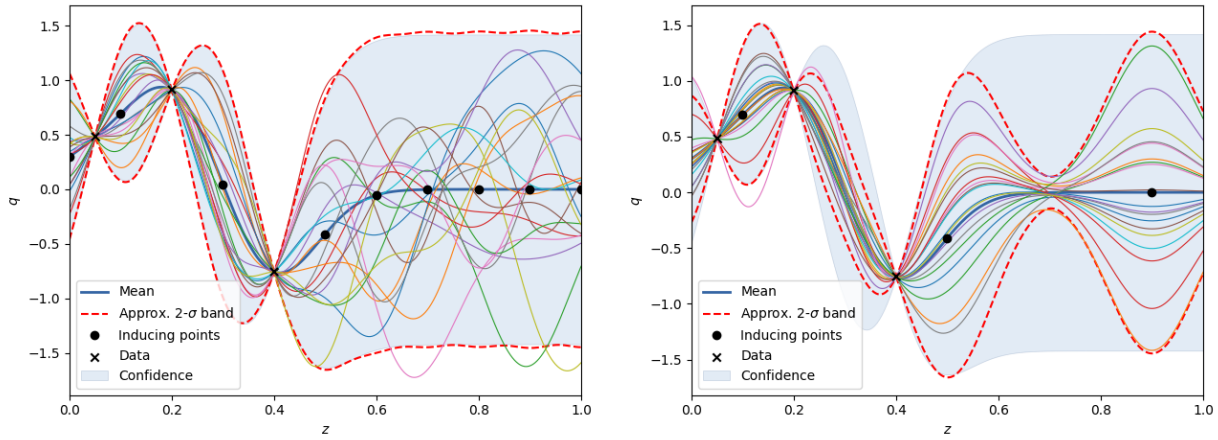
The proposed indicator is pictured in Figure 4 with nine and three inducing points, respectively. The approach gives an empirical 2-σ band very close to the real one in Figure 4(a) associated with a value of the quality indicator of 0.995. On the contrary, the approximated random field in Figure 4(b) is quite different from the underlying GP, and the quality indicator only has a value of 0.825.

For deeper understanding of this indicator, we study the behaviour of α on an analytical example. With D the dimension of the input parameters $\boldsymbol{z}$, we study the following quantity of interest:

$$q(\boldsymbol{z}) = \sin(\boldsymbol{L}^\intercal \boldsymbol{z}) = \sin\left(\sum_{i=1}^{D} L_i z_i\right). \tag{23}$$

To vary the complexity of q, we regard parameters D and $\boldsymbol{L}$ are random variables, which yields a random function $q_{D,\boldsymbol{L}}$. Here, we take $D \sim \mathcal{U}(\llbracket 1, 4 \rrbracket)$ and $\boldsymbol{L} \sim \mathcal{U}([3, 20])^D$.

For each sampled function $q_{D^{(k)},\boldsymbol{L}^{(k)}}$, a GP is built on 10 training data $q_{D^{(k)},\boldsymbol{L}^{(k)}}(\boldsymbol{z}_{train})$. A set of points $\boldsymbol{z}_{draw}$ is drawn uniformly within the input space. The predictive covariance K on these points is given by Equation (13). A set of well spread inducing points of size $N_{ind} \sim \mathcal{U}(\llbracket 10, 60 \rrbracket)$ is also drawn (*e.g.* with a maximin Latin Hypercube Sampling (LHS)



(a) (b)

Figure 4: Approximated random field with (a) 9 inducing points, $\alpha = 0.995$ and (b) 3 inducing points, $\alpha = 0.825$.

strategy) and 10^5 realisations $q^{(i)}_{D^{(k)},\boldsymbol{L}^{(k)}}(\boldsymbol{z}_{draw})$ are computed at 10^3 locations $\boldsymbol{z}_{draw}$. The empirical covariance between all these realisations gives an approximation $\widetilde{K}$ of K and the indicator α from Equation (20) is computed.

All this process is repeated 300 times, and we plot in Figure 5 the evolution of the normalised Frobenius distance between K and $\widetilde{K}$ with respect to α.

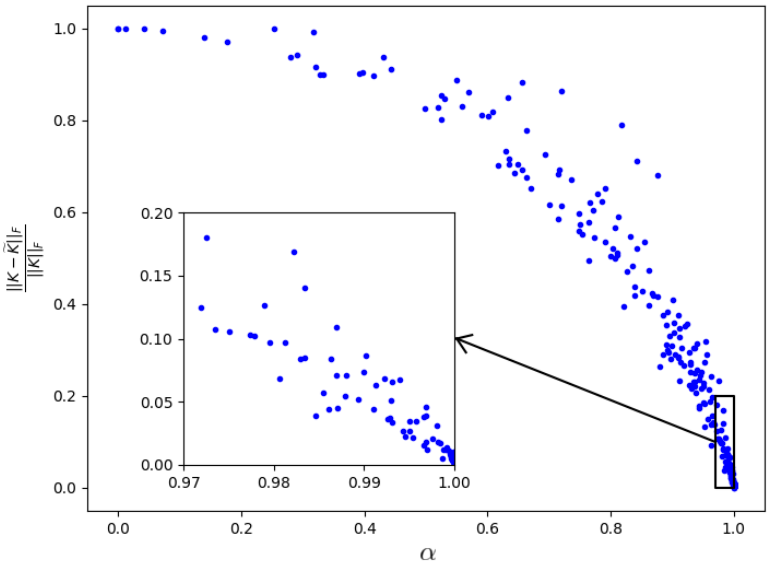


Figure 5: Accuracy of the empirical covariance $\widetilde{K}$ approximated with inducing points with respect to the proposed indicator.

Figure 5 shows that a strong correlation exists between the proposed indicator α and the accuracy of the empirical covariance. A value of the indicator very close to 1 seems to ensure an accurate representation of the random field. Notably, the zoomed window permits to see that when α is greater than 0.99 or 0.995, the worst recorded empirical covariances only yield 5 to 10% relative error.

Note that several appropriate norms for covariance matrices exist. Namely, the logarithmic, Cholesky-based, and Riemannian distances between covariance matrices allow to preserve positive definiteness and have many desirable properties [43]. However, because we draw very badly conditioned empirical covariances, very small negative eigenvalues may arise, that prohibit the use of such distances. For its simplicity and interpretability, the Frobenius distance is the most reliable choice.

3.2.3. Adaptive strategy

To take advantage of the proposed quality indicator, we propose an iterative algorithm ensuring that α is sufficiently close to 1 before returning the approximated realisations. The number of inducing points is hence increased as long as a prescribed value for α has not been reached. At each iteration, these points are spread in the input space using the classical maximin criterion on a Latin Hypercube Sampling (LHS). The algorithm is given in Algorithm 1.

Algorithm 1 Algorithm for choosing the number of inducing points

Choose threshold $\alpha_* \in [0, 1]$
Choose initial number of inducing points N_{init} and added per iteration N_{add}
Build a GP on the training data
$N_{ind} = N_{init}$
while True **do**
    Construct a DoE of N_{ind} points using a maximin LHS
    Compute α from Equation (20)
    if $\alpha \geq \alpha_*$ **then**
        Draw realisations with Equation (18) using the DoE of inducing points
        Return realisations
    else
        N_{ind} += N_{add}
    end if
end while

In the following, the threshold α_* will take values of 0.99 or 0.995, basing on Figure 5.

The simple approach proposed here is very intuitive and reveals very cost-efficient.

4. Application to Optimisation Under Uncertainty

In this section, we give a general technique, heavily inspired from the Surrogate-Assisted Bounding-Box approach introduced in [44, 31], with the aim of making use of the non-

parametric approximations presented in the previous section in an Optimisation Under Uncertainty (OUU) context.

A general formulation of the proposed approach is presented in Section 4.1, and its application with non-parametric measure approximation is made explicit in Section 4.2.

4.1. General formulation

The proposed approach integrates the concept of tunable accuracy within the OUU process. This concept has notably been presented in [17] and exploited in [30, 44, 31], with the aim of tuning the accuracy of each computation of $\boldsymbol{\rho}$ on the fly in order to focus computational power on the most promising designs.

In practice, robustness and reliability measures are computed using some evaluations of $\boldsymbol{q}$, and their approximation error can be estimated. We propose to treat these approximated measures as random vectors, that can be compared under the *ϵ-relaxed probabilistic constrained Pareto dominance* paradigm, proposed in Section 2.3, in order to discard all dominated approximations

Following the developments from [31], we further propose to rank all remaining measures with the POP_{min} metric, which is an approximation of the Pareto Optimal Probability (POP) with desirable properties. Using the notations from Section 3, the metric writes as follows:

$$POP_{min}\big(\boldsymbol{P}(\boldsymbol{x}_k)\big) = \min_{\substack{j\in[\![1,N_x]\!]\\ j\neq k}} \Big(\mathbb{P}_{\boldsymbol{P}}\big[\boldsymbol{P}(\boldsymbol{x}_j) \not\succ_c \boldsymbol{P}(\boldsymbol{x}_k)\big]\Big). \tag{24}$$

The accuracy is then adaptively tuned by improving these approximations only on the most promising individuals. We refer to this approach as Measure Approximation with Tunable Accuracy (MATA), as different levels of refinement coexist, depending on the estimated performances.

In addition, we construct a surrogate model on these measure approximations in the design space. This model is referred to in the following as the Surrogate-Assisting (SA) model. It directly yields predictions of the robustness and reliability measures at any design $\boldsymbol{x}$, so as to bypass $\boldsymbol{\rho}$ estimations when the SA model is sufficiently accurate. In essence, it permits to further lower the required number of function evaluations when the measures $\boldsymbol{\rho}$ are easier to model in $\mathcal{X}$ than $\boldsymbol{q}$ in the coupled $\mathcal{X}\times\Xi$ space.

These two methods (SA and MATA) couple very efficiently to lower the cost of an uncertainty-based optimisation process. Figure 6 gives a flowchart of the global strategy, called SAMATA. The specific notations are presented in the following sections 4.1.1 and 4.1.2.

