

SMAI-JCM

SMAI JOURNAL OF
COMPUTATIONAL MATHEMATICS

Active learning strategies for the
estimation of a feasible set defined
from a vector output black-box
simulator

CLÉMENT DUHAMEL, CÉLINE HELBERT, MIGUEL MUNOZ ZUNIGA,
CLÉMENTINE PRIEUR & DELPHINE SINOQUET

Volume 12 (2026), p. 331-369.

<https://doi.org/10.5802/smai-jcm.151>

© The authors, 2026.



*The SMAI Journal of Computational Mathematics is a member
of the Centre Mersenne for Open Scientific Publishing*

<http://www.centre-mersenne.org/>

Submissions at <https://smai-jcm.centre-mersenne.org/ojs/submission>

e-ISSN: 2426-8399



Active learning strategies for the estimation of a feasible set defined from a vector output black-box simulator

CLÉMENT DUHAMEL ¹
CÉLINE HELBERT ²
MIGUEL MUNOZ ZUNIGA ³
CLÉMENTINE PRIEUR ⁴
DELPHINE SINOQUET ⁵

¹ Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France

E-mail address: clement.duhamel@univ-grenoble-alpes.fr

² Centrale Lyon, CNRS, INSA Lyon, Université Claude Bernard Lyon 1, Université Jean Monnet, ICJ UMR5208, Ecully, France.

³ IFP Energies Nouvelles, 92852 Rueil-Malmaison, France

⁴ Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France

⁵ IFP Energies Nouvelles, 92852 Rueil-Malmaison, France.

Abstract. Many industrial problems require the use of black-box numerical simulators, for which it is essential to determine the set of so-called feasible input parameters. A set of parameters is feasible if the output of the code on these parameters satisfies given constraints, for example, by remaining below a certain threshold. Active learning is an effective approach to solve this type of problem, by sequentially enriching a design of experiments using a well-chosen acquisition criterion, based here on a Gaussian process surrogate model. In this work, we focus specifically on simulators with vector outputs. We propose several enrichment strategies to simultaneously explore the entire collection of feasible sets associated with each output component. These enrichment strategies are first tested and compared on analytical test functions, before being applied to the pre-calibration of a simulator dedicated to wind turbine design. The aim is to identify input parameter configurations that respect the vibration constraints imposed on the simulator outputs. Numerical results demonstrate the efficiency of the three proposed strategies, which are compared against two baseline strategies (random sampling and Sobol' sequence).

2020 Mathematics Subject Classification. 60G60, 60G15, 62P30.

Keywords. Gaussian process regression, excursion set estimation, multi-output black-box functions, active learning, sequential design of experiments.

1. Introduction

Estimation of feasible sets is a key issue in many industrial fields, particularly for the optimal design of complex systems. When optimal design is done through the use of a numerical simulator, identifying a feasible set comes down to searching input parameters that lead to outputs which respect a given constraint, such as not exceeding a certain threshold. This type of problem is met, for example, for the tuning of a vehicle pollution control system [23]. Similar techniques are also applied in environmental monitoring, such as optimizing the trajectory of an autonomous underwater vehicle for oceanographic data collection, as explored in [26].

Mathematically, in the following, a feasible set, also called an excursion set, is defined by

$$\Gamma^* := \left\{ \mathbf{x} \in \mathbb{X}, g(\mathbf{x}) \leq T \right\}, \quad (1.1)$$

where $g : \mathbb{X} \rightarrow \mathbb{R}$ is a costly to evaluate black-box model, $\mathbb{X}$ the space of inputs (called design space) and T a given threshold. To best estimate the set Γ^* , function g is approximated from a limited number

of evaluations obtained by an initial design of experiments (DoE) which is then relevantly enriched in areas of interest. The aim is to extend the design progressively and minimize the number of evaluations of g necessary to discover the targeted set Γ^* (see for example [6, 32]). In this active learning approach, new points are iteratively selected through the optimization of an acquisition criterion that leverages information provided by a computationally efficient surrogate model of g . Among the substitution models, we use Gaussian process regression (GPR) [53], which models g as a realization of a Gaussian process (GP). This model is popular because it offers both prediction and associated prediction error. Acquisition criteria for estimating an excursion set include classical criteria, such as Bichon criterion [8], which aims to estimate the boundary of the set Γ^* .

This work focuses on vector black-box models, which provide, for each set of input parameters, several quantities of interest produced simultaneously (isotopic data). In this framework, the vector black-box model is denoted $\mathbf{g} := (g_1, \dots, g_p)^\top : \mathbb{X} \rightarrow \mathbb{R}^p$, and for a vector of thresholds $\mathbf{T} := (T_1, \dots, T_p)^\top \in \mathbb{R}^p$, the partial excursion sets are defined for $i = 1, \dots, p$ by

$$\Gamma_i^* := \{\mathbf{x} \in \mathbb{X}, g_i(\mathbf{x}) \leq T_i\}, \quad (1.2)$$

and the global excursion set Γ^* is defined as the intersection of all partial excursion sets

$$\Gamma^* := \{\mathbf{x} \in \mathbb{X}, \mathbf{g}(\mathbf{x}) \leq \mathbf{T}\} = \bigcap_{i=1}^p \Gamma_i^*, \quad (1.3)$$

where the symbol $\leq$ between two vectors means that each component of the first vector is less than or equal to the corresponding component of the second vector. Equations (1.2) and (1.3) correspond to two different vector generalizations of the initial problem (Eq. (1.1)).

In literature, existing works target the global excursion set Γ^* . For example, [26] focuses on the estimation of the global excursion set Γ^* using extensions of the stepwise uncertainty reduction (SUR) criteria, such as the reduction of the excursion measure variance (EMV) and the integrated Bernoulli variance (IBV) [6]. These criteria are based on a substitution model defined as a multi-output Gaussian process (see for example [41]) and a generalization of the coverage probability. The advantage of using this model (see [49] for Bayesian optimization context) lies in its ability to take into account correlations between simulator outputs, given that in practice, the different outputs of a numerical simulator are rarely independent. Here in this work we focus on the estimation of each partial excursion set Γ_i^* which differs from that of the global set, as it requires the discovery of the boundaries of each partial set rather than those of the global set. To this end, three enrichment strategies are proposed, all related to Bichon criterion. The first two use scalar criteria with independent substitution models, while the third one relies on a multi-output Gaussian process, taking the correlation between outputs into account. These three strategies are compared against two baseline methods: random sampling and Sobol' sequence-based sampling. These strategies are first evaluated on analytical test cases, then applied to a real problem.

The application that motivates our work, concerns the calibration of a numerical simulator for wind turbine design. This simulator analyzes the vibration frequencies and deformation modes of a wind turbine structure at standstill, in response to wind loads. The aim is to determine the values of the input parameters of the simulator, so that the vibration frequencies $\lambda_i(\Theta)$ and deformation modes $\text{Mod}_i(\Theta)$ are sufficiently close to the reference frequencies λ_i^* and modes Mod_i^* based on experimental data. The specification for this application is the simultaneous estimation of partial excursion sets for several output components. For this application, all output components are evaluated simultaneously for the same input point; such a setting is called isotopic setting.

The rest of this article is structured in four main sections. Section 2 presents the framework for estimating excursion sets for scalar models, recalling the use of Gaussian process regression. It also discusses the main challenges encountered for vector output modeling. Section 3 introduces three experimental design enrichment strategies inspired by Bichon criterion, adapted to the simultaneous

estimation of partial excursion sets for each model output component. These strategies are compared against two baseline methods (random sampling and Sobol' sequence-based sampling) on analytical examples in Section 4. Finally, in Section 5, these methods are applied to the pre-calibration of a wind turbine simulator, in order to identify the input parameters guaranteeing the correspondence of simulated frequencies and modes with experimental references.

2. General framework

2.1. Reminders on excursion set estimation for scalar models

2.1.1. Sequential design of experiments and surrogate model

Let $\mathbb{X} \in \mathbb{R}^d$ a compact set, $g : \mathbb{X} \rightarrow \mathbb{R}$ a black-box function and T a given threshold. The objective is to estimate accurately the excursion set defined as

$$\Gamma^* := \{\mathbf{x} \in \mathbb{X}, g(\mathbf{x}) \leq T\}, \quad (2.1)$$

with a limited number of costly evaluations of g .

A sequential enrichment strategy is used to build the experimental design, consisting of points where the function g is evaluated. This active learning approach is well known in the context of black-box optimization. It relies on a substitution model, built from the available number of evaluations of g , to select new points to evaluate at each iteration. The main steps involved in building a sequential DoE are described below.

The first step is to define an initial space filling DoE with few points well spread in the input space. The Latin hypercube sampling (LHS) method introduced by [43] and various improvements of it (see [15, 20]) can be used for this first experimental design. A substitution model, less costly to evaluate than function g , is then built from the evaluations of g at the points of this initial design.

An acquisition criterion (also known as an enrichment criterion), based on the substitution model, is optimized to determine a new point to be added to the DoE. The function g is then evaluated on that point and the substitution model is updated accordingly. We then alternate optimization of the acquisition criterion and update of the substitution model, until a stopping criterion is verified. This may correspond, for example, to a predefined simulation budget or to a residual uncertainty threshold calculated according to the target objective. Figure 2.1 provides a synthetic scheme of this sequential strategy.

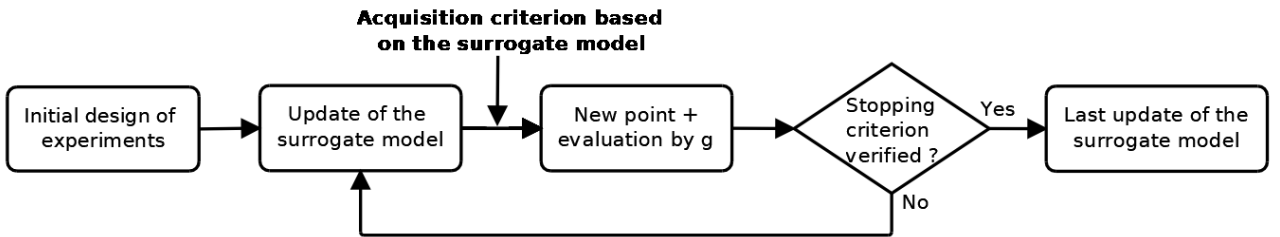


FIGURE 2.1. Construction of a sequential DoE.

In the sequential strategy for enriching a DoE, the substitution model plays an essential role. In this article, we focus on Gaussian process regression (kriging), whose basics are reviewed in the following subsection. The main advantage of Gaussian process regression lies in its ability to provide not only a prediction, but also an estimation of the error associated with that prediction. Other substitution models, such as polynomials chaos (e.g., [62]), regression trees (e.g., [27]) or neural networks (e.g., [7]), can also be mentioned.

2.1.2. Gaussian process regression

Gaussian Process Regression (GPR) is based on the concept that the black-box function g can be viewed as a realization of a Gaussian process (GP) ξ , expressed as:

$$\xi(\mathbf{x}) := m(\mathbf{x}) + \mathbf{Z}(\mathbf{x}), \quad \forall \mathbf{x} \in \mathbb{X} \quad (2.2)$$

with m the trend of ξ (deterministic part) and $\mathbf{Z}$ a centered GP with covariance kernel $k : \mathbb{X}^2 \rightarrow \mathbb{R}$ (stochastic part).

The choice of the covariance kernel k is crucial, since it determines the regularity of the predictor (see [48]). Typically, the covariance kernel is selected from a family of stationary kernels parameterized by one or more hyperparameters. These hyperparameters are subsequently estimated using methods such as maximum likelihood (e.g., [29, 33, 35]) or cross-validation (e.g., [35]). In the following, we will employ a Matérn 5/2 kernel class (producing trajectories of class $\mathcal{C}^2$), with hyperparameters estimated via maximum likelihood (see Section 3.3).

To limit the difficulty of estimating the mean function m , it is common practice to set m as a linear combination of given basis functions $\mathbf{f} := (f_i)_{i=1}^l$ with coefficients $\boldsymbol{\beta} := \{\beta_i\}_{i=1}^l$ to be estimated. Let us define the following notations:

- $\chi_n := (\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)})$ represents the initial DoE;
 - $g(\chi_n)$ and $\xi(\chi_n)$ are the vectors of evaluations of g and ξ on χ_n ;
 - $\mathcal{E}_n$ is the event $\xi(\chi_n) = g(\chi_n)$;
 - $K := (k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}))_{1 \leq i, j \leq n}$ denotes the covariance matrix of $\xi(\chi_n)$;
 - $\mathbf{k}(\mathbf{x})$ is the cross-covariance matrix between $\xi(\chi_n)$ and $\xi(\mathbf{x})$, defined as
- $$\mathbf{k}(\mathbf{x}) := (k(\mathbf{x}, \mathbf{x}^{(1)}), \dots, k(\mathbf{x}, \mathbf{x}^{(n)}))^\top, \quad \forall \mathbf{x} \in \mathbb{X};$$
- $\mathbf{f}(\mathbf{x}) := (f_1(\mathbf{x}), \dots, f_l(\mathbf{x}))^\top$ represents the evaluation vector of $\mathbf{f}$ on $\mathbf{x}$;
 - $\mathbb{F} \in \mathbb{R}^{n \times l}$ is the matrix whose i^{th} row is $\mathbf{f}(\mathbf{x}^{(i)})^\top$.

When $\boldsymbol{\beta}$ is known, the conditioned process $\xi | \mathcal{E}_n$ remains Gaussian [46]. Expressions of posterior mean, variance, and covariance can be found in [19], using the same notation. These formulas align with those of the Best Linear Unbiased Predictor (BLUP) with respect to the mean squared error.

When $\boldsymbol{\beta}$ is unknown, its maximum likelihood estimator (MLE) is given by

$$\hat{\boldsymbol{\beta}} := (\mathbb{F}^\top K^{-1} \mathbb{F})^{-1} \mathbb{F}^\top K^{-1} g(\chi_n).$$

The formulas for the best linear unbiased predictor, as well as the corresponding mean squared error and covariance error functions, are given by:

$$m_n(\mathbf{x}) = \mathbf{f}(\mathbf{x})^\top \hat{\boldsymbol{\beta}} + \mathbf{k}(\mathbf{x})^\top K^{-1} (g(\chi_n) - \mathbb{F} \hat{\boldsymbol{\beta}}), \quad (2.3)$$

$$\sigma_n^2(\mathbf{x}) = \sigma^2(\mathbf{x}) - \mathbf{k}(\mathbf{x})^\top K^{-1} \mathbf{k}(\mathbf{x}) + (\mathbf{f}(\mathbf{x})^\top - \mathbf{k}(\mathbf{x})^\top K^{-1} \mathbb{F}) (\mathbb{F}^\top K^{-1} \mathbb{F})^{-1} (\mathbf{f}(\mathbf{x})^\top - \mathbf{k}(\mathbf{x})^\top K^{-1} \mathbb{F})^\top, \quad (2.4)$$

$$k_n(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') - \mathbf{k}(\mathbf{x})^\top K^{-1} \mathbf{k}(\mathbf{x}') + (\mathbf{f}(\mathbf{x})^\top - \mathbf{k}(\mathbf{x})^\top K^{-1} \mathbb{F}) (\mathbb{F}^\top K^{-1} \mathbb{F})^{-1} (\mathbf{f}(\mathbf{x}')^\top - \mathbf{k}(\mathbf{x}')^\top K^{-1} \mathbb{F})^\top. \quad (2.5)$$

Within a Bayesian framework with non-informative prior on parameter $\boldsymbol{\beta}$, quantities in (2.3), (2.4) and (2.5) can still be interpreted as conditional expectation, variance and covariance [37].

2.1.3. Scalar Bichon criterion

The choice of the acquisition criterion is also crucial in the framework of Figure 2.1, and depends on both the employed substitution model and the final objective: global exploration, optimization, excursion set estimation, etc. Among the multiple criteria for sequential enrichment of a DoE (see for example [6, 22, 39, 40, 50]), Bichon criterion [8] is specifically tailored to excursion set estimation. It is built from the Expected Improvement (EI) criterion from [40], originally developed for global optimization. This adaptation considers both the variability of the surrogate model and the potential improvement in the characterization of the excursion set boundary.

The new points selected based on Bichon criterion are those that maximize the Expected Feasibility Function (EFF), i.e.,

$$\mathbf{x}^{(n+1)} \in \operatorname{argmax}_{\mathbf{x} \in \mathbb{X}} \text{EFF}(\mathbf{x}), \quad (2.6)$$

where EFF is defined by

$$\text{EFF}(\mathbf{x}) := \mathbb{E} \left[(\kappa \sigma_n(\mathbf{x}) - |T - \xi(\mathbf{x})|)^+ \mid \mathcal{E}_n \right], \quad (2.7)$$

with $(\cdot)^+ := \max(\cdot, 0)$ and κ a fixed parameter. The influence of the proportionality coefficient κ is small [8] and can be set to 1. The function EFF represents, on average, the distance of the surrogate model to the limits of the interval $[T - \kappa \sigma_n(\mathbf{x}), T + \kappa \sigma_n(\mathbf{x})]$ only if the surrogate model belongs to this interval and vanishes otherwise. Maximizing the criterion in Equation (2.6) focuses on selecting points either near the estimated boundary of the excursion set (approximated via GPR), or associated with large values of $\sigma_n(\mathbf{x})$, i.e. a high model uncertainty. This approach balances exploitation and exploration of the design space. A statistical interpretation of Bichon criterion has been proposed in [19] establishing a connection with statistical test theory [14].

2.2. Main challenges for models with vector output

2.2.1. Excursion set in the case of several outputs

Vector-valued black-box models are widely used in industry to optimize and calibrate complex systems. They provide several critical quantities of interest as outputs, such as mechanical stress and acoustic response in automotive, or mechanical stress and production cost in industrial manufacturing. These models can be leveraged for improving system performance and efficiency, after a calibration step on experimental data. It is sometimes crucial to check, for a given set of input parameters, which constraints on the outputs are respected, to ensure that the system meets desired specifications or performance levels.

In the framework under study, simulator data are assumed to be isotopic, meaning that all output components of the simulator are evaluated simultaneously. This property has to be taken into account when selecting a new point for the design. In this context, the vector-valued black-box model is denoted by $\mathbf{g} := (g_1, \dots, g_p)^\top$, and the partial and global excursion sets are defined in Equations (1.2) and (1.3).

In this work, we aim to simultaneously estimate each of the partial excursion sets Γ_i^* . As already mentioned in the introduction, this problem is quite different from that of estimating the global excursion set [26], since it requires discovering each of the boundaries of partial excursion sets. To illustrate this difference, we consider the following 2D function where the first component is the Branin function

$$g_1(\mathbf{x}) := \left(\bar{\mathbf{x}}_2 - \frac{5\bar{\mathbf{x}}_1^2}{4\pi^2} + \frac{5\bar{\mathbf{x}}_1}{\pi} - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos(\bar{\mathbf{x}}_1) + 10 \quad (2.8)$$

and the second component is a modified version of the Branin function (see [41])

$$g_2(\mathbf{x}) := \left(\bar{\mathbf{x}}_2 - \frac{3\bar{\mathbf{x}}_1^2}{4\pi^2} + \frac{4\bar{\mathbf{x}}_1}{\pi} - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos(\bar{\mathbf{x}}_1) + 2\bar{\mathbf{x}}_1 - 9\bar{\mathbf{x}}_2 + 32 \quad (2.9)$$

with $\bar{\mathbf{x}}_1 := 15\mathbf{x}_1 - 5$ and $\bar{\mathbf{x}}_2 := 15\mathbf{x}_2$. Figure 2.2 shows for this function the level sets for each component (left and middle), and global excursion set (right) for $T_1 = T_2 = 10$.

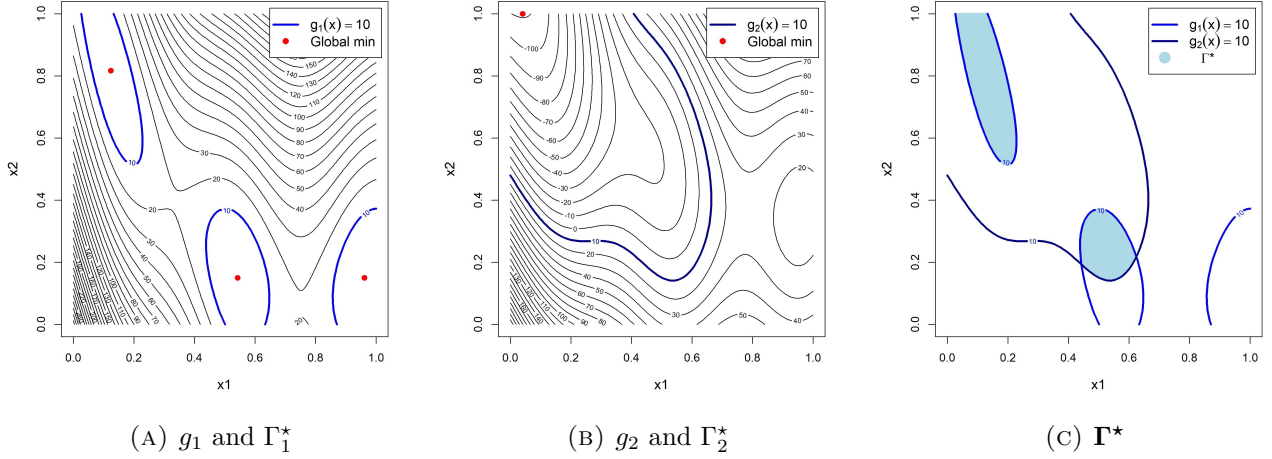


FIGURE 2.2. (A) Level sets for the first component of the 2D function and boundaries of the partial excursion set Γ_1^* for threshold $T_1 = 10$. (B) Level sets for the second component and boundaries of the partial excursion set Γ_2^* for threshold $T_2 = 10$. (C) Comparison of these partial excursion sets with the global excursion set Γ^* .

To the best of our knowledge, no existing criterion is specifically designed for the simultaneous estimation of each partial excursion set Γ_i^* . We propose three novel enrichment strategies based on Bichon criterion to address this objective. The first two strategies rely on scalar criteria built on independent surrogate models. The third strategy introduces an innovative criterion utilizing a multi-output Gaussian process, exploiting the correlations between the different output components. To assess performances of these three strategies, they are also compared to two baseline enrichment methods: random sampling and Sobol' sequence-based sampling. Before detailing the proposed strategies, we introduce multi-output Gaussian processes.

2.2.2. Multi-output gaussian processes

Gaussian process regression for vector-valued functions, also known as multi-output Gaussian process (MOGP) regression is a generalization of Gaussian process regression to build a surrogate of a deterministic vector-valued function $\mathbf{g} : \mathbb{X} \subset \mathbb{R}^d \rightarrow \mathbb{R}^p$. The use of MOGPs is based on the fundamental belief that the output components are correlated in some manner. Therefore, a crucial aspect of MOGPs is to take advantage of correlations between output components to mutualize the information acquired on each output and obtain more accurate predictions than if the output components were modeled independently. In the following, we are interested in a framework where the evaluation points are the same for each component of model output (isotopic data). This often occurs when the

p output responses at a point $\mathbf{x}$ can be obtained through a single simulation (see for example [41] for more details on MOGPs).