4.1.1. Surrogate-Assisting strategy (SA)

Surrogate-Assisting (SA) strategies are common practice for accelerating the optimisation process. In the proposed approach, a surrogate model is constructed and updated throughout the optimisation process directly on the robustness and reliability measures $\boldsymbol{\rho}$ in the design space. For each visited design $\boldsymbol{x}_i$, an aleatoric approximation $\boldsymbol{P}(\boldsymbol{x}_i)$ is computed. The SA

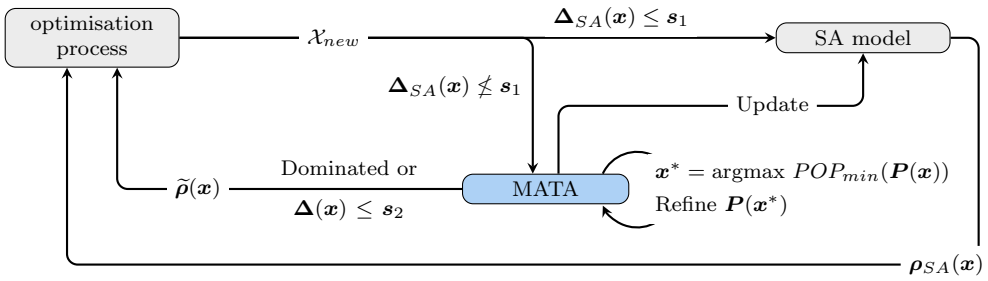


Figure 6: SAMATA strategy flowchart.

model then builds on the training set $\left\{\left(\boldsymbol{x}_i, \boldsymbol{P}(\boldsymbol{x}_i)\right)\right\}_i$. When enough information is gathered at a new design $\boldsymbol{x}_{new}$, the SA model bypasses the computation of $\boldsymbol{P}(\boldsymbol{x}_{new})$ and returns the current prediction $\boldsymbol{\rho}_{SA}(\boldsymbol{x}_{new})$. One can expect this approach to be extensively exploited at the end of the optimisation process, when a lot of training points are available and most new designs are gathered in the optimal area.

The SA model takes as training data aleatoric approximations of $\boldsymbol{\rho}$, and must yield an accuracy metric at any location. We propose to write the accuracy of the SA model at a new design $\boldsymbol{x}_{new}$ as a vector $\boldsymbol{\Delta}_{SA}(\boldsymbol{x}_{new})$. The SA prediction $\boldsymbol{\rho}_{SA}(\boldsymbol{x}_{new})$ is then directly returned to the optimisation process whenever $\boldsymbol{\Delta}_{SA}(\boldsymbol{x}_{new}) \leq \boldsymbol{s}_1$, where $\boldsymbol{s}_1$ is a user-defined threshold chosen beforehand..

Similarly to [31], we propose to carry out the whole optimisation process several times with decreasing values of $\boldsymbol{s}_1$. All training data $\left\{\left(\boldsymbol{x}_i, \boldsymbol{P}(\boldsymbol{x}_i)\right)\right\}_i$ are transfered from one iteration to another. This procedure allows to get several intermediate results, that compromise between computational cost and accuracy. With high $\boldsymbol{s}_1$ value, the number of evaluations of $\boldsymbol{q}$ is kept quite low as the SA model prediction $\boldsymbol{\rho}_{SA}$ is quickly exploited. On the contrary, with a small $\boldsymbol{s}_1$ value, many training data are required before employing the SA model, thus ensuring a better accuracy of $\boldsymbol{\rho}_{SA}$.

The structure of this approach is depicted in Algorithm 2. The computation and refinement of the probabilistic approximations $\boldsymbol{P}(\boldsymbol{x})$ (lines 4 and 12 of the algorithm) are described in the following section 4.1.2.

4.1.2. Measure Approximation with Tunable Accuracy (MATA)

Throughout the optimisation process, we denote $\mathcal{X}_c$ the set of designs at which an approximation $\boldsymbol{P}$ has been computed. At each iteration, among the new designs $\mathcal{X}_{new}$ given by the optimisation algorithm, all designs which computation is not bypassed by the SA strategy are added to $\mathcal{X}_c$ and first approximations $\boldsymbol{P}$ and $\boldsymbol{\Delta}$ are computed. Within this set, we denote by $\mathcal{X}_r$ the set of designs that must be further refined. These designs satisfy two criteria: (i) they belong to the current Pareto front $\mathcal{P}(\boldsymbol{P}(\mathcal{X}_c))$ and (ii) the user-defined accuracy threshold $\boldsymbol{s}_2$ is not reached. Formally:

$$\mathcal{X}_r = \left\{\boldsymbol{x} \in \mathcal{X}_c \mid \boldsymbol{\Delta}(\boldsymbol{x}) \not\leq \boldsymbol{s}_2 \;\&\; \boldsymbol{P}(\boldsymbol{x}) \in \mathcal{P}(\boldsymbol{P}(\mathcal{X}_c))\right\}, \tag{25}$$

where $\boldsymbol{\Delta}(\boldsymbol{x})$ is a chosen measure of the variability of $\boldsymbol{P}(\boldsymbol{x})$, here its standard deviation, and $\mathcal{P}$ corresponds to the non-dominated measures under the *ϵ-relaxed probabilistic constrained*

Algorithm 2 Algorithm of the Surrogate-Assisting strategy

Set initial thresholds $\boldsymbol{s}_1$
Initialise the training set $\mathcal{T} = \left\{\left(\boldsymbol{x}_{train_i}, \boldsymbol{P}(\boldsymbol{x}_{train_i})\right)\right\}_i$ empty
for k from 0 to $N_{threshold}$ **do**
    Refine approximations in $\mathcal{T}$ ▷ Alg. 3
    Launch optimiser
    while optimisation running **do**
        Update the SA model on $\mathcal{T}$
        Get new designs $\mathcal{X}_{new}$ to visit
        $\mathcal{X}_{SA} = \left\{\boldsymbol{x} \in \mathcal{X}_{new} \mid \boldsymbol{\Delta}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1\right\}$
        $\mathcal{X}_c = \mathcal{X}_{new} \setminus \mathcal{X}_{SA}$
        $\forall \boldsymbol{x} \in \mathcal{X}_{SA}$, return $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ to the optimiser
        $\forall \boldsymbol{x} \in \mathcal{X}_c$, compute $\boldsymbol{P}(\boldsymbol{x})$ and return $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ to the optimiser ▷ Alg. 3
        $\forall \boldsymbol{x} \in \mathcal{X}_c$, add $\left(\boldsymbol{x}, \boldsymbol{P}(\boldsymbol{x})\right)$ to $\mathcal{T}$
    end while
    Take next $\boldsymbol{s}_1$ value
end for

Pareto dominance paradigm.

Given this set of designs to refine, we propose to use the POP ranking (Eq. 24) to select the most promising design within $\mathcal{X}_r$. The variability of the associated approximation $\boldsymbol{P}(\boldsymbol{x})$ is then reduced, at the cost of some evaluations of $\boldsymbol{q}$. We denote $\boldsymbol{x}^*$ the design maximising the POP_{min} metric.

$$\boldsymbol{x}^* = \underset{\boldsymbol{x} \in \mathcal{X}_r}{\operatorname{argmax}}\ POP_{min}(\boldsymbol{P}(\boldsymbol{x})). \tag{26}$$

The MATA refinement technique, for a given set of designs $\mathcal{X}_c$ and a fixed threshold $\boldsymbol{s}_2$, is detailed in Algorithm 3.

Algorithm 3 Algorithm of the Measure Approximation with Tunable Accuracy technique

Read threshold $\boldsymbol{s}_2$ and design set $\mathcal{X}_c$
Compute first approximations $\boldsymbol{P}(\boldsymbol{x})$ or retrieve the ones that already exist
Extract $\mathcal{P}(\boldsymbol{P}(\mathcal{X}_c))$
Compute $\mathcal{X}_r$ with Equation (25)
while $\mathcal{X}_r$ is non-empty **do**
    Find $\boldsymbol{x}^*$ with Equation (26)
    Refine approximation $\boldsymbol{P}(\boldsymbol{x}^*)$ with some evaluations of $\boldsymbol{q}$
    Update $\mathcal{P}(\boldsymbol{P}(\mathcal{X}_c))$ and $\mathcal{X}_r$
end while
return $\boldsymbol{P}(\boldsymbol{x})$ for all $\boldsymbol{x}$ in $\mathcal{X}_c$

Similarly to the SA strategy, we propose to iterate over decreasing values of $\boldsymbol{s}_2$ in order to obtain intermediate approximations of the Pareto front.

4.2. Application to non-parametric measure approximations

The above SAMATA strategy is presented in its most general form. In practice, specific choices must be made on the computation of approximations $\boldsymbol{P}(\boldsymbol{x})$, metrics $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ and $\boldsymbol{\Delta}(\boldsymbol{x})$, and SA predictions $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ and $\boldsymbol{\Delta}_{SA}(\boldsymbol{x})$.

The inducing point strategy presented in Section 3.2.1 permits to generate realisations of $\boldsymbol{q}$ on an extensive set of points $\boldsymbol{z}_{draw}$, to empirically estimate realisations of the measures $\boldsymbol{P}$, *e.g.* using Equation (16). In short, the obtained empirical distribution of $\boldsymbol{P}$ comes from the Gaussian assumption of the GP surrogate model built on $\boldsymbol{q}$, composed with the specific statistical estimator.

The mean of this approximated distribution is returned to the optimisation process as $\widetilde{\boldsymbol{\rho}}$, and three times the standard deviation is used as accuracy metric $\boldsymbol{\Delta}$. Concerning the refinements, we use the criterion proposed in [31], corresponding to a linear combination of measure-specific criteria.

Finally, the SA model is constructed in a specific way to best exploit all the available information. SA-based measure approximations should take the form of set of samples and be jointly drawn with the realisations obtained in (16). To this extent, we propose to define the SA-based realisations as random draws of a GP surrogate model conditioned on the computed joint realisations $\boldsymbol{P}_c^{(i)}$. In practice, on any set of design $\boldsymbol{x}_{SA}$, SA-based measure realisations $\boldsymbol{P}_{SA}^{(i)}(\boldsymbol{x}_{SA})$ are computed as such:

$$\boldsymbol{P}_{SA}^{(i)}(\boldsymbol{x}_{SA}) = \boldsymbol{\mu}_{SA}^{(i)}(\boldsymbol{x}_{SA}) + L_{SA}\boldsymbol{Y}^{(i)}, \quad \boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{0}, I), \tag{27}$$

where the predictive mean writes:

$$\boldsymbol{\mu}_{SA}^{(i)}(\boldsymbol{x}_{SA}) = \boldsymbol{k}_*^\intercal K_c^{-1} \boldsymbol{P}_c^{(i)}, \tag{28}$$

and L_{SA} results from a Cholesky decomposition performed on the predictive covariance $\Sigma_{GP}(\boldsymbol{x}_{SA})$ defined in Equation (13). In the above, $\boldsymbol{P}_c^{(i)}$ refers to realisations drawn from Equation (16), on the set designs $\boldsymbol{x}_c$. K_c corresponds to the autocovariance matrix $k(\boldsymbol{x}_c, \boldsymbol{x}_c)$ and $\boldsymbol{k}_*$ to the covariance $k(\boldsymbol{x}_c, \boldsymbol{x}_{SA})$. Note that hyperparameters are only optimised once for numerical efficiency.

This SA model construction is illustrated on an example set of samples in Figure 7, where realisations of $\boldsymbol{P}$ are jointly drawn at ten locations in ascending order. Two bimodal sets of samples are notably drawn at $x = 0.15$ and $x = 0.45$ in order to emphasise the need for non-parametric reconstruction. Some SA-based functional realisations $\boldsymbol{P}_{SA}^{(i)}$ form Equation (27) are represented. The two bi-modal densities are naturally well captured using the proposed approach. All computed $\boldsymbol{P}_c^{(i)}$ and SA-based $\boldsymbol{P}_{SA}^{(i)}$ realisations are jointly redrawn at each computed measure refinement, in order to use the joint dominance probability from Equation (12).

These formulations can be readily incorporated within the SAMATA strategy, presented in Section 4.1 and pictured in Figure 6. The specific algorithm is given in Algorithm 4.

Finally, specific numerical choices such as the optimisation algorithm and the threshold values $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ will be made case by case.

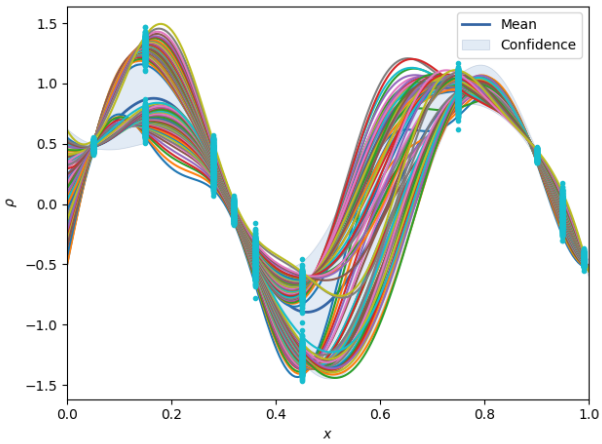


Figure 7: Realisations of the SA model. Training data in light blue and underlying GP model in dark blue, in the background.

Algorithm 4 SAMATA algorithm with joint measure approximations

Launch N_{init} evaluations $\boldsymbol{q}(\boldsymbol{x}_i, \boldsymbol{\xi}_i)$
Generate a first coupled space surrogate model $\widehat{\boldsymbol{q}}$
Loop over values of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$
while optimisation running **do**
    Get new designs $\mathcal{X}_{new}$
    Initialise $\mathcal{X}_c$ empty
    Compute $\boldsymbol{P}_{SA}$ for all $\boldsymbol{x} \in \mathcal{X}_{new}$ from (27)
    Compute $\mathcal{X}_{SA}^0 = \left\{\boldsymbol{x} \in \mathcal{X}_{new} \mid \boldsymbol{\Delta}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1\right\}$ and $\mathcal{X}_c^0 = \mathcal{X}_{new} \setminus \mathcal{X}_{SA}^0$
    Deduce $\mathcal{X}_r^0$ from $\mathcal{X}_c^0$ using (25)
    $k = 0$
    while $\mathcal{X}_r^k \neq \emptyset$ **do**
        Compute $\forall \boldsymbol{x} \in \mathcal{X}_{new}, POP_{min}(\boldsymbol{x})$ w.r.t. all designs using (24) and (12)
        Find $\boldsymbol{x}^*$ from $\mathcal{X}_r^k$ using (26)
        if $\boldsymbol{x}^* \notin \mathcal{X}_c$ **then**
            Add $\boldsymbol{x}^*$ in $\mathcal{X}_c$
        end if
        Find $\boldsymbol{\xi}^*$ that maximises the refinement criterion from [31]
        Launch evaluation of $\boldsymbol{q}(\boldsymbol{x}^*, \boldsymbol{\xi}^*)$
        Update the coupled space surrogate model $\widehat{\boldsymbol{q}}$
        Update all approximations $\boldsymbol{P}_c^{(i)}(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{X}_c$ using empirical estimators like (16)
        Update $\boldsymbol{P}_{SA}$ for all $\boldsymbol{x} \in \mathcal{X}_{new} \setminus \mathcal{X}_c$
        Compute $\mathcal{X}_c^{k+1}$, $\mathcal{X}_{SA}^{k+1}$ and $\mathcal{X}_r^{k+1}$
        $k = k + 1$
    end while
    Return $\mathbb{E}\big[\boldsymbol{P}(\boldsymbol{x})\big]$ to the optimiser for all $\boldsymbol{x} \in \mathcal{X}_{new}$
end while
Return all measure approximations $\boldsymbol{P}(\boldsymbol{x})$ and their associated POP_{min} score

5. Analytic comparison

In this section, the proposed SAMATA approach depicted in Alg. 4 is quantitatively compared to SABBa from [31] with Coupled-Space surrogate models, which was shown very cost efficient. As a reference, we also compare these techniques to an A Priori Metamodel strategy, where a Coupled-Space surrogate model is constructed in the coupled $\mathcal{X} \times \Xi$ space on a given number of data points and is readily employed as is for measure estimation and optimisation.

The three approaches are compared in terms of the expected modified Hausdorff distance between the approximated Pareto optimal designs and the true ones. Such a metric was proposed in [31] in the context of Bounding-Box approach. Here, by construction, joint realisations $\boldsymbol{P}^{(i)}(\boldsymbol{x})$ of the aleatory measure approximations can be drawn simultaneously on a set of designs $\boldsymbol{x}$ using (16) and (27). This readily permits to employ the aforementioned metric, which formula is recalled here:

$$Q_{\mathcal{B}} = \mathbb{E}_{\widetilde{\mathcal{X}}_{\mathcal{P}}}\left[d'_H(\mathcal{X}_{\mathcal{P}}, \widetilde{\mathcal{X}}_{\mathcal{P}})\right], \tag{29}$$

where d'_H is the modified Hausdorff distance from [45], $\mathcal{X}_{\mathcal{P}}$ is the Pareto front pre-image, and $\widetilde{\mathcal{X}}_{\mathcal{P}}$ is an aleatory variable corresponding to the pre-image of the currently approximated aleatory Pareto front. This expected value is empirically estimated, exploiting the capability of drawing joint realisations of $\boldsymbol{P}$. Note that in a normalised input space, a value $Q_{\mathcal{B}} = 1$ corresponds to a bad score while $Q_{\mathcal{B}} = 10^{-2}$ reveals a very accurate approximation of the Pareto front.

The NSGA-II optimisation algorithm is used in the following test-cases. The number of individuals per generation is set to 32. Both the optimisation and refinement processes are sequential, and the coupled-space surrogate $\hat{q}$ is initialised with a Latin Hypercube Sampling (LHS) of 30 points.

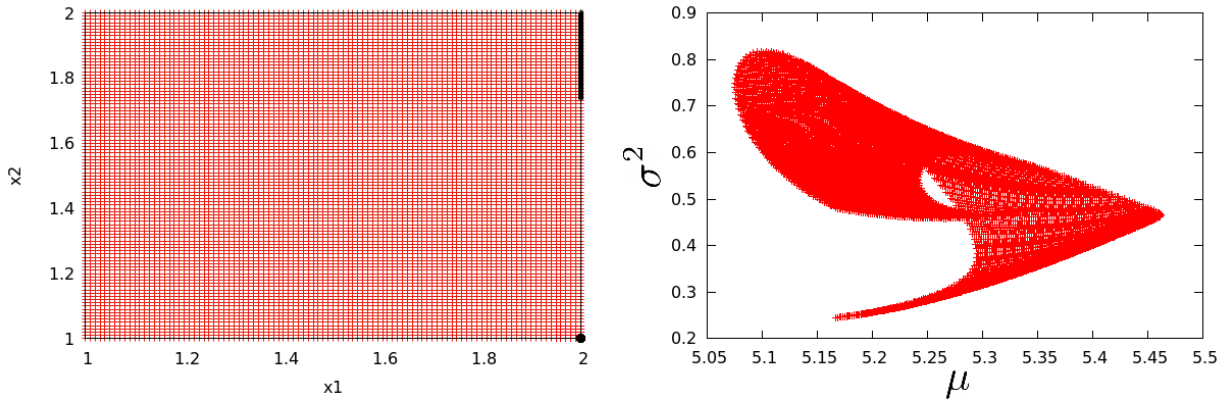
Each technique is run ten times to get the mean and standard deviation of the convergence curves.