Let us assume that $\mathbf{g} := (g_1, \dots, g_p)^\top$ is the realization of a vector-valued GP $\boldsymbol{\xi} := (\xi_1, \dots, \xi_p)^\top$. This process is characterized by its mean $M := (M_1, \dots, M_p) : \mathbb{R}^d \rightarrow \mathbb{R}^p$ and covariance $K := (K_{i,j})_{1 \leq i,j \leq p} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathcal{S}_p^+(\mathbb{R})$ functions defined for $(i, j) \in \{1, \dots, p\}^2$ by

$$M_i(\mathbf{x}) := \mathbb{E}(\xi_i(\mathbf{x})) \quad \text{and} \quad K_{i,j}(\mathbf{x}, \mathbf{x}') := \text{Cov}(\xi_i(\mathbf{x}), \xi_j(\mathbf{x}')), \quad (2.10)$$

with $\mathcal{S}_p^+(\mathbb{R})$ denoting real symmetric positive semi-definite matrices of size p . We note $\Sigma(\mathbf{x}) := K(\mathbf{x}, \mathbf{x})$ the covariance matrix of $\boldsymbol{\xi}(\mathbf{x})$. Let $\chi_n := (\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)})^\top$ denote the DoE at step n . We define

$$\boldsymbol{\xi}(\chi_n) := (\xi_1(\mathbf{x}^{(1)}), \dots, \xi_1(\mathbf{x}^{(n)}), \dots, \xi_p(\mathbf{x}^{(1)}), \dots, \xi_p(\mathbf{x}^{(n)}))^\top \in \mathbb{R}^{pn \times 1}, \quad (2.11)$$

$$\mathbf{g}(\chi_n) := (g_1(\mathbf{x}^{(1)}), \dots, g_1(\mathbf{x}^{(n)}), \dots, g_p(\mathbf{x}^{(1)}), \dots, g_p(\mathbf{x}^{(n)}))^\top \in \mathbb{R}^{pn \times 1}, \quad (2.12)$$

and $\mathcal{E}_n$ the event $\boldsymbol{\xi}(\chi_n) = \mathbf{g}(\chi_n)$. The Gaussian random vector $\boldsymbol{\xi}(\chi_n)$ is characterized by

$$\mathbf{M}(\chi_n) := (M_1(\mathbf{x}^{(1)}), \dots, M_1(\mathbf{x}^{(n)}), \dots, M_p(\mathbf{x}^{(1)}), \dots, M_p(\mathbf{x}^{(n)}))^\top \in \mathbb{R}^{pn \times 1}. \quad (2.13)$$

and

$$\mathbf{K}_{\chi_n, \chi_n} := \begin{bmatrix} K_{11}(\chi_n, \chi_n) & \cdots & K_{1p}(\chi_n, \chi_n) \\ \vdots & \ddots & \vdots \\ K_{p1}(\chi_n, \chi_n) & \cdots & K_{pp}(\chi_n, \chi_n) \end{bmatrix} \in \mathbb{R}^{pn \times pn}, \quad (2.14)$$

with $K_{ii'}(\chi_n, \chi_n) = \mathbb{E}[(\xi_i(\chi_n) - \mathbb{E}[\xi_i(\chi_n)])(\xi_{i'}(\chi_n) - \mathbb{E}[\xi_{i'}(\chi_n)])^\top] \in \mathbb{R}^{n \times n}$ the cross-covariance matrix between $\xi_i(\chi_n)$ and $\xi_{i'}(\chi_n)$. Similarly, we note $\mathbf{K}_{\chi_n}(\mathbf{x}) \in \mathbb{R}^{pn \times p}$ the cross-covariance matrix between $\boldsymbol{\xi}(\chi_n)$ and $\boldsymbol{\xi}(\mathbf{x})$ defined from blocks

$$K_{ii'}(\mathbf{x}, \chi_n) := (K_{ii'}(\mathbf{x}, \mathbf{x}^{(1)}), \dots, K_{ii'}(\mathbf{x}, \mathbf{x}^{(n)}))^\top \in \mathbb{R}^{1 \times n}. \quad (2.15)$$

Surrogate update formulas, which extend the scalar case formulas (Section 2.1) to the vector case (with β known), are provided as follows (see for example [41] for derivation):

$$M_n(\mathbf{x}) = M(\mathbf{x}) + \mathbf{K}_{\chi_n}(\mathbf{x})^\top \mathbf{K}_{\chi_n, \chi_n}^{-1} (\mathbf{g}(\chi_n) - \mathbf{M}(\chi_n)), \quad (2.16)$$

$$\Sigma_n(\mathbf{x}) = \Sigma(\mathbf{x}) - \mathbf{K}_{\chi_n}(\mathbf{x})^\top \mathbf{K}_{\chi_n, \chi_n}^{-1} \mathbf{K}_{\chi_n}(\mathbf{x}), \quad (2.17)$$

$$K_n(\mathbf{x}, \mathbf{x}') = K(\mathbf{x}, \mathbf{x}') - \mathbf{K}_{\chi_n}(\mathbf{x})^\top \mathbf{K}_{\chi_n, \chi_n}^{-1} \mathbf{K}_{\chi_n}(\mathbf{x}'). \quad (2.18)$$

where M_n , Σ_n and K_n denote the mean, variance and covariance functions, respectively, of the process $\boldsymbol{\xi}_n := \boldsymbol{\xi} | \mathcal{E}_n$.

In the case where the a priori trend is no longer known, but assumed to be a linear combination of basis functions with coefficients to be estimated, there are also update formulas [38]. The sequential procedure in the vector case is similar to the scalar one (see Section 2.1, Figure 2.1): an initial DoE is generated and an initial MOGP is fitted, then they are followed by a succession of MOGP updates and DoE augmentation from a well-chosen enrichment criterion.

3. Three new criteria for estimating partial excursion sets for vector-valued black-box models

3.1. Two natural extensions of scalar Bichon criterion

In this sub-section, for the sake of clarity, the two natural extensions of scalar Bichon criterion are presented in the context of a black-box function with $p = 2$ output components, but can be theoretically generalized to $p > 2$ output components.

3.1.1. Alternating Scalar Bichon criterion

The first criterion we introduce, denoted in the sequel Alternating Scalar Bichon criterion, consists in alternatively maximizing Bichon criterion of each output component with the associated ξ_i surrogate model. For $i = 1, 2$, Bichon criterion associated with surrogate model ξ_i and threshold T_i (see Section 2.1.3) is noted EFF_i . Alternating Scalar Bichon criterion is then expressed as follows:

$$\mathbf{x}^{(n+1)} \in \begin{cases} \operatorname{argmax}_{\mathbf{x} \in \mathbb{X}} \text{EFF}_1(\mathbf{x}) & \text{if } n+1 \text{ is even} \\ \operatorname{argmax}_{\mathbf{x} \in \mathbb{X}} \text{EFF}_2(\mathbf{x}) & \text{otherwise.} \end{cases} \quad (3.1)$$

Alternating Scalar Bichon criterion places equal importance on exploring partial excursion set associated to each output component, regardless of the complexity of their respective estimation. While straightforward to implement, this criterion appears somewhat limited when there may be a significant disparity in the complexity of estimating the two partial excursion sets.

3.1.2. Pareto Scalar Bichon criterion

Instead of alternately maximizing one of the Bichon criteria, it is possible to select a point that offers an optimized trade-off for simultaneously improving knowledge of both outputs near their respective thresholds.

To achieve this, we solve a multi-objective optimization problem where each objective corresponds to Bichon criterion associated with surrogate model ξ_i and threshold T_i . The Pareto solution set is defined by:

$$\mathcal{P} := \{\mathbf{x} \in \mathbb{X} \mid \forall \mathbf{y} \in \mathbb{X}, \exists i \in \{1, 2\}, \text{EFF}_i(\mathbf{y}) < \text{EFF}_i(\mathbf{x})\}. \quad (3.2)$$

Let denote $\mathcal{P}_{\text{EFF}}$ the Pareto front, i.e., the outputs associated with the Pareto solutions. $\mathcal{P}_{\text{EFF}}$ is the set of points that are not dominated by any other point, meaning no other point offers better values for all objectives simultaneously. Each point on the Pareto solution set represents an optimal balance between the different objectives (see, for example, [42] for more details and Figure 3.1 for illustration).

Several methods can be used to select a unique point on the Pareto front. A common approach involves optimizing scalar aggregation functions, which combine multiple objectives into a single criterion [56]. We use a specific example of such a method that minimizes the Euclidean distance to the ideal point, defined as the point with the maximum value for each output [60]. This method was chosen for its simplicity and adaptive behavior: it naturally favors balanced solutions when both objectives can be improved simultaneously, and automatically prioritizes one of the components when they conflict. To illustrate this, consider a 2D example where the goal is to improve the estimation of two partial excursion set boundaries. When the estimated boundaries are close or intersect, the method naturally selects their intersection, improving both components simultaneously. When the estimated boundaries are distant, it favors an extreme point on the Pareto front, focusing on one boundary to achieve significant improvement on at least one component.

In summary, Pareto Scalar Bichon consists in selecting $\mathbf{x}^{(n+1)}$ such that

$$\mathbf{x}^{(n+1)} \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{P}} \left\{ \sqrt{(\text{EFF}_1(\mathbf{x}) - I_1)^2 + (\text{EFF}_2(\mathbf{x}) - I_2)^2} \right\}. \quad (3.3)$$

with $(I_1, I_2) := (\max_{\mathbf{x} \in \mathcal{P}} \text{EFF}_1(\mathbf{x}), \max_{\mathbf{x} \in \mathcal{P}} \text{EFF}_2(\mathbf{x}))$ the ideal point obtained through two single-objective optimizations. To ensure fairness and avoid biasing the selection towards a particular component due to differences in their magnitudes, the output components are normalized based on the initial DoE (see Section 3.3 for more details). For our analysis, we wanted to visualize the Pareto front, so we chose to optimize on $\mathcal{P}$ since the set of Pareto solutions was available. However, in practice, to reduce computation time, it is preferable to optimize on $\mathbb{X}$ in Equation (3.3).

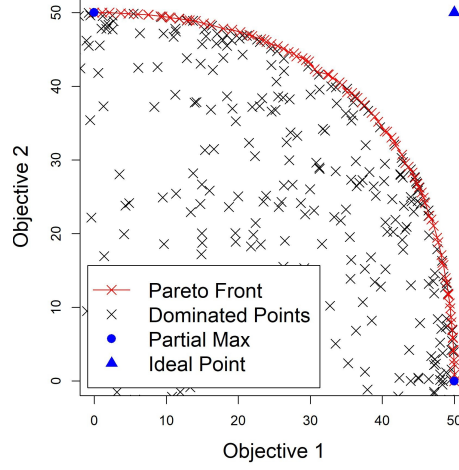


FIGURE 3.1. Example of a Pareto front.

If the Bichon criterion of one of the two output components is relatively flat, corresponding to an already well-identified excursion set, this will lead to an enrichment oriented more towards the other component. Compared to Alternating Scalar Bichon, Pareto Scalar Bichon offers a compromise for the simultaneous estimation of both sets of partial excursions, taking into account the potential uneven difficulty estimation.

3.2. A Vector output extension of Bichon criterion

In this section, we look at a generalization of Bichon criterion (Section 2.1.3) in the context of MOGP regression, again with the aim of estimating the partial excursion sets simultaneously.