5.1. Test-case 1: Unconstrained Taguchi optimisation

This problem is a bi-objective robust optimisation proposed in [30]. There are two design variables $\boldsymbol{x}$ and one uncertain parameter ξ. The problem reads:

$$\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \begin{pmatrix} \mu(\boldsymbol{x}) \\ \sigma^2(\boldsymbol{x}) \end{pmatrix} \\
\text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_\xi[q(\boldsymbol{x}, \xi)] \\
& \sigma^2(\boldsymbol{x}) = \mathbb{V}_\xi[q(\boldsymbol{x}, \xi)] \\
\text{with:} \quad & q(\boldsymbol{x}, \xi) = \xi - x_1\xi^5 + \cos(2\pi x_2 \xi) + 5 \\
& \xi \sim \mathcal{U}([0, 1]) \\
\text{by changing:} \quad & (x_1, x_2) \in [1, 2]^2
\end{aligned} \tag{30}$$

The Pareto front associated with this problem is discontinuous, and the optimal set in the design space consists in segment and a point (see Fig. 8).



(a) (b)

Figure 8: 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 performance of the three compared strategies is depicted in Figure 9, showing the mean convergence over ten runs and the Bollinger Band corresponding to the associated standard deviation. Both the Bounding-Box and the proposed non-parametric SAMATA approaches show a steeper convergence than the A Priori Metamodel technique. In addition, the non-parametric approach reveals significantly more parsimonious than the other two techniques, reaching a $Q_{\mathcal{B}}$ metric of 10^{-2} around 60 function evaluations, corresponding to a gain of nearly one order of magnitude over the reference methods.

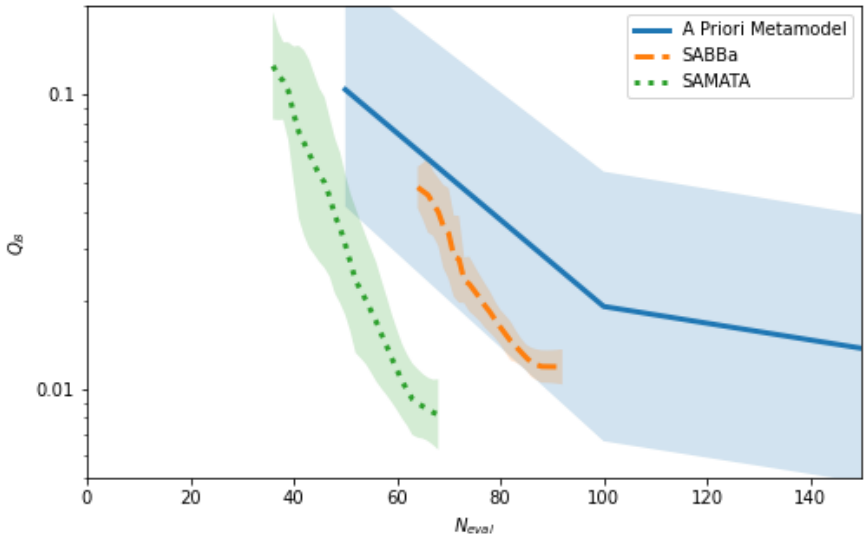


Figure 9: Test-case 1: $Q_{\mathcal{B}}$ metric for the A Priori Metamodel (solid blue), SABBa (dashed orange) and the proposed non-parametric SAMATA (dotted green) approaches.

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

This second test-case, proposed in [31], assesses the performance of the proposed non-parametric approach on a mono-objective optimisation problem with a reliability based constraint, in a higher dimensional context. The objective is a robustness measure derived from a simplified Rosenbrock function and the constraint a quantile formulation of the Six-Hump Camel function. It features four design variables and three uncertain parameters, and is stated as follows:

$$
\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mu(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = \mathrm{q}^{95\%}(\boldsymbol{x}) \leq 1 \\
\text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[q_1(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \mathrm{q}^{95\%}(\boldsymbol{x}) = \mathrm{q}^{95\%}_{\boldsymbol{\xi}}[q_2(\boldsymbol{x}, \boldsymbol{\xi})] \\
\text{with:} \quad & q_1(\boldsymbol{x}, \boldsymbol{\xi}) = \sum_{i=1}^{3} \left[\big(1 - x_i\big) + 3 \left(1 + \frac{\arctan\big(5(\xi_i - 0.5)\big)}{2} \right) \big(x_{i+1} - x_i^2\big)^2 \right] \\
& q_2(\boldsymbol{x}, \boldsymbol{\xi}) = \left(4 - 2.1 x_1^2 + \frac{x_1^4}{3} \right) x_1^2 + x_1 x_2 + \big(-4 + 4 x_2^2 \big) x_2^2 \\
& \qquad\qquad + \frac{\cos(2\pi\xi_1) - \sin(\frac{\pi}{2}\xi_1) - \xi_1 - \big(\cos(2\pi 0.05) - \sin(\frac{\pi}{2} 0.05) - 0.05 \big)}{5} \\
& \boldsymbol{\xi} \sim \mathcal{U}([0,1]^3) \\
\text{by changing:} \quad & \boldsymbol{x} \in [-0.2, 1.2]^4
\end{aligned}
\tag{31}
$$

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

The convergence curves are depicted in Figure 10. Similarly to Figure 9, the A Priori Metamodel is the less efficient strategy, and the proposed non-parametric SAMATA approach is the most parsimonious. The standard deviation over the ten runs are greater in this test-case, but the proposed method permits to reach a metric of 10^{-1} for less than 70 function evaluations, against 110 for the Bounding-Box approach and more than 200 with the A Priori Metamodel.

5.3. Benefit of sampling-based approximations

The so-called projected processes techniques [32, 33, 34, 35] permit to solve Optimisation Under Uncertainty (OUU) problems in a Bayesian Optimisation setting (also called Efficient Global Optimisation, EGO). They however rely on independent Gaussian approximations of the robustness and reliability measures. Such Gaussian approximations are cheap to estimate and particularly adapted for expectation-like measures, but might fail to accurately capture more complex or correlated distributions. For deeper insights on this method, see [33].

We run ten optimisations of the first test-case using the proposed approach, but replacing the jointly drawn realisations of $\boldsymbol{P}$ with samples from a Gaussian distribution which mean and variance correspond to the empirical moments from the initial samples. In this manner,

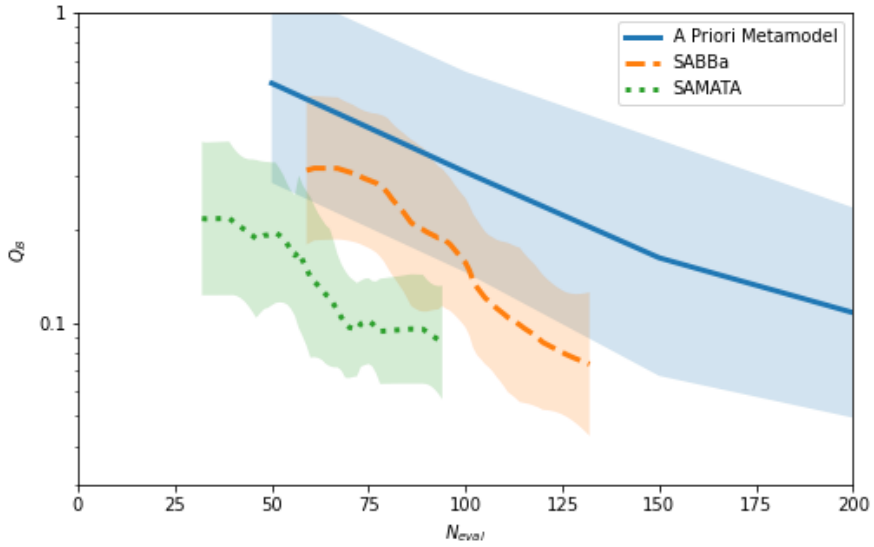


Figure 10: Test-case 2: $Q_{\mathcal{B}}$ metric for the A Priori Metamodel (solid blue), SABBa (dashed orange) and the proposed non-parametric SAMATA (dotted green) approaches.

we aim to investigate solely the impact of using a different density estimation, with either the proposed sample-based strategy or a simple moment-based Gaussian fit.

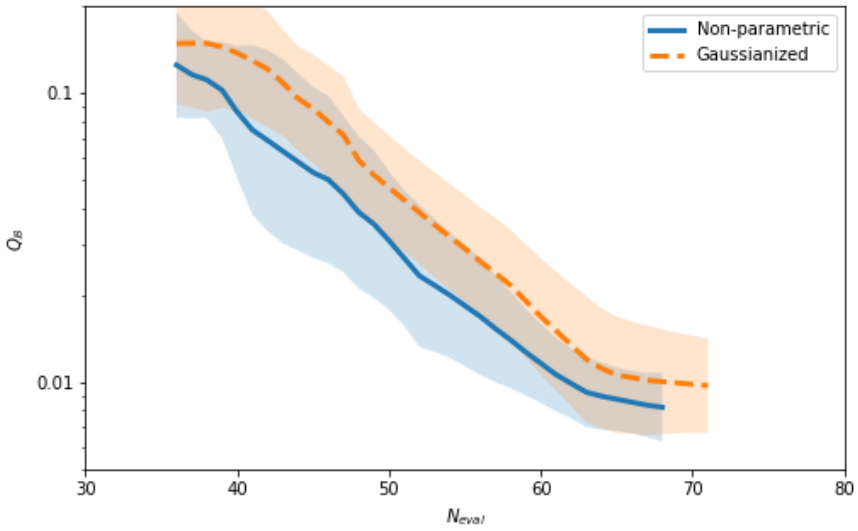


Figure 11: Convergence curves of the non-parametric SAMATA strategy (solid blue) against its Gaussianised equivalent (dashed orange). Variability represented with the transluscent $\pm\sigma$ band.

Figure 11 shows the two convergence curves, with their Bollinger bands. Although the convergence seems to plateau around the same metric value, the Gaussianised approach

shows slightly inferior accuracy compared to the original non-parametric strategy along the whole study. Intuitively, we argue that accurate distribution shapes and dependencies are lost when Gaussian density is forced, which may deteriorate the intermediate approximations of the Pareto optimal set.

6. ORC application

We perform here a robust shape optimisation of a typical converging-diverging turbine nozzle for ORC applications. It was extensively studied in the context of deterministic [46, 47, 48] and multi-point optimisation [49]. Some recent works tackled such design under epistemic uncertainties [50] due to turbulence modelling, and investigated the robustness of the blade design under aleatoric uncertainties in the operating conditions, thermodynamic model parameters and geometric tolerances [51].We propose here to minimise the mean of the performance function while the mean mass-flow rate is constrained to lie within a prescribed range centered on the nominal value.

The Biere is a reference two-dimensional benchmark geometry to test the design of devices operating with the siloxane fluid MDM (Octamethyltrisiloxane, $C_8H_{24}O_2Si_3$. This blade profile is meant to obtain a convergent-divergent cascade passage which serves to accelerate the fluid up to supersonic speed. Compressibility effects play a crucial role and a typical fish-tail shock pattern is generated downstream the trailing edge. Strong shocks may cause significant losses, design of the trailing edge region is thus critical for the turbine efficiency. The Mach field is depicted in Figure 12 with the baseline blade profile.

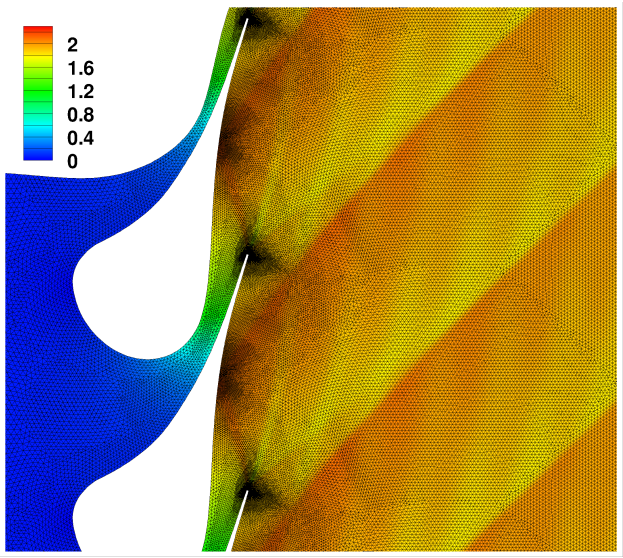


Figure 12: Mach contours at nominal conditions for the baseline profile and computational grid of 36k cells.

6.1. Problem setting

The objective function ΔP is defined as the standard deviation of the azimuthal distribution of static pressure half an axial chord downstream of the blade Trailing Edge (TE).

Minimising ΔP should yield a significant reduction of the shock strength. As mentioned above, a constraint is imposed on the mass-flow rate per unit span $\dot{m}$, normalised with respect to the nominal value.

Uncertainties on the operating conditions are considered both at the inlet and outlet of the turbine, through changes of boundary conditions. Following [52], we model the operational variability as independent and uniform uncertain variables, gathered in $\boldsymbol{\xi} = [P_{in}^t, T_{in}^t, P_{out}^s]$. The associated ranges are reported in Table 1, alongside Nominal and Mean conditions. Uncertainties on the parameters of the thermodynamical model are neglected since previous studies [53] provided evidence of their limited impact.

$\boldsymbol{\xi}$	P_{in}^t [Pa]	T_{in}^t [K]	P_{out}^s [Pa]
Nominal ($\boldsymbol{\xi}_0$)	8×10^5	545.15	1.072×10^5
Mean ($\boldsymbol{\xi}_\mu$)	8×10^5	545.15	1.5×10^5
Random	$\mathcal{U}[7.95 \times 10^5, 8.05 \times 10^5]$	$\mathcal{U}[541.15, 549.15]$	$\mathcal{U}[1 \times 10^5, 2 \times 10^5]$

Table 1: Operating Conditions: Nominal, Mean and Random.

The optimisation problem follows a multi-objective Taguchi formulation:

$$\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \begin{pmatrix} \mu_1(\boldsymbol{x}) \\ \sigma^2(\boldsymbol{x}) \end{pmatrix} \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = \mu_2(\boldsymbol{x}) \in [0.98\dot{m}_0, 1.02\dot{m}_0] \\
\text{where:} \quad & \mu_1(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\Delta P(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \sigma^2(\boldsymbol{x}) = \mathbb{V}_{\boldsymbol{\xi}}[\Delta P(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \mu_2(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\dot{m}(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \dot{m}_0 = 344.843 \ kg.s^{-1} \\
\text{with:} \quad & \boldsymbol{\xi} \sim \mathcal{U}(\Xi) \\
\text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}
\end{aligned} \tag{32}$$

where $\Xi = [7.95, 8.05] \times [541.15, 549.15] \times [1, 2]$. The design space $\mathcal{X}$ allows to modify the shape of the blade as much as possible while ensuring convergence of the numerical simulation. In practice, it consists of 9 of the 30 Control Points (CP) of a B-spline curve of degree 3, that can be displaced in the direction normal to the baseline geometry. Such parametrisation was proposed in [54] and is illustrated in Figure 13.

A mono-objective version of (32) is also treated for comparison purposes. Results will also be compared with two deterministic optima obtained at the Nominal and Mean operating conditions from Table 1.

Simulation is performed using the the Non-Ideal Compressible-Fluid Dynamics solver included in the SU2 [55, 56, 57, 58] suite on unstructured grids, generated using an in-house meshing tool. A dedicated mesh deformation tool based on Radial Basis Functions (RBF) allows high flexibility and robustness while maintaining the grid connectivity throughout the optimisation steps.

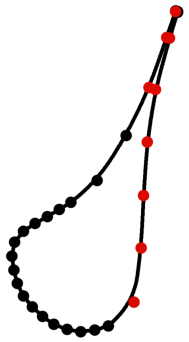


Figure 13: B-splines parametrisation. Fixed CP in black, moving CP in red.

6.2. Mono-objective results

Because nominal conditions are not centered within the uncertain range, we conduct the reference deterministic optimisations both at Nominal and Mean conditions. Similarly to [59], a state-of-the-art Bayesian Optimisation strategy is employed, with an Expected Improvement (EI) criterion multiplied by the Probability of Feasibility (PF) [60]. Convergence is reached after roughly **100** function evaluations.

The robust optimisation problem is solved using the proposed non-parametric SAMATA strategy. The NSGA-II optimisation procedure is again exploited, and constraints are handled through penalisation. The initial DoE consists of 30 evaluations drawn with a maximin Latin Hypercube Sampling (LHS) in the coupled space. Each population contains 32 individuals, and measure refinements are performed sequentially. We iteratively take threshold values of 10%, 5%, 3%, 2% and 1%.

Convergence values are gathered in Table 2, with a final optimum centered on $\boldsymbol{\rho_f} = 8087$, with a standard deviation of the approximation error of 44.26. This optimum satisfies the 1% accuracy imposed by the normalised thresholds and only required **145** simulation calls, which is not significantly higher than the deterministic optimisation. The proposed approach permits to reduce by around one order of magnitude the number of required function evaluations with respect to [59] that employed a more classical nested approach.

N_{eval}	$\boldsymbol{\rho_f}$ [Pa]		$\boldsymbol{\rho_g}$ $[kg.s^{-1}]$	
	Mean	Std. dev.	Mean	Std. dev.
76	8.328×10^3	6.355×10^2	3.464×10^2	1.065×10^{-1}
91	8.854×10^3	3.986×10^2	3.514×10^2	3.149×10^{-2}
106	8.869×10^3	1.124×10^2	3.387×10^2	1.534×10^{-2}
118	8.288×10^3	1.068×10^2	3.500×10^2	1.144×10^{-2}
145	8.087×10^3	4.426×10^1	3.505×10^2	1.812×10^{-2}

Table 2: Optimal outputs associated to the different threshold values.

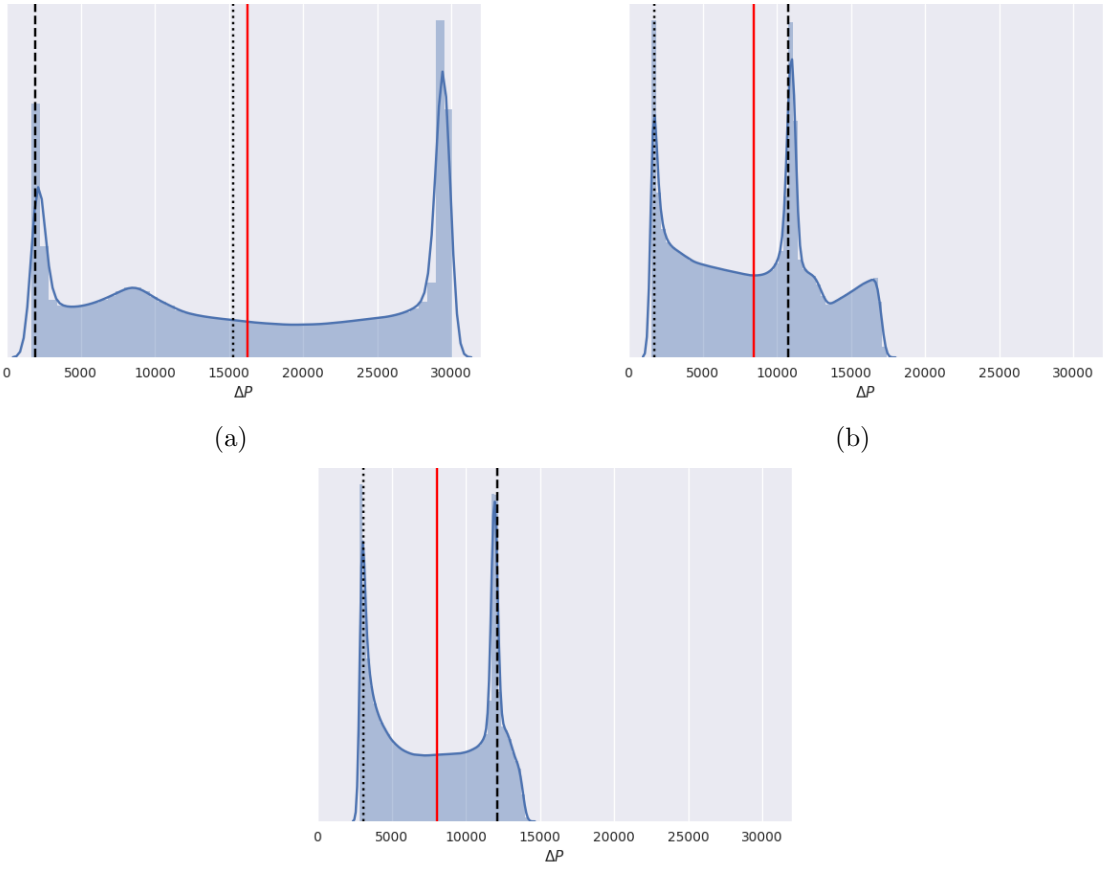
The point-wise ($\boldsymbol{\xi}_0$ and $\boldsymbol{\xi}_\mu$) and mean values of the objective function are computed at the Nominal $\boldsymbol{x}_0^*$, Mean $\boldsymbol{x}_\mu^*$ and Robust $\boldsymbol{x}_{OUU}^*$ operating conditions. Results are gathered in

519 Table 3 and the full distributions of ΔP over $\boldsymbol{\xi}$ at these given designs are plotted in Figure
520 14. These distributions are obtained with an extensive Monte Carlo on a GP built over Ξ
521 at a given $\boldsymbol{x}$, with 50 training points.