Recall that $\Sigma_n(\mathbf{x})$ is the covariance matrix of $\boldsymbol{\xi}_n(\mathbf{x}) := (\xi_{n,1}(\mathbf{x}), \dots, \xi_{n,p}(\mathbf{x}))^\top = \boldsymbol{\xi}(\mathbf{x}) | \mathcal{E}_n$. We extend the scalar Bichon criterion (Section 2.1, Equations (2.6) and (2.7)) to the vector setting as follows:

$$\mathbf{x}^{(n+1)} \in \operatorname{argmax}_{\mathbf{x} \in \mathbb{X}} \operatorname{VEFF}(\mathbf{x}) \quad (3.4)$$

with

$$\operatorname{VEFF}(\mathbf{x}) := \det(\Sigma_n(\mathbf{x}))^{\frac{1}{2p}} \mathbb{E} \left[\left(\kappa - \min_i \left(\frac{|T_i - \xi_i(\mathbf{x})|}{\sigma_{n,i}(\mathbf{x})} \right) \right)^+ \middle| \mathcal{E}_n \right] \quad (3.5)$$

and $\sigma_{n,i}(\mathbf{x}) := \sqrt{(\Sigma_n(\mathbf{x}))_{i,i}}$. The minimum in (3.5) allows to select a point that improves knowledge of, at least, one of the partial excursion sets. The multiplicative term $\sigma_n(\mathbf{x})$ of the scalar Bichon criterion, which helps improving the exploration ability, is replaced by the determinant of the covariance matrix $\Sigma_n(\mathbf{x})$ raised to the power $1/(2p)$. This multiplicative term gives a smaller weight to points $\mathbf{x}$ for which the components of $\boldsymbol{\xi}_n(\mathbf{x})$ have strong correlation. The $1/(2p)$ power ensures homogeneity with the scalar case. This term is simplified to $\sigma_n(\mathbf{x})$ when all marginal variables are independent. It is important to note that the correlation of $\boldsymbol{\xi}$ plays a role not only in the multiplicative term with the determinant, but also in the expectation, as the latter involves a minimum over random variables, thereby depends on the correlation structure. In addition, the statistical interpretation introduced in

Section 2.1 and detailed in [19] can be extended to this new criterion by defining

$$v_{\mathbf{x}} := \min_i \left(\frac{|T_i - \xi_i(\mathbf{x})|}{\sigma_{n,i}(\mathbf{x})} \right) \Big| \mathcal{E}_n. \quad (3.6)$$

In this perspective, the quantity κ in Equation (3.5) is interpreted as a quantile of the distribution of $v_{\mathbf{x}}$.

In the following, we restrict ourselves to the case of a vector-valued black-box model with two output components ($p = 2$). In this context, Proposition 3.1 presents an explicit formulation of Vector Bichon criterion, used for its practical implementation. The proof of this proposition is given in Appendix A.

Proposition 3.1. *Noting $F_{v_{\mathbf{x}}}$ the cumulative distribution function of $v_{\mathbf{x}}$, we have*

$$\text{VEFF}(\mathbf{x}) = \det(\Sigma_n(\mathbf{x}))^{\frac{1}{2p}} \int_0^\kappa F_{v_{\mathbf{x}}}(t) dt. \quad (3.7)$$

Furthermore, when $p = 2$, for every $\mathbf{x} \in \mathbb{X}$ and every $t \in \mathbb{R}$,

$$F_{v_{\mathbf{x}}}(t) = \sum_{i=1}^2 \left(\phi(t + \alpha_i) - \phi(-t + \alpha_i) \right) - \mathbb{P}((U_1, U_2) \in [\alpha_1 - t, \alpha_1 + t] \times [\alpha_2 - t, \alpha_2 + t]), \quad (3.8)$$

with $\alpha_i := \frac{T_i - M_{n,i}(\mathbf{x})}{\sigma_{n,i}(\mathbf{x})}$, ϕ c.d.f of $\mathcal{N}(0, 1)$, $(U_1, U_2) \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$ and ρ the correlation coefficient between $\xi_{n,1}(\mathbf{x})$ and $\xi_{n,2}(\mathbf{x})$.

It should be pointed out that the values α_i and ρ , defined in Proposition 3.1, are functions of n and $\mathbf{x}$, although this dependence is not explicitly stated in the simplified notation. Note also that in Equation (3.8), $F_{v_{\mathbf{x}}}$ depends only on α_i and ρ . A substitution model depending only on α_i and ρ is used to calculate the integral of $F_{v_{\mathbf{x}}}$ in Equation (3.7) (see Appendix B for more details on the calculation of this model, performed outside the enrichment loop). The generalization of Proposition 3.1 to $p > 2$ is conceivable, keeping Equation (3.7) unchanged and modifying Equation (3.8) to include orthants of probabilities of sizes ranging from 2 to p .

3.3. Implementation choices

The implementation choices adopted for the numerical tests of Sections 4 and 5 are detailed below.

3.3.1. Model selection

For both scalar Bichon criteria (Alternating and Pareto), the ξ_i s are assumed to be GPs with constant mean and Matérn 5/2 covariance:

$$k(\mathbf{x}, \mathbf{x}') := \sigma_c^2 \prod_{j=1}^d R_{\text{Matérn } 5/2}(h_j, \theta_j), \quad \forall (\mathbf{x}, \mathbf{x}') \in \mathbb{X}^2, \quad (3.9)$$

with a vector of parameters $\boldsymbol{\theta}$ belonging to $\mathbb{R}_+^{*d}$, estimated by maximizing the likelihood at each iteration, $h_j = |\mathbf{x}_j - \mathbf{x}'_j|$ for $j \in \{1, \dots, d\}$, σ_c a hyperparameter to set or estimate and

$$R_{\text{Matérn } 5/2}(h_j, \theta_j) := \left(1 + \frac{\sqrt{5}|h_j|}{\theta_j} + \frac{5h_j^2}{3\theta_j^2} \right) \exp\left(-\frac{\sqrt{5}|h_j|}{\theta_j} \right). \quad (3.10)$$

Moreover, ξ_1 and ξ_2 are assumed to be independant from each other. This is referred to as a single-output Gaussian process (SOGP).

For Vector Bichon criterion, we use a MOGP with an ICM-type separable kernel (see [34]). Separable models provide a structured approach to handling output correlations, which can facilitate parameter estimation and improve computational efficiency in certain cases. They can also offer flexibility in

MOGPs, as discussed in [3, 41]. They can be expressed as a product of a kernel function in the input space and a kernel function that encodes inter-output correlations. Specifically,

$$(K(\mathbf{x}, \mathbf{x}')_{i,j})_{1 \leq i,j \leq p} := k(\mathbf{x}, \mathbf{x}') \mathbf{B}, \quad (3.11)$$

where k is a scalar kernel on $\mathbb{X}^2$ and $\mathbf{B}$ is a $p \times p$ symmetric positive definite matrix. Hyperparameters of the covariance structure can be estimated using maximum likelihood (see, e.g., [9]).

As outlined in [3, 41], different structures for the matrix $\mathbf{B}$ can reduce the number of parameters to estimate. The choice of $\mathbf{B}$ in our case follows [49], which corresponds to a special case of the “spherical parametrization” introduced in [47]. For $p = 2$, the matrix $\mathbf{B}$ is parameterized as:

$$\mathbf{B} = \sigma_{kOut}^2 \begin{pmatrix} 1 & \cos(\theta_{kOut}) \\ \cos(\theta_{kOut}) & 1 \end{pmatrix}, \quad (3.12)$$

where $\sigma_{kOut} \in \mathbb{R}$ is a scaling factor common to all components (homoscedasticity), and $\theta_{kOut} \in [0, \pi]$ is the parameter controlling inter-output correlation under the spherical parameterization. The scalar kernel k is selected as in Equation (3.9), with $\sigma_c = 1$.

The scalar and vector models presented above are implemented using the *kerpp* package from [17]. Hyperparameter optimization is performed using the local derivative-free optimizer COBYLA algorithm [52] with a multistart of 10, on the following search ranges: $\theta_j \in [0.1, 500]$, $\theta_{kOut} \in [0, \pi]$, $\sigma_c^2 \in [10^{-1}, 10^5]$ et $\sigma_{kOut}^2 \in [10^{-1}, 10^5]$, assuming normalized inputs and outputs.

In the following, we will refer to the two scalar strategies associated with Alternating and Pareto Bichon criteria respectively as “Alternating Scal” and “Pareto Scal”. Vector strategy associated with Vector Bichon criterion will be referred to as “Vect”. In addition to these three proposed strategies, we consider two baseline enrichment methods for comparison: “Random”, which selects new points uniformly at random in the design space $\mathbb{X}$, and “Sobol”, which sequentially extracts points from a Sobol’ sequence with an initial position randomly chosen between 1000 and 100 000. For both baseline methods, scalar surrogate models (similar to those used for Alternating Scal and Pareto Scal) are maintained and updated at each iteration to ensure fair comparison.

3.3.2. Criteria implementation

The implementation of Alternating Scal and Pareto Scal are respectively based on Equations (3.1) and (3.3), with $\kappa = 1$ for the construction of EFF_i . Pareto Scal requires scaled data, that are obtained using the mean and standard deviation of the values of the initial DoE. The Pareto solutions of problem (3.2) are obtained using the NSGA-II algorithm (a widely used multi-objective evolutionary algorithm; [16]) in the *mco* package [57], with 150 generations of size 100, these choices correspond to the recommended default settings in *mco*. Convergence checks are reported in Appendix C.

The implementation of Vect is based on Equations (3.7) and (3.8) of Proposition 3.1. As in the scalar case, the coefficient κ is set to 1. As previously discussed, according to Equation (3.8), $F_{v\mathbf{x}}$ depends solely on α_i and ρ . We thus propose to approximate the function

$$H : \begin{cases} \mathbb{R} \times \mathbb{R} \times [-1, 1] & \longrightarrow \mathbb{R} \\ (\alpha_1, \alpha_2, \rho) & \longmapsto \int_0^\kappa F_{v\mathbf{x}}(t) dt \end{cases} \quad (3.13)$$

beforehand and then replace the integral calculation by its surrogate in the enrichment procedure. The function H is approximated with a Gauss–Legendre quadrature method (see Appendix B for details on interpolation and quadrature methods). Then at each iteration of the enrichment, the approximation of H is evaluated at $(\alpha_1, \alpha_2, \rho)$ with $\alpha_i := \frac{T_i - M_{n,i}(\mathbf{x})}{\sigma_{n,i}(\mathbf{x})}$ and ρ the correlation coefficient between $\xi_{n,1}(\mathbf{x})$ and $\xi_{n,2}(\mathbf{x})$.

3.3.3. Other parameters

Criteria are optimized with the genetic algorithm Genoud (with *pop.size* = 1000) (package *rgenoud*, *mebane2011genetic*). The BFGS quasi-Newton option is enabled (`BFGS = TRUE`, `BFGSburnin = 2`) to perform local refinement on the best candidate every two generations, combining global exploration with gradient-based local optimization. Initial designs of experiments are Latin Hypercube Sampling (LHS) designs optimized from Maximin distance [20], whose size is given in the next section.

4. Numerical experiments on toy examples

We present here the numerical experiments carried out on two analytical test functions in order to compare the different criteria proposed in Section 3.

The performances of the different criteria are compared after n iterations by computing the partial relative approximation error

$$\text{Err}_i := \frac{\mathbb{P}_{\mathbb{X}}(\hat{\Gamma}_i \Delta \Gamma_i^*)}{\mathbb{P}_{\mathbb{X}}(\Gamma_i^*)} \quad (4.1)$$

on each output component, where $\mathbb{P}_{\mathbb{X}}$ denotes the uniform distribution over $\mathbb{X}$, and $\hat{\Gamma}_i$ is defined as the naive estimator

$$\hat{\Gamma}_i := \left\{ \mathbf{x} \in \mathbb{X}, M_{n,i}(\mathbf{x}) \leq T_i \right\} \quad (4.2)$$

where $M_{n,i}$ denotes the i^{th} component of M_n . Each partial relative error measures, for the i^{th} output component, the relative volume of the symmetric difference between estimator $\hat{\Gamma}_i$ and true partial excursion set Γ_i^* defined in Equation (1.2). The sum of partial relative errors, defined by

$$\text{Err}_{\text{sum}} := \sum_i \frac{\mathbb{P}_{\mathbb{X}}(\hat{\Gamma}_i \Delta \Gamma_i^*)}{\mathbb{P}_{\mathbb{X}}(\Gamma_i^*)}, \quad (4.3)$$

is also computed to analyze performances of the various acquisition criteria.