Blade profile	$\Delta P(\boldsymbol{x}, \boldsymbol{\xi}_0)$ [Pa]	$\Delta P(\boldsymbol{x}, \boldsymbol{\xi}_\mu)$ [Pa]	$\mathbb{E}_{\boldsymbol{\xi}}[\Delta P(\boldsymbol{x}, \boldsymbol{\xi})]$ [Pa]
Nominal optimum $\boldsymbol{x} = \boldsymbol{x}_0^*$	1.958×10^3	1.526×10^4	1.624×10^4
Mean optimum $\boldsymbol{x} = \boldsymbol{x}_\mu^*$	1.079×10^4	1.773×10^3	8.472×10^3
Robust optimum $\boldsymbol{x} = \boldsymbol{x}_{OUU}^*$	1.210×10^4	3.082×10^3	8.084×10^3

Table 3: Nominal values and statistics for different blade profiles.

522 As expected, the Nominal optimum yields very high variability in Fig. 14(a) but very
523 low nominal output. The Mean and Robust optima are quite similar, Figs 14(b) and 14(c).
524 However, the Robust optimal blade manages to reach a lower value for the expectation of
525 ΔP, with a slightly modified distribution shape.



(a)

(b)

(c)

Figure 14: Distribution of ΔP at (a) $\boldsymbol{x}_0^*$, (b) $\boldsymbol{x}_\mu^*$ and (c) $\boldsymbol{x}_{OUU}^*$. $\Delta P(\boldsymbol{x}, \boldsymbol{\xi}_0)$ represented as a dashed black line, $\Delta P(\boldsymbol{x}, \boldsymbol{\xi}_\mu)$ as a dotted black line and $\mathbb{E}_{\boldsymbol{\xi}}[\Delta P(\boldsymbol{x}, \boldsymbol{\xi})]$ as a plain red line.

6.3. Bi-objective results

We then tackle the complete Taguchi formulation (32) of the Optimisation Under Uncertainty of the ORC turbine blade. This formulation aims to find the Pareto front between good expected performance and low variability. The proposed non-parametric SAMATA algorithm is applied, with again the NSGA-II optimisation method and normalised thresholds of 10%, 5%, 2% and 1%.

The Pareto fronts associated with these threshold values are depicted in Figure 15, and reveal great improvements in the Pareto front accuracy throughout the iterations. The total computational cost for attaining the chosen thresholds is of **72**, **122**, **276** and **419** simulation calls respectively. Each cloud of samples associated with a given design has its transparency driven by its POP value (1: fully visible, 0: transparent). The mono-objective optimum found in the preceding section is represented as a red dot and lies on the Pareto front, as expected.

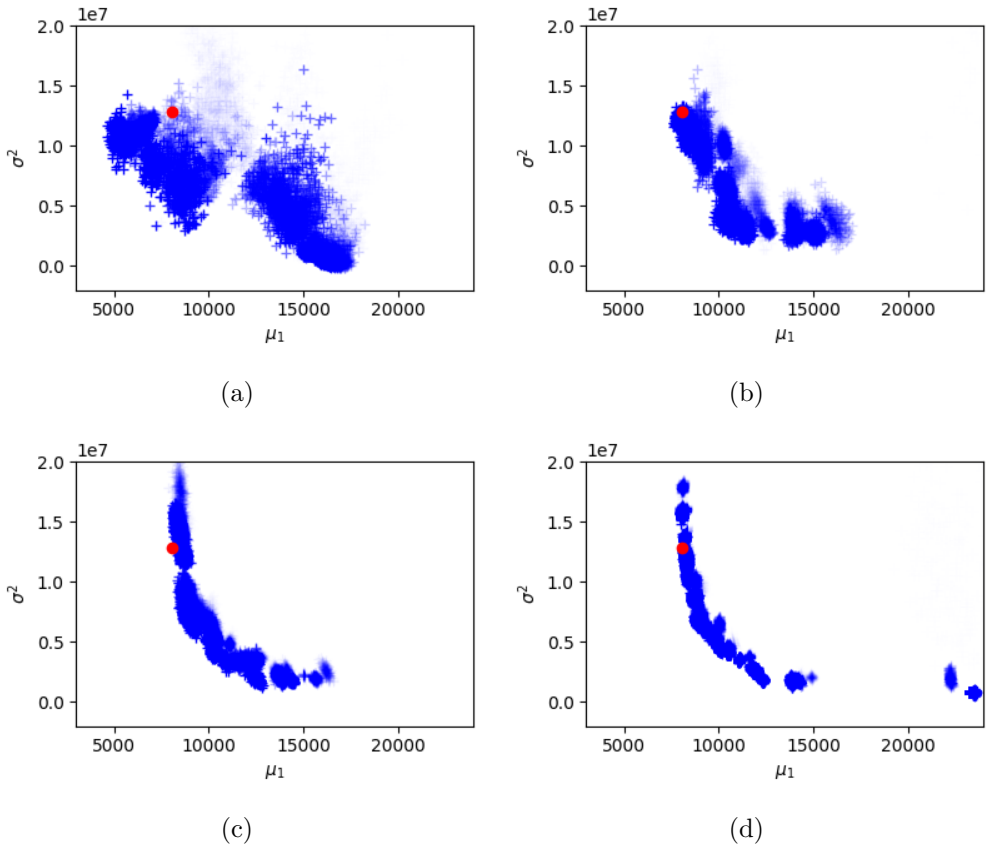


Figure 15: Pareto front of the bi-objective OUU problem with normalised threshold values of (a) 10% (**72** evaluations), (b) 5% (**122** evaluations), (c) 2% (**276** evaluations) and (d) 1% (**419** evaluations). Transparency driven by POP value and mono-objective optimum from previous section as a red dot.

7. Conclusions

This paper describes an efficient strategy for solving multi-objective and constrained Optimisation Under Uncertainty problems. It consists in coupling two primary techniques: i) a Measure Approximation with Tunable Accuracy (MATA) approach that only refines promising individuals up to a given threshold, ii) a Surrogate-Assisting (SA) strategy that constructs a surrogate model of the objective and constraint functions in the design space to bypass further evaluations when it is sufficiently accurate.

To apply this framework in practice, we propose non-parametric approximations of the error distributions using a sampling-based technique. The robustness and reliability measures for a given design are represented as a set of realisations that may exhibit complex shapes and dependencies. These realisations usually reveal more representative and can be jointly drawn between several designs. Issues associated with drawing such joint realisations are highlighted, and an adaptive algorithm is proposed to keep the associated cost manageable. This strategy show similarities with Nyström approximation for Karhunen-Loève Expansion (KLE) but allows for intuitive control of the overall accuracy and adaptive sampling. A Surrogate-Assisting (SA) model that takes advantage of these joint realisations is introduced to apply the SAMATA strategy. Quantitative comparisons with state-of-the-art approaches on several algebraic test cases reveal a drastic increase in the overall parsimony.

Finally, the proposed approach is applied to an engineering problem dealing with the geometric optimisation of the blade profile in an Organic Rankine Cycle turbine. Convergence is attained for a cost of 100 to 150 evaluations in a mono-objective setting, which is one order of magnitude less than the state-of-the-art approach. The Pareto front of a bi-objective formulation is accurately recovered at the cost of 100 to 400 evaluations.

Several perspectives can be drawn. Regarding overall parsimony, cost improvement could be achieved by improving the Uncertainty Propagation (UP) techniques associated with specific statistical measures. For example, extreme quantile estimation and refinement should be performed with appropriate techniques such as [61] or [62]. Such approaches can be readily incorporated into the proposed framework. Similarly, we only performed sequential refinement in this work, while techniques such as Kriging believer can be exploited to return a set of new locations. Multi-point refinement is critical to take advantage of nowadays' high parallelisation capacities of computational clusters. Finally, the proposed non-parametric error estimation relies on coupled-space surrogate models that may become unfeasible in high dimensional uncertain space or when the simulator is intrinsically stochastic. Specific developments for using SAMATA in this context would effectively widen its range of application.

Appendix A. Link between inducing points and Nyström method for KLE

In practice, the proposed approach is equivalent to constructing a Karhunen-Loève Expansion (KLE) of a Gaussian random field, when computing the basis functions with Nyström approximation. Computational details are given hereafter to show the equivalence.

To this extent, note that we can write the eigendecomposition (also called spectral decomposition) of the autocovariance matrix K_{tot} from Equation (18), which is Symmetric Positive Definite (SPD) by definition:

$$K_{tot} = U_{tot}\Lambda_{tot}U_{tot}^{\intercal}, \tag{A.1}$$

where Λ_{tot} is the diagonal matrix containing the eigenvalues associated with the eigenvectors stored in the columns of U_{tot}. To make this decomposition unique, the eigenvalues are sorted in decreasing order, $\Lambda_{tot_{11}} \geq \Lambda_{tot_{22}} \geq ... \geq 0$, and the eigenvectors are orthonormal, so that $U_{tot}^{\intercal} = U_{tot}^{-1}$, which gives:

$$U_{tot_{.i}}^{\intercal}U_{tot_{.j}} = \sum_{k} U_{tot_{ki}}U_{tot_{kj}} = \delta_{ij}, \tag{A.2}$$

with δ_{ij} the Kronecker delta.

The inverse of K_{tot} can thus be written:

$$K_{tot}^{-1} = U_{tot}\Lambda_{tot}^{-1}U_{tot}^{\intercal}, \tag{A.3}$$

and the square root $K_{tot}^{-1/2}$ of this matrix is:

$$K_{tot}^{-1/2} = U_{tot}\Lambda_{tot}^{-1/2}, \tag{A.4}$$

which verifies $K_{tot}^{-1} = K_{tot}^{-1/2}K_{tot}^{-1/2\intercal}$.

KLE and Nyström approximation. A realisation of a random field described with a truncated Karhunen-Loève Expansion reads as follows:

$$\boldsymbol{q}_{KLE}^{(i)}(\boldsymbol{z}) = \sum_{j=1}^{N_{KLE}} \sqrt{\lambda_j}\phi_j(\boldsymbol{z})\theta_j^{(i)}, \tag{A.5}$$

where the eigenvalues λ_j and eigenvectors ϕ_j are solution of the following homogeneous Fredholm integral equation of the second kind:

$$\begin{aligned} &\int k(\boldsymbol{z},\boldsymbol{z}')\phi_j(\boldsymbol{z})\mathrm{d}\boldsymbol{z} = \lambda_j\phi_j(\boldsymbol{z}') \\ \text{with} \quad &\int \phi_i(\boldsymbol{z})\phi_j(\boldsymbol{z})\mathrm{d}\boldsymbol{z} = \delta_{ij}. \end{aligned} \tag{A.6}$$

As presented in [63], the Nyström approximation consists in computing an empirical estimation of the above integral. This Monte Carlo approximation based on points $\boldsymbol{x}_k$ gives:

$$\begin{aligned} &\frac{1}{N_{MC}}\sum_{k=1}^{N_{MC}} k(\boldsymbol{z}_k,\boldsymbol{z}')\phi_j(\boldsymbol{z}_k)\mathrm{d}\boldsymbol{z} \approx \lambda_j\phi_j(\boldsymbol{z}') \\ \text{with} \quad &\frac{1}{N_{MC}}\sum_{k=1}^{N_{MC}} \phi_i(\boldsymbol{z}_k)\phi_j(\boldsymbol{z}_k)\mathrm{d}\boldsymbol{z} \approx \delta_{ij}. \end{aligned} \tag{A.7}$$

Replacing $\boldsymbol{z}'$ with all $\boldsymbol{z}_{tot_k}$, and setting $N_{MC} = N_{tot}$ so that the Monte Carlo estimation is performed on locations $\boldsymbol{z}_{tot}$, Equation (A.7) yields:

$$\begin{aligned} &\frac{1}{N_{tot}} K_{tot}\Phi \approx \Lambda\Phi \\ \text{with} \quad &\frac{1}{N_{tot}} \Phi_{\cdot i}^{\intercal}\Phi_{\cdot j} \approx \delta_{ij}. \end{aligned} \tag{A.8}$$

Matching (A.8) against (A.1) and (A.2) permits to derive:

$$\begin{aligned} \Phi &\approx \sqrt{N_{tot}} U_{tot}, \\ \Lambda &\approx \frac{\Lambda_{tot}}{N_{tot}}, \end{aligned}$$

which gives the Nyström approximation of the j-th eigenfunction from Eq. (A.7):

$$\phi_j(\boldsymbol{z}) \approx \frac{\sqrt{N_{tot}}}{\Lambda_{tot_{jj}}} \sum_{k=1}^{N_{tot}} k(\boldsymbol{z}, \boldsymbol{z}_{tot_k}) U_{tot_{kj}}. \tag{A.9}$$

Using these approximations, KLE-based realisations write:

$$\begin{aligned} \boldsymbol{q}_{KLE}^{(i)}(\boldsymbol{z}) &= \sum_{j=1}^{N_{tot}} \left(\sqrt{\frac{\Lambda_{tot_{jj}}}{N_{tot}}} \frac{\sqrt{N_{tot}}}{\Lambda_{tot_{jj}}} \sum_{k=1}^{N_{tot}} \left(k(\boldsymbol{z}, \boldsymbol{z}_{tot_k}) U_{tot_{kj}} \right) \theta_{tot_j}^{(i)} \right) \\ &= \sum_{j=1}^{N_{tot}} \left(\frac{\theta_{tot_j}^{(i)}}{\sqrt{\Lambda_{tot_{jj}}}} \sum_{k=1}^{N_{tot}} \left(k(\boldsymbol{z}, \boldsymbol{z}_{tot_k}) U_{tot_{kj}} \right) \right). \end{aligned} \tag{A.10}$$

Inducing points. We proposed to draw realisations using Equation (18). To simplify the following, we assume here that $N_{train} = 0$ and $\boldsymbol{q}_{tot} = \boldsymbol{q}_{ind}$. This implies that matrix L_{ind} corresponds to the square root matrix $K_{tot}^{1/2}$.

By definition, in this context, $\boldsymbol{q}_{tot}^{(i)} = K_{tot}^{1/2} \boldsymbol{\theta}_{tot}^{(i)}$, with $\boldsymbol{\theta}_{tot}^{(i)} \sim \mathcal{N}(\boldsymbol{0}, I)$. A realisation at location $\boldsymbol{z}$ then writes:

$$\begin{aligned} \boldsymbol{q}^{(i)}(\boldsymbol{z}) &= k(\boldsymbol{z}, \boldsymbol{z}_{tot}) K_{tot}^{-1} \boldsymbol{q}_{tot}^{(i)} \\ &= k(\boldsymbol{z}, \boldsymbol{z}_{tot}) K_{tot}^{-1/2} \boldsymbol{\theta}_{tot}^{(i)} \\ &= k(\boldsymbol{z}, \boldsymbol{z}_{tot}) U_{tot} \Lambda_{tot}^{-1/2} \boldsymbol{\theta}_{tot}^{(i)} \\ &= \sum_{j=1}^{N_{tot}} \left(\frac{\theta_{tot_j}^{(i)}}{\sqrt{\Lambda_{tot_{jj}}}} \sum_{k=1}^{N_{tot}} \left(k(\boldsymbol{z}, \boldsymbol{z}_{tot_k}) U_{tot_{kj}} \right) \right), \end{aligned} \tag{A.11}$$

which corresponds exactly to Equation (A.10).

Note that in the case $\boldsymbol{q}_{tot} \neq \boldsymbol{q}_{ind}$, which permits to condition the realisations on some training data, the same equivalence can be shown by comparing (18) to a conditional Karhunen-Loève model [64].

References

[1] V. Keshavarzzadeh, R. G. Ghanem, D. A. Tortorelli, Shape optimization under uncertainty for rotor blades of horizontal axis wind turbines, Computer Methods in Applied Mechanics and Engineering 354 (2019) 271–306.

[2] P. Seshadri, P. Constantine, G. Iaccarino, G. Parks, A density-matching approach for optimization under uncertainty, Computer Methods in Applied Mechanics and Engineering 305 (2016) 562–578.

[3] J. Zhang, M. Xiao, P. Li, L. Gao, Quantile-based topology optimization under uncertainty using kriging metamodel, Computer Methods in Applied Mechanics and Engineering 393 (2022) 114690.

[4] V. Keshavarzzadeh, R. M. Kirby, A. Narayan, Stress-based topology optimization under uncertainty via simulation-based gaussian process, Computer Methods in Applied Mechanics and Engineering 365 (2020) 112992.

[5] X. Liu, T. Li, Z. Zhou, L. Hu, An efficient multi-objective reliability-based design optimization method for structure based on probability and interval hybrid model, Computer Methods in Applied Mechanics and Engineering 392 (2022) 114682.

[6] C. Comis Da Ronco, R. Ponza, E. Benini, Aerodynamic shape optimization of aircraft components using an advanced multi-objective evolutionary approach, Computer Methods in Applied Mechanics and Engineering 285 (2015) 255–290.

[7] E. Zitzler, D. Kalyanmoy, L. Thiele, Comparison of multiobjective evolutionary algorithms: Empirical results, Evolutionary computation 8 (2) (2000) 173–195.

[8] H. Ishibuchi, N. Tsukamoto, Y. Nojima, Evolutionary many-objective optimization: A short review, in: 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), IEEE, 2008, pp. 2419–2426.

[9] X.-S. Yang, Engineering optimization: an introduction with metaheuristic applications, John Wiley & Sons, 2010.

[10] Y. Jin, J. Branke, et al., Evolutionary optimization in uncertain environments-a survey, IEEE Transactions on evolutionary computation 9 (3) (2005) 303–317.

[11] C.-K. Goh, K. C. Tan, Evolutionary multi-objective optimization in uncertain environments, Issues and Algorithms, Studies in Computational Intelligence 186 (2009) 5–18.

[12] K. C. Tan, C. K. Goh, Handling uncertainties in evolutionary multi-objective optimization, in: IEEE World Congress on Computational Intelligence, Springer, 2008, pp. 262–292.

[13] C. P. Ho, P. Parpas, Multilevel optimization methods: Convergence and problem structure (2016).

[14] M. El-Beltagy, A. Keane, Optimisation for multilevel problems: A comparison of various algorithms, in: Adaptive computing in design and manufacture, Springer, 1998, pp. 111–120.

[15] A. March, K. Willcox, Provably convergent multifidelity optimization algorithm not requiring high-fidelity derivatives, AIAA journal 50 (5) (2012) 1079–1089.