Partial relative errors Err_i and their sum Err_{sum} (Equations (4.1) and (4.3)) require computing the probability $\mathbb{P}_{\mathbb{X}}$ of various subsets of $\mathbb{X}$. For the 2D-input Branin example, these probabilities are computed through a deterministic 400×400 grid (quasi-exact integration). For the 4D-input Hartmann example, we use a quasi-Monte Carlo approximation with a Sobol' sequence of size 10 000 (package *randtoolbox* [21]), which yields estimation errors of order 10^{-3} to 10^{-4} [13, 45], one to two orders of magnitude smaller than the target relative errors (10^{-2}). This precision ensures that data profile classifications (Sections 4.1 and 4.2) based on the threshold $C = 10\%$ reflect genuine algorithmic performance.

4.1. Performance tests on 2D-input Branin-modified function with two outputs

We assess the performance of the proposed criteria using 2D-input Branin function. The first output matches standard Branin function (see for example [51, 55]), while the second is a modified version as in [41]. This is the same test case as in Figure 2.2, defined by Equations (2.8) and (2.9), with a threshold vector $\mathbf{T} := (10, 10)$. The first partial excursion set has three connected components, while the second has only one. Figure 4.1 shows the level sets of the test function and the corresponding partial excursion sets.

4.1.1. Standard performance tests

We repeated the enrichment for forty different initial LHS Maximin DoE of size five. Thirty enrichment iterations are carried out for each of the DoE enrichment strategies and their respective partial relative

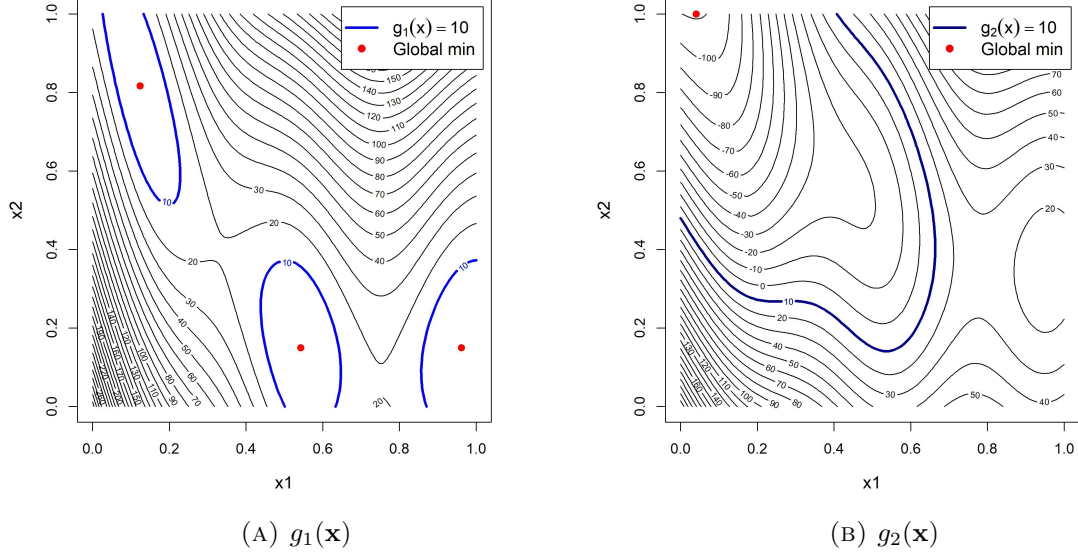


FIGURE 4.1. Representation of the model based on 2D-input Branin function with the first component on the left and the second one on the right and the contour of excursion sets in blue (for $\mathbf{T} = (10, 10)$).

errors and their sum (Equations (4.1) and (4.3)) are monitored. Figure 4.2 shows, in logarithmic scale, the evolution of average errors over the 40 initial DoEs as a function of the number of iterations (top), along with boxplots illustrating the distribution of the error, at selected iterations (bottom). The results show good performances for the three proposed strategies, with errors decreasing significantly as the number of iterations grows, and clearly outperforming the two baseline methods (Random and Sobol). Note that for all five strategies, the relative partial error associated with the second component decreases faster than that associated with the first component, in line with the relative simplicity of the partial excursion set Γ_2^* compared to Γ_1^* (see Figure 4.2). Despite an apparent slight advantage for the Vect strategy at the start of enrichment, Figure 4.2 shows similar results among the three proposed enrichment strategies (Vect, Alternating Scal, and Pareto Scal).

4.1.2. Data profiles as a function of total computing time

These results must be set against the different computation times of the various enrichment strategies. Figure 4.3 shows that the computation time for the Vect strategy sharply increases with the number of iterations. Occasional spikes in computation time are observed for a few runs, particularly for Pareto Scal strategy, due to the combined challenges of Gaussian process hyperparameter optimization and Pareto front computation via NSGA-II. These rare outliers have minimal impact on average performance. In the following, we focus on an indicator that assesses the precision level of a method while accounting for its relative computation time.

Inspired from classical data profiles for optimization [44], the proposed analysis involves examining the percentage of curves, representing the relative error, that consistently remain below a fixed threshold from a given computation time onward. These graphs, referred to as “data profiles”, are functions of the total computation time, which includes both the time needed for enrichment (acquisition criterion optimization and surrogate model updates) and the time spent on evaluating the black-box

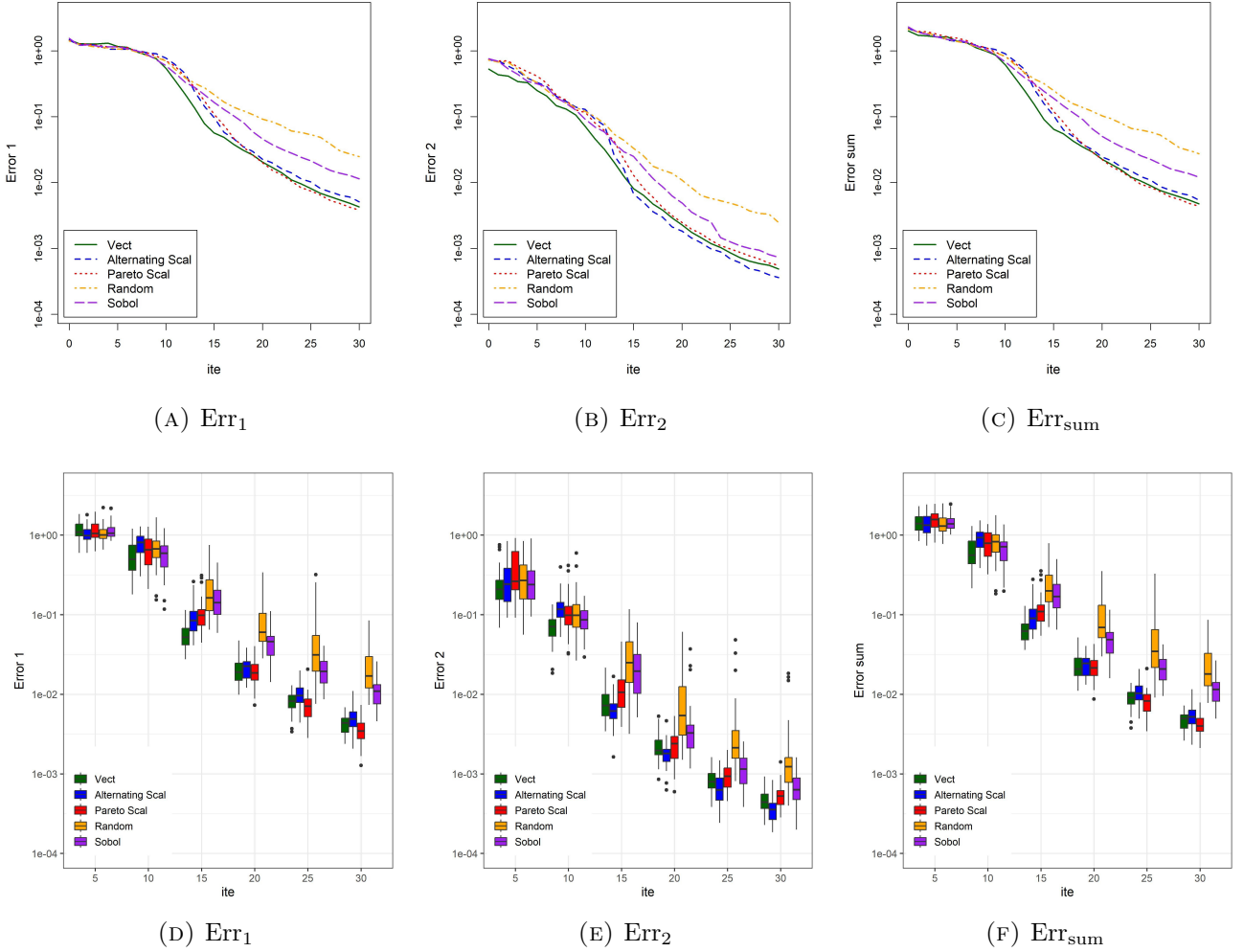


FIGURE 4.2. Partial relative errors and their sum with respect to the number of iterations when estimating excursion sets of the enhanced 2D-input Branin test function with $T = (10, 10)$. The five enrichment strategies Vect, Alternating Scal, Pareto Scal, Random and Sobol are performed from an initial DoE of size 5 and with 30 enrichment iterations. Top: Evolution of average errors over 40 LHS Maximin initial DoEs randomly chosen. Bottom: Boxplots (whisker diagrams) showing the distribution of errors at selected iterations across the 40 repetitions.

model $\mathbf{g}$. They offer a clear visualization of how well the data conforms to a specified threshold. This method is particularly suited to industrial contexts where maintaining errors below a defined tolerance level is critical.

The calculation time required for the output of the toy example is negligible. Therefore, to illustrate the efficiency and computation time behavior of the proposed methods in real-world scenarios (i.e., when dealing with high simulation times, such as in actual industrial problems), we will consider three different settings. Each setting will correspond to the assumption that the output calculation times for the toy problem are: 3 hours, 10 minutes, and 1 minute. More precisely, let t denote the given

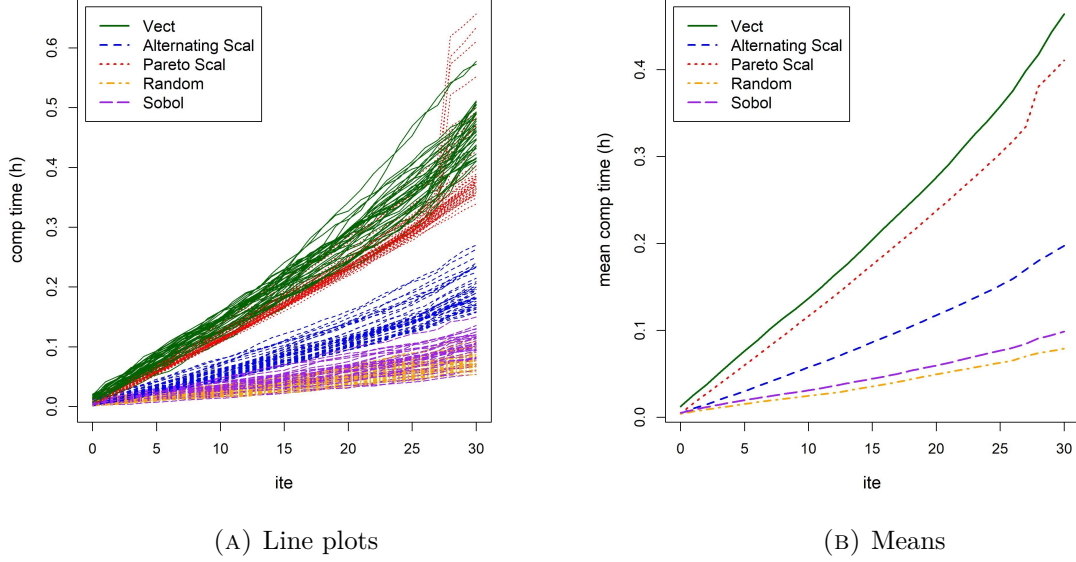


FIGURE 4.3. Enrichment calculation time (assuming that the evaluation time of $\mathbf{g}$ is negligible) for the different criteria, in the case of enrichment of 40 LHS Maximin initial DoEs of size 5 with 30 iterations, for the model based on 2D-input Branin function with $\mathbf{T} = (10, 10)$.

calculation time and C the threshold value for the errors, the partial data profile for the i^{th} output component represents the percentage of curves for which the errors for the i^{th} component is definitively below the threshold C is defined as:

$$\text{DP}_i(t, C) := 100 \times \frac{\#\{j \in \{1, \dots, N_{\text{Repet}}\}, \forall \tilde{t} \geq t, \text{Err}_i^{(j)}(\tilde{t}) < C\}}{N_{\text{Repet}}}, \quad (4.4)$$

where N_{Repet} is the number of repetitions, $\text{Err}_i^{(j)}$ is replaced by the estimated relative error given in Equation (4.1) for the i^{th} component and the j^{th} repetition, and $\#\{A\}$ denotes the cardinality of the set A .

The total data profile represents the percentage of errors definitively below the threshold C across both output components from time t and is expressed as:

$$\text{DP}_{\text{tot}}(t, C) := 100 \times \frac{\#\{j \in \{1, \dots, N_{\text{Repet}}\}, \forall i, \forall \tilde{t} \geq t, \text{Err}_i^{(j)}(\tilde{t}) < C\}}{N_{\text{Repet}}}. \quad (4.5)$$

Using Equations (4.4) and (4.5), we represent both partial and total data profiles for a fixed threshold $C = 10\%$. The results obtained (Figure 4.4) show that for long evaluation times of $\mathbf{g}$ (3 hours), Vect strategy seems to perform slightly better than the other two proposed strategies, as its data profiles reaches 100% more quickly. However, when the evaluation time of $\mathbf{g}$ is shorter, Vect strategy becomes less advantageous and the strategy associated with Alternating Scal seems much more interesting. This is due to the significant computation time (a few minutes) required to update the vector surrogate model, which is negligible when the evaluation time of $\mathbf{g}$ is 3 hours. It should also be noted that the strategy associated with Pareto Scal is more expensive in computation time than that associated with Alternating Scal (Figure 4.3), which explains its inferior performance when the computation

time of $\mathbf{g}$ is very short (1 minute). When the evaluation time for $\mathbf{g}$ is 10 minutes, we find ourselves in an intermediate setting where the Vect strategy retains a slight advantage. Regarding the two baseline methods, while their performance in terms of accuracy is clearly inferior to the three proposed strategies, they demonstrate competitive computational efficiency for relatively short evaluation times of $\mathbf{g}$ (e.g., 1 minute), thanks to their negligible enrichment computation time.

4.1.3. An example where Alternating Scal fails

We look at a modified 2D test case with a specific threshold value $\mathbf{T}$ for which Alternating Scal fails. The test function is still based on 2D-input Branin function, but we choose $\mathbf{T} = (10, 10\,000)$ instead of $\mathbf{T} = (10, 10)$. The partial excursion set associated with the first output component is the same as in the previous setting (Figure 4.1a), while that of the second component corresponds to the full design space $\mathbb{X}$.

We are still considering the enrichment of $N_{\text{repet}} = 40$ initial LHS Maximin DoEs of size 5, with 30 iterations. Figure 4.5 shows the evolution of mean partial relative errors and their sum (top), along with boxplots at selected iterations (bottom). The results reveal that Alternating Scal performs much worse than the other two proposed criteria, with jerky improvements, and even underperforms the two baseline methods in this specific case. This is consistent with the fact that Alternating Scal go on enriching the DoE with respect to the second component (one in two) while the excursion set associated to this component is already fully identified. Moreover, for all strategies, the error associated with the second output component is zero, since the partial excursion set Γ_2^* is very simple to identify.

Unlike the other two proposed strategies, Alternative Scalar criterion is more sensitive to the balance between the difficulty of estimating different sets of partial excursions.

4.2. Performance tests based on 4D-input Hartmann-modified function with two outputs

The test function used is based on 4D-input Hartmann function, and is composed with two output components. The first component is 4D-input Hartmann function [51] defined by

$$\forall \mathbf{x} \in [0, 1]^4, g_1(\mathbf{x}) = \frac{1}{0.839} \left[1.1 - \sum_{i=1}^4 \alpha_i \exp \left(- \sum_{j=1}^4 A_{ij} (x_j - P_{ij})^2 \right) \right], \quad (4.6)$$

with matrices A and P given by

$$A = \begin{pmatrix} 10 & 3 & 17 & 3.50 \\ 0.05 & 10 & 17 & 0.1 \\ 3 & 3.5 & 1.7 & 10 \\ 17 & 8 & 0.05 & 10 \end{pmatrix}, \quad P = 10^{-4} \begin{pmatrix} 1312 & 1696 & 5569 & 124 \\ 2329 & 4135 & 8307 & 3736 \\ 2348 & 1451 & 3522 & 2883 \\ 4047 & 8828 & 8732 & 5743 \end{pmatrix}$$

and $\alpha = (1.0, 1.2, 3.0, 3.2)^\top$. The second output component is obtained by modifying 4D-input Hartmann function, adding noise, generated as uniformly distributed over the interval $[-1, 1]$, to the coefficients α and A :

$$A' = A + 5A_{\text{noise}} \quad \text{and} \quad \alpha' = \alpha + \alpha_{\text{noise}}.$$

No modification of P is made to preserve a certain correlation between the two output components. Rounding the values of $5A_{\text{noise}}$ and α_{noise} to the hundredth and the unit respectively, we finally obtain

$$(A'_{ij}) = \begin{pmatrix} 7 & -2 & 14 & 0.50 \\ -0.95 & 9 & 19 & 5.1 \\ 7 & 1.5 & 1.7 & 10 \\ 16 & 13 & 4.05 & 5 \end{pmatrix} \quad \text{and} \quad \alpha' = (0.44, 0.25, 2.41, 2.63)^\top.$$

MULTI-OUTPUT EXCURSION SETS

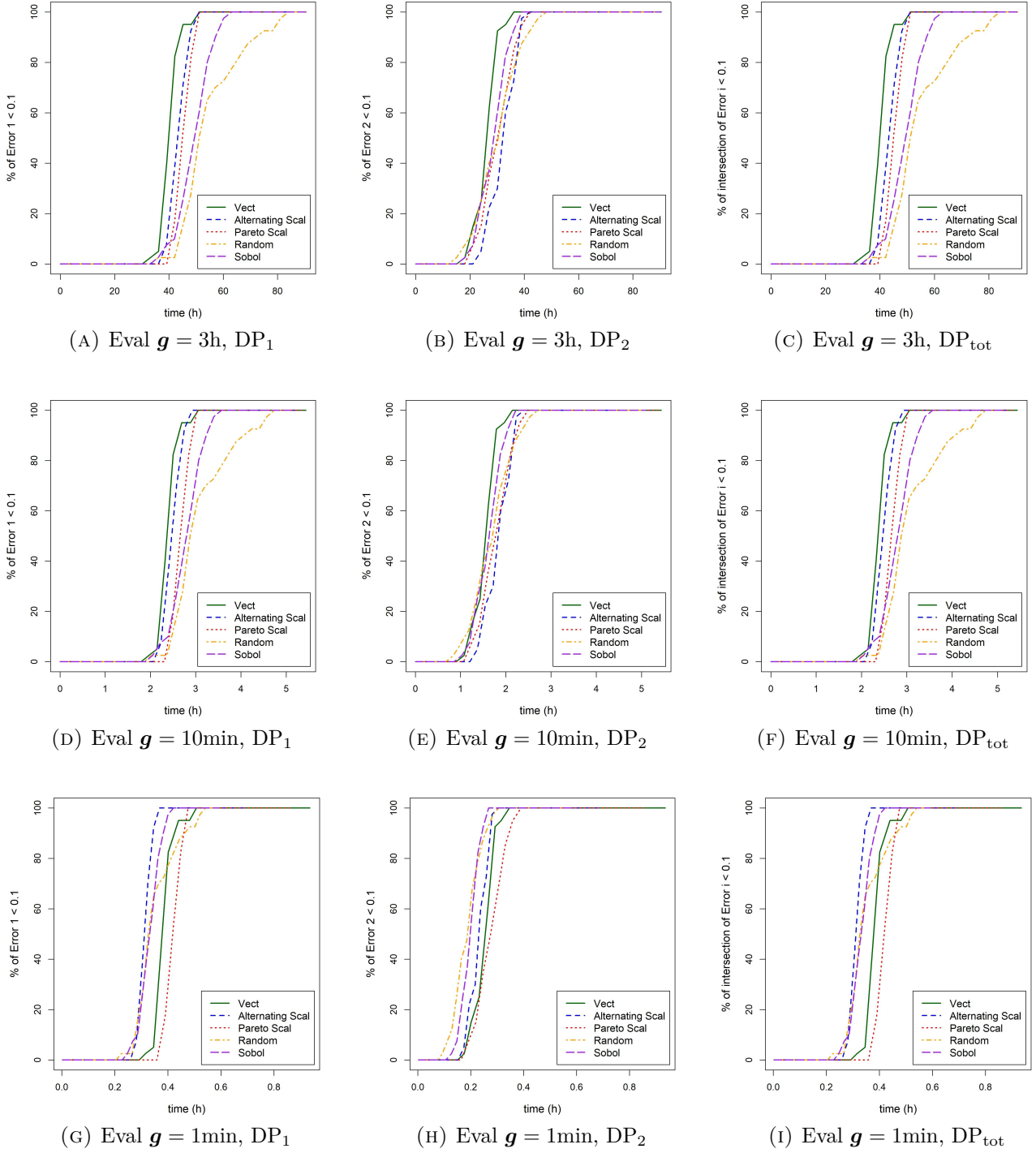


FIGURE 4.4. Data profiles as a function of total computing time of partial relative errors for evaluation times of g of 3h, 10min and 1min, with $C = 10\%$, for the different criteria, in the case of enrichment of 40 LHS Maximin initial DoEs of size 5 with 30 iterations, for the model based on 2D-input Branin function with $T = (10, 10)$.