[16] M. K. Zahir, Z. Gao, Variable-fidelity optimization with design space reduction, Chinese Journal of Aeronautics 26 (4) (2013) 841–849.

[17] V. Picheny, D. Ginsbourger, Y. Richet, Noisy expected improvement and on-line computation time allocation for the optimization of simulators with tunable fidelity (2010).

[18] J. Teich, Pareto-front exploration with uncertain objectives, in: International Conference on Evolutionary Multi-Criterion Optimization, Springer, 2001, pp. 314–328.

[19] E. J. Hughes, Evolutionary multi-objective ranking with uncertainty and noise, in: International Conference on Evolutionary Multi-Criterion Optimization, Springer, 2001, pp. 329–343.

[20] P. Limbourg, D. E. S. Aponte, An optimization algorithm for imprecise multi-objective problem functions, in: 2005 IEEE Congress on Evolutionary Computation, Vol. 1, IEEE, 2005, pp. 459–466.

[21] H. Eskandari, C. D. Geiger, R. Bird, Handling uncertainty in evolutionary multiobjective optimization: Spga, in: 2007 IEEE Congress on Evolutionary Computation, IEEE, 2007, pp. 4130–4137.

[22] D.-w. Gong, N.-n. Qin, X.-y. Sun, Evolutionary algorithms for multi-objective optimization problems with interval parameters, in: 2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA), IEEE, 2010, pp. 411–420.

[23] G. L. Soares, F. G. Guimarães, C. A. Maia, J. A. Vasconcelos, L. Jaulin, Interval robust multi-objective evolutionary algorithm, in: 2009 IEEE Congress on Evolutionary Computation, IEEE, 2009, pp. 1637–1643.

[24] M. Li, R. Silva, F. Guimarães, D. Lowther, A new robust dominance criterion for multiobjective optimization, IEEE Transactions on Magnetics 51 (3) (2015) 1–4.

[25] F. Khosravi, M. Müller, M. Glaß, J. Teich, Uncertainty-aware reliability analysis and optimization, in: 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE), IEEE, 2015, pp. 97–102.

[26] F. Khosravi, M. Borst, J. Teich, Probabilistic dominance in robust multi-objective optimization, in: 2018 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2018, pp. 1–6.

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

[28] F. Khosravi, A. Rass, J. Teich, Efficient computation of probabilistic dominance in multi-objective optimization, ACM Trans. Evol. Learn. Optim. 1 (4) (oct 2021). doi:10.1145/3469801.
URL https://doi.org/10.1145/3469801

[29] M. Mlakar, T. Tušar, B. Filipič, Comparing solutions under uncertainty in multiobjective optimization, Mathematical Problems in Engineering 2014 (2014).

[30] F. Fusi, P. M. Congedo, An adaptive strategy on the error of the objective functions for uncertainty-based derivative-free optimization, Journal of Computational Physics 309 (2016) 241–266.

[31] M. Rivier, P. Congedo, Surrogate-assisted bounding-box approach applied to constrained multi-objective optimisation under uncertainty, Reliability Engineering System Safety 217 (2022) 108039. doi:https://doi.org/10.1016/j.ress.2021.108039.
URL https://www.sciencedirect.com/science/article/pii/S0951832021005445

[32] S. Da Veiga, F. Delbos, Robust optimization for expensive simulators with surrogate models: Application to well placement for oil recovery, Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures 11 (2013) 3321–3328.

[33] V. Baudoui, Optimisation robuste multiobjectifs par modèles de substitution, Ph.D. thesis, Toulouse, ISAE (2012).

[34] J. Janusevskis, R. Le Riche, Simultaneous kriging-based estimation and optimization of mean response, Journal of Global Optimization 55 (2) (2013) 313–336.

[35] B. J. Williams, T. J. Santner, W. I. Notz, Sequential design of computer experiments to minimize integrated response functions, Statistica Sinica (2000) 1133–1152.

[36] R. F. Coelho, Probabilistic dominance in multiobjective reliability-based optimization: Theory and implementation, IEEE Transactions on evolutionary computation 19 (2) (2014) 214–224.

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

[38] C. E. Rasmussen, C. K. I. Williams, Gaussian processes for machine learning., Adaptive computation and machine learning, MIT Press, 2006.

[39] C. E. Powell, et al., Generating realisations of stationary gaussian random fields by circulant embedding, matrix 2 (2) (2014) 1.

[40] M. Pourahmadi, Covariance estimation: The glm and regularization perspectives, Statistical Science (2011) 369–387.

[41] C. Lantuéjoul, N. Desassis, Simulation of a gaussian random vector: a propagative version of the gibbs sampler, in: The 9th international geostatistics congress, 2012, pp. 174–181.

[42] A. M. Panunzio, R. Cottereau, G. Puel, Large scale random fields generation using localized karhunen–loève expansion, Advanced Modeling and Simulation in Engineering Sciences 5 (1) (2018) 20.

[43] M. Herdin, N. Czink, H. Ozcelik, E. Bonek, Correlation matrix distance, a meaningful measure for evaluation of non-stationary mimo channels, in: 2005 IEEE 61st Vehicular Technology Conference, Vol. 1, IEEE, 2005, pp. 136–140.

[44] M. Rivier, P. M. Congedo, Surrogate-assisted bounding-box approach for optimization problems with tunable objectives fidelity, Journal of Global Optimization 75 (4) (2019) 1079–1109.

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

[46] M. Pini, G. Persico, D. Pasquale, S. Rebay, Adjoint method for shape optimization in real-gas flow applications, ASME Journal of Engineering for Gas Turbines and Power 137 (3) (2015).

[47] S. Vitale, T. A. Albring, M. Pini, N. R. Gauger, P. Colonna, Fully turbulent discrete adjoint solver for non-ideal compressible flow applications, Journal of the Global Power and Propulsion Society 1 (2017) Z1FVOI.

[48] G. Persico, P. Rodriguez-Fernandez, A. Romei, High-fidelity shape-optimization of non-conventional turbomachinery by surrogate evolutionary strategies, Journal of Turbomachinery 141 (8) (2019) 081010.

[49] M. Pini, G. Persico, V. Dossena, Robust adjoint-based shape optimization of supersonic turbomachinery cascades, in: ASME Turbo Expo 2014: Turbine Technical Conference and Exposition, American Society of Mechanical Engineers, 2014, pp. V02BT39A043–V02BT39A043.

[50] N. Razaaly, G. Gori, G. Iaccarino, P. Congedo, Optimization of an orc supersonic nozzle under epistemic uncertainties due to turbulence models, in: GPPS 2019, 2019.

[51] N. Razaaly, G. Persico, P. M. Congedo, Impact of geometric, operational, and model uncertainties on the non-ideal flow through a supersonic orc turbine cascade, Energy 169 (2019) 213–227.

[52] M. Pini, G. Persico, V. Dossena, Robust adjoint-based shape optimization of supersonic turbomachinery cascades, in: ASME Turbo Expo 2014: Turbine Technical Conference and Exposition, American Society of Mechanical Engineers, 2014, pp. V02BT39A043–V02BT39A043.

[53] P. Congedo, G. Geraci, R. Abgrall, V. Pediroda, L. Parussini, Tsi metamodels-based multi-objective robust optimization, Engineering Computations (Swansea, Wales) 30 (8) (2013) 1032–1053.

[54] G. Farin, Curves and Surfaces for CAGD: A Practical Guide, 5th Edition, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[55] F. Palacios, M. F. Colonno, A. C. Aranake, A. Campos, S. R. Copeland, T. D. Economon, A. K. Lonkar, T. W. Lukaczyk, T. W. R. Taylor, J. J. Alonso, Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design, in: 51st AIAA Aerospace Sciences Meeting and Exhibit, 2013.

[56] T. D. Economon, D. Mudigere, G. Bansal, A. Heinecke, F. Palacios, J. Park, M. Smelyanskiy, J. J. Alonso, P. Dubey, Performance optimizations for scalable implicit {RANS} calculations with {SU2}, Computers & Fluids 129 (2016) 146 – 158. doi:http://dx.doi.org/10.1016/j.compfluid.2016.02.003.
URL http://www.sciencedirect.com/science/article/pii/S0045793016300214

[57] M. Pini, S. Vitale, , P. Colonna, G. Gori, A. Guardone, T. Economon, J. Alonso, F. Palacios, Su2: the open-source software for non-ideal compressible flows, in: NICFD 2016: 1st International Seminar on Non-Ideal Compressible-Fluid Dynamics for Propusion & Power, Varenna, Italy, 2016.

[58] S. Vitale, G. Gori, M. Pini, A. Guardone, T. D. Economon, F. Palacios, J. J. Alonso, P. Colonna, Extension of the SU2 open source CFD code to the simulation of turbulent flows of fluids modelled with complex thermophysical laws, no. AIAA Paper 2015-2760, 2015.

[59] N. Razaaly, Rare event estimation and robust optimization methods with applications to orc turbine cascade, Ph.D. thesis, Université Paris-Saclay (2019).

[60] J. R. Gardner, M. J. Kusner, Z. E. Xu, K. Q. Weinberger, J. P. Cunningham, Bayesian optimization with inequality constraints., in: ICML, 2014, pp. 937–945.

[61] B. Echard, N. Gayton, M. Lemaire, Ak-mcs: an active learning reliability method combining kriging and monte carlo simulation, Structural Safety 33 (2) (2011) 145–154.

[62] N. Razaaly, P. M. Congedo, Novel algorithm using active metamodel learning and importance sampling: application to multiple failure regions of low probability, Journal of Computational Physics 368 (2018) 92–114.

[63] C. K. Williams, M. Seeger, Using the nyström method to speed up kernel machines, in: Advances in neural information processing systems, 2001, pp. 682–688.

[64] R. Tipireddy, D. A. Barajas-Solano, A. M. Tartakovsky, Conditional karhunen-lo\eve expansion for uncertainty quantification and active learning in partial differential equation models, arXiv preprint arXiv:1904.08069 (2019).