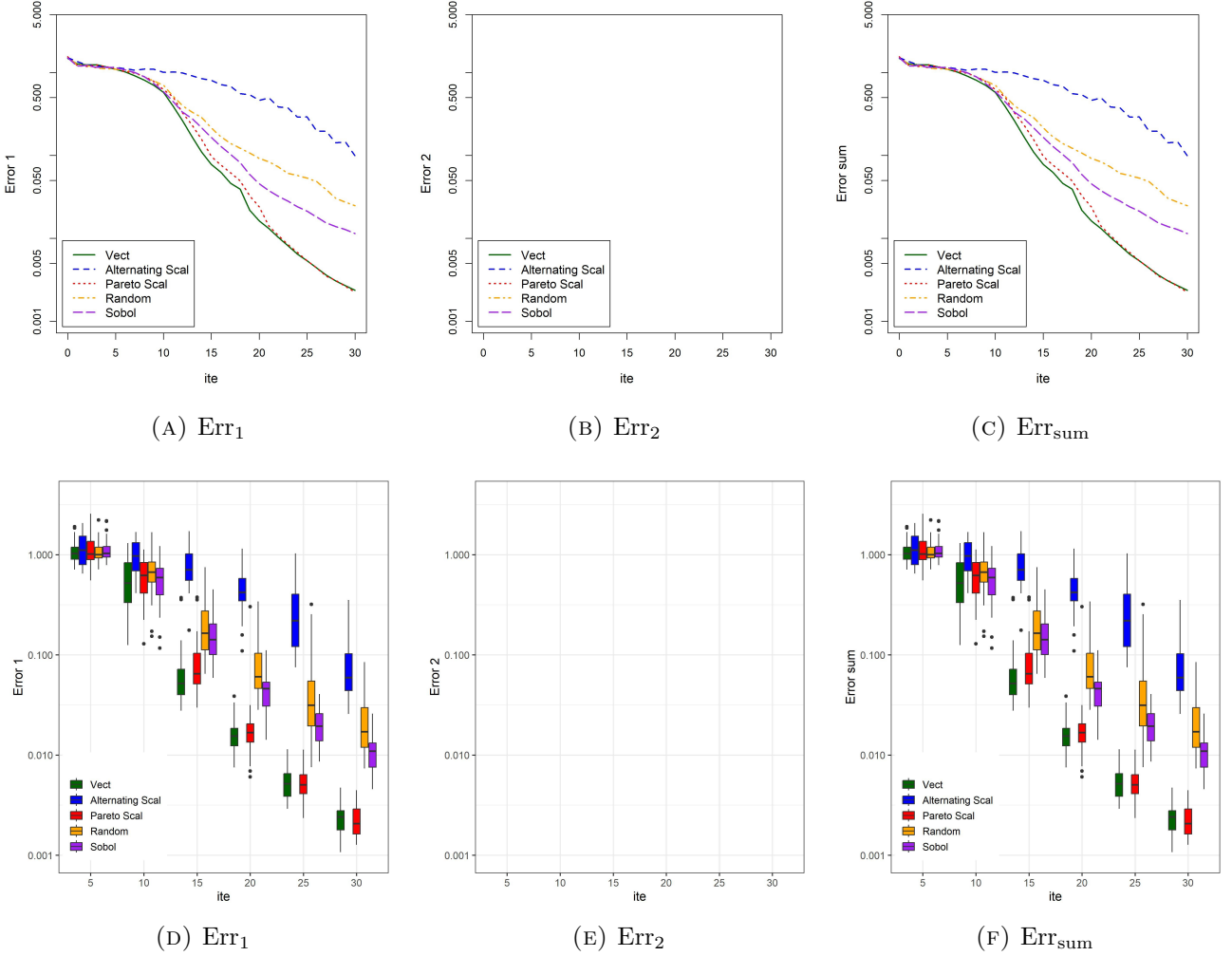


FIGURE 4.5. Partial relative errors and their sum with respect to the number of iterations when estimating excursion sets of the enhanced 2D-input Branin test function with $\mathbf{T} = (10, 10000)$. The five enrichment strategies Vect, Alternating Scal, Pareto Scal, Random and Sobol are performed from an initial DoE of size 5 and with 30 enrichment iterations. Top: Evolution of average errors over 40 LHS Maximin initial DoEs randomly chosen. Bottom: Boxplots (whisker diagrams) showing the distribution of errors at selected iterations across the 40 repetitions.

The choice of the threshold vector, $\mathbf{T} = (-1, -1.6)$, yields relative volumes around 9.5% (relatively to the volume of the total design space) for each partial excursion set, while the volume of the global excursion set represents 5.67% of the total design space. Clustering tests has been carried out on a large data set and show that each of the partial excursion sets is structured in a single connected component.

4.2.1. *Standard performance tests*

In the context of the enrichment of 40 LHS Maximin initial DoEs of size 20, we compare over 200 enrichment iterations the different DoE enrichment strategies by studying the partial relative errors and their sum (Equations (4.1) and (4.3)). Figure 4.6 shows, in logarithmic scale, the evolution of averaged errors over the 40 DoEs as a function of the number of iterations (top), along with boxplots at selected iterations (bottom). The results show good performances for the three proposed strategies, with errors decreasing significantly, and significantly outperforming the two baseline methods (Random and Sobol).

Alternating Scal performs well in estimating the partial excursion set associated with the second component. However, this increased performance on the second component seems to be detrimental to the estimation of the partial excursion set associated with the first one. Vect manages better the trade-off between simultaneous learning of both components, by concentrating efforts where they are most needed, i.e., on the first component. Pareto Scal performs slightly worse than the other two criteria in estimating partial excursion sets, but is nonetheless relatively efficient.

4.2.2. *Data profiles as a function of total computing time*

These results must be set against the higher computation times of Vect strategy compared with Alternating Scal and Pareto Scal strategies (Figure 4.7). With increasing dimension (4D versus 2D), the computation time gap between the baselines and Alternating Scal narrows, as surrogate model update time becomes relatively more significant for the baselines. In the following, we plot data profiles in terms of total computation time (enrichment and $\mathbf{g}$ evaluations) in a similar way to what was done in Section 4.1 (see Equations (4.4) and (4.5)). The threshold is fixed to $C = 10\%$ and the different evaluation times of $\mathbf{g}$ are 3 hours, 10 minutes and 1 minute. The results obtained (Figure 4.8) show that for large evaluation times (3 hours), the curves are analogous to standard data profiles (Figure 4.7). However, when the evaluation time of $\mathbf{g}$ is shorter, Alternating Scal strategy becomes more attractive than the other two proposed strategies, due to its low computation time for sequential DoE enrichment. The long computation time required to update the vector surrogate model of order of a few minutes, is no longer negligible when the evaluation time of $\mathbf{g}$ is small (1 minute). Finally, Pareto Scal and Vect strategies show comparable performances when the evaluation time of $\mathbf{g}$ is small (1 minute), with a slight advantage for Vect strategy. Regarding the baseline methods, their overall performance in this 4D case is less competitive compared to the 2D Branin case, as their computational time advantage over the proposed strategies diminishes with increasing dimension while their accuracy remains significantly lower.

5. Application to pre-calibration of a wind turbine numerical simulator

5.1. Introduction

The application that motivates our work is the pre-calibration of a wind turbine numerical simulator. The numerical simulator outputs the vibration frequencies and deformation modes of a wind turbine at standstill and in response to wind loads. Beyond calibration, the benefits of studying vibratory signatures (frequencies and modes) of a wind turbine is to identify structural faults: mass or stiffness defects, premature wear, component misalignment, etc. (see [12]).

In this context, the pre-calibration goal is to determine the set of values for input parameters, such that the associated vibration frequencies and deformation modes calculated by the simulator match, within specified tolerance thresholds, the reference frequencies and modes obtained from experimental data.

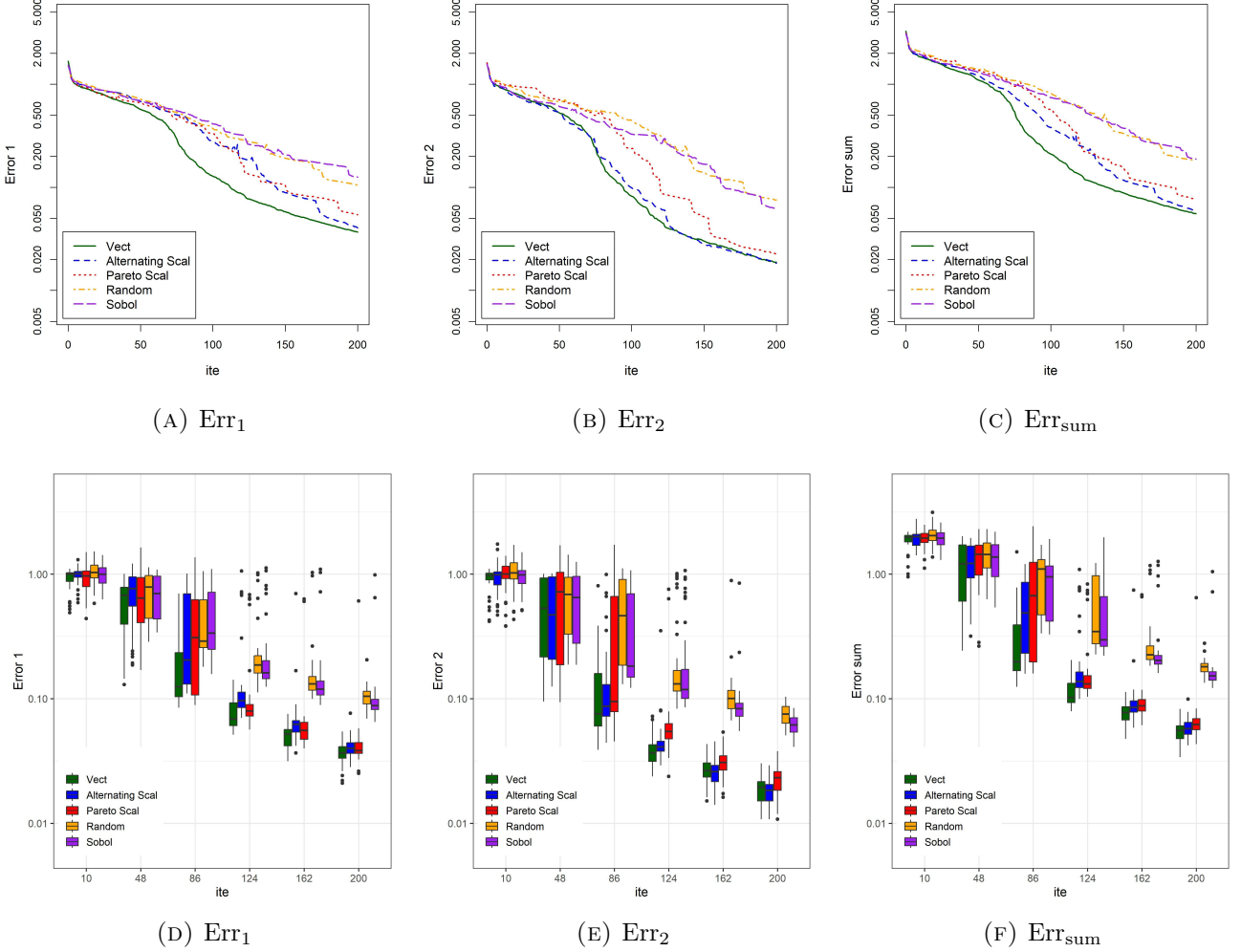


FIGURE 4.6. Partial relative errors and their sum with respect to the number of iterations when estimating excursion sets of 4D-input Hartmann function with $\mathbf{T} = (-1.6, -1)$. The five enrichment strategies Vect, Alternating Scal, Pareto Scal, Random and Sobol are performed from an initial DoE of size 20 and with 200 enrichment iterations. Top: Evolution of average errors over 40 LHS Maximin initial DoEs randomly chosen. Bottom: Boxplots (whisker diagrams) showing the distribution of errors at selected iterations across the 40 repetitions.

The wind turbine under study is the DTU 10-MW Reference Wind Turbine [5]. The numerical simulator [1] is considered as a vector-valued black-box model, taking as input system parameters Θ corresponding to the stiffness coefficients of certain materials, and returning as output the vibration frequencies $\lambda_i(\Theta)$ and deformation modes $\text{Mod}_i(\Theta)$ of the mechanical structure (Figure 5.1), for i ranging from 1 to 26. Simulator data are isotopic, which means that the different simulator output components are evaluated simultaneously for a given evaluation point.

Reference frequencies and modes, denoted by λ_i^* and Mod_i^* , are obtained using the Operational Modal Analysis (OMA) method (see for example [11]). From observed data, this method provides

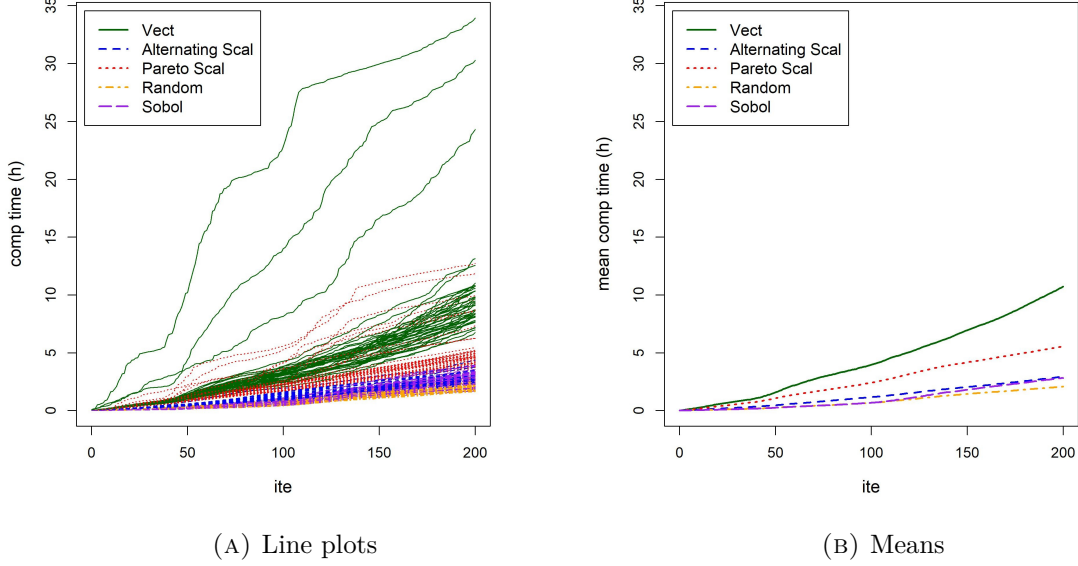


FIGURE 4.7. Enrichment calculation time (assuming that the evaluation time of $\mathbf{g}$ is negligible) for the different criteria, in the case of enrichment of 40 LHS Maximin initial DoEs of size 20 with 200 iterations, for the model based on 4D-input Hartmann function with $\mathbf{T} = (-1.6, -1)$.

the vibratory characteristics of mechanical structures, such as wind turbines, without the need for controlled excitation. Unlike classical modal analysis [28], OMA is based on random excitation modeled as a stochastic process, such as wind, and uses only measured responses to determine the vibration frequencies and modes of the structure. OMA is often combined with algorithms such as the SSI-Cov method, which exploits correlation matrices to estimate state matrices and extract vibration frequencies and modes (see for example [54, 59]). The OMA method used to calibrate numerical simulators by reducing the discrepancy between measured and simulated data is described in more details in [12].

5.2. Pre-calibration as a multiple excursion set estimation problem

We begin by recalling that we use a black-box simulator, which takes as input the Θ stiffness parameters of the wind turbine blades (flap and edge) and returns as output the vibration frequencies $\lambda_i(\Theta)$ and the associated deformation modes $\text{Mod}_i(\Theta)$, for i ranging from 1 to 26. As these frequencies and modes are conjugated in pairs, we limit our study to odd values of i ranging from 1 to 13. According to the information provided by the mechanical experts, the design space, corresponding to the set of simulator input parameters, is defined as $\mathbb{X} := [0.8, 1.2] \times [0.6, 1.4]$. The reference frequencies λ_i^* and modes Mod_i^* should normally be obtained by OMA analysis (see Section 5.1) in an industrial context. Since we did not have access to these data, as proof of concept we simulated here these reference frequencies and modes with $\Theta^* := (\Theta_1^*, \Theta_2^*) = (1, 1)$. Adding noise to these values is crucial to obtain a realistic estimation problem, as measurements are always affected by noise. We use Gaussian multiplicative noise, multiplying the real and imaginary parts of the frequencies obtained by $1 + \epsilon_i$, where ϵ_i is a realization of a centered Gaussian distribution with a standard deviation of 0.025. This guarantees a relative error of less than 5% in 95% of cases. The same strategy noise model is applied to the

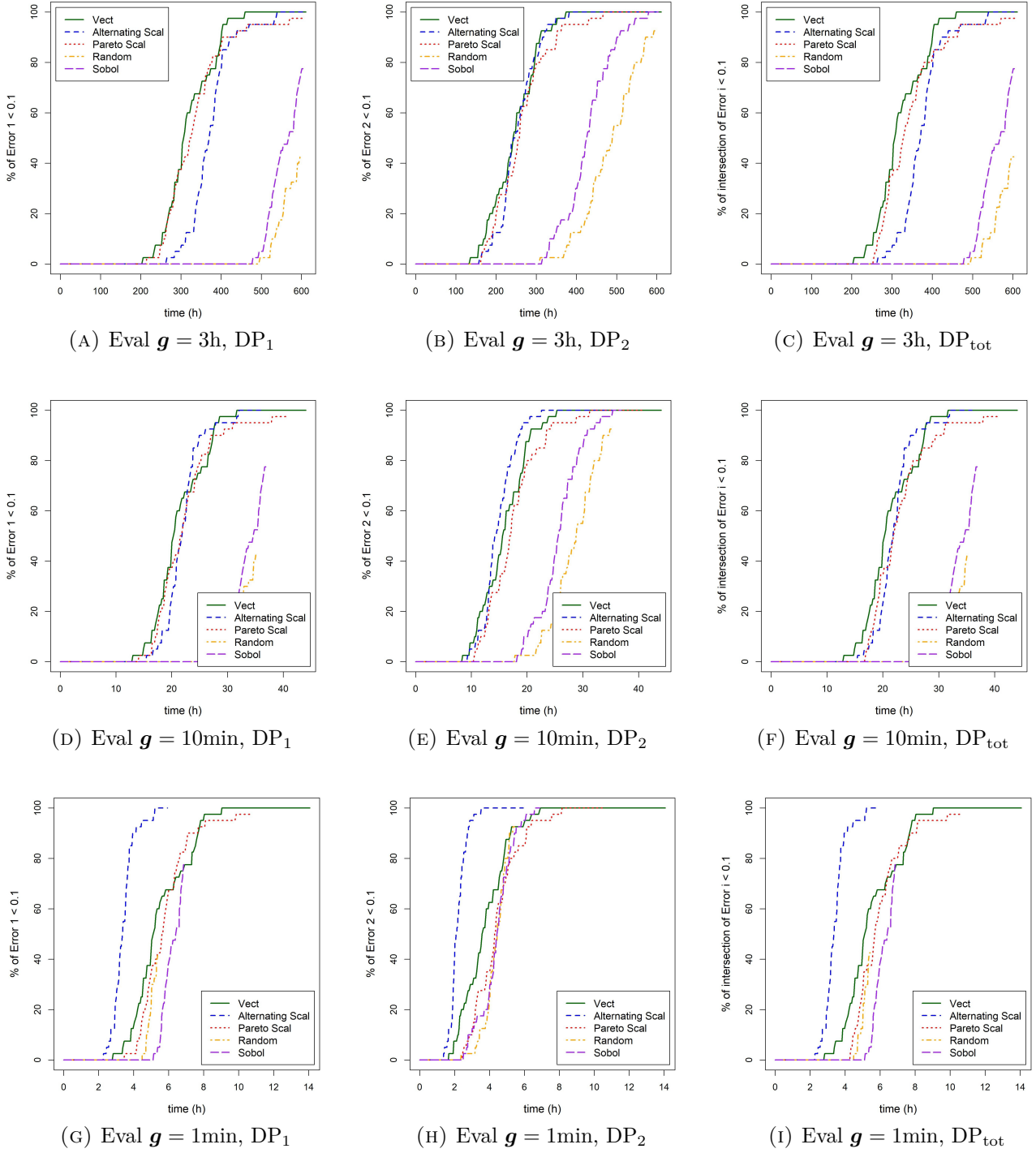


FIGURE 4.8. Data profiles as a function of total computing time of partial relative errors for evaluation times of g of 3h, 10min and 1min, with $C = 10\%$, for the different criteria, in the case of enrichment of 40 LHS Maximin initial DoEs of size 20 with 200 iterations, for the enhanced 4D-input Hartmann test function with $\mathbf{T} = (-1.6, -1)$.

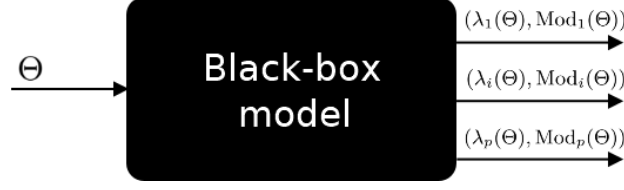


FIGURE 5.1. Schematic diagram of the black-box wind model.

simulated reference modes with a standard deviation of 0.05, ensuring a relative error of less than 10% in 95% of cases.

Following expert opinion, proximity of modes is evaluated by:

$$\text{Meas}_i^{\text{Mod}}(\Theta) := \left(1 - \frac{|\langle \text{Mod}_i^*, \text{Mod}_i(\Theta) \rangle|^2}{\|\text{Mod}_i^*\|^2 \|\text{Mod}_i(\Theta)\|^2}\right), \quad (5.1)$$

with $i \in \{1, \dots, 13\}$ and $\langle \cdot, \cdot \rangle$ the canonical Hermitian scalar product. This measure evaluates the degree of proximity between two vectors, indicating their level of collinearity (evaluated as 0 in the case of perfect collinearity and 1 in the case of orthogonality). This dissimilarity measure is combined with tolerance thresholds $(T_1^{\text{Mod}}, \dots, T_{13}^{\text{Mod}})$.

For vibration frequencies, we measure the proximity by:

$$\text{Meas}_i^\lambda(\Theta) := \frac{|\lambda_i(\Theta) - \lambda_i^*|}{|\lambda_i^*|}. \quad (5.2)$$

This dissimilarity measure is also combined with tolerance thresholds $(T_1^\lambda, \dots, T_{13}^\lambda)$.

For this application, the excursion sets of interest, denoted as $\Gamma_{\text{Mod},i}^*$ and $\Gamma_{\lambda,i}^*$, are defined for $1 \leq i \leq 13$ as:

$$\left\{ \begin{array}{l} \Gamma_{\text{Mod},1}^* := \left\{ \Theta \in \mathbb{X}, g_1^{\text{Mod}}(\Theta) := \text{logit}(\text{Meas}_1^{\text{Mod}}(\Theta)) \leq T_1^{\text{Mod}} \right\}, \\ \vdots \\ \Gamma_{\text{Mod},13}^* := \left\{ \Theta \in \mathbb{X}, g_{13}^{\text{Mod}}(\Theta) := \text{logit}(\text{Meas}_{13}^{\text{Mod}}(\Theta)) \leq T_{13}^{\text{Mod}} \right\}, \\ \Gamma_{\lambda,1}^* := \left\{ \Theta \in \mathbb{X}, g_1^\lambda(\Theta) := \text{logit}(\text{Meas}_1^\lambda(\Theta)) \leq T_1^\lambda \right\}, \\ \vdots \\ \Gamma_{\lambda,13}^* := \left\{ \Theta \in \mathbb{X}, g_{13}^\lambda(\Theta) := \text{logit}(\text{Meas}_{13}^\lambda(\Theta)) \leq T_{13}^\lambda \right\}, \end{array} \right. \quad (5.3)$$

where $\mathbb{X} := [0.8, 1.2] \times [0.6, 1.4]$ and logit denotes the function defined by $\text{logit}(x) := \ln(x/(1-x))$. This function removes the constraint that $\text{Meas}_i^{\text{Mod}}$ and Meas_i^λ to belong to the interval $(0, 1)$, a constraint incompatible with classic GPR.

In our study, we focus on the two principal modes:

$$\left\{ \begin{array}{l} \Gamma_{\text{Mod},1}^* := \left\{ \Theta \in \mathbb{X}, g_1^{\text{Mod}}(\Theta) := \text{logit}(\text{Meas}_1^{\text{Mod}}(\Theta)) \leq T_1^{\text{Mod}} \right\}, \\ \Gamma_{\text{Mod},2}^* := \left\{ \Theta \in \mathbb{X}, g_2^{\text{Mod}}(\Theta) := \text{logit}(\text{Meas}_2^{\text{Mod}}(\Theta)) \leq T_2^{\text{Mod}} \right\}, \end{array} \right. \quad (5.4)$$

which clearly identify Θ_1^* and Θ_2^* . This limitation is necessary since the methods presented in Section 3 are restricted to the simultaneous estimation of only two partial excursion sets.

The short computation times of the black-box $\mathbf{g}$, around 5 seconds, allow us to represent its level sets on a 30×30 grid (Figure 5.2). The dissimilarity measure for the first mode allows good identification of Θ_1^* but not Θ_2^* , whereas the dissimilarity measure for the second mode allows good identification of $\Theta_1^* - \Theta_2^*$.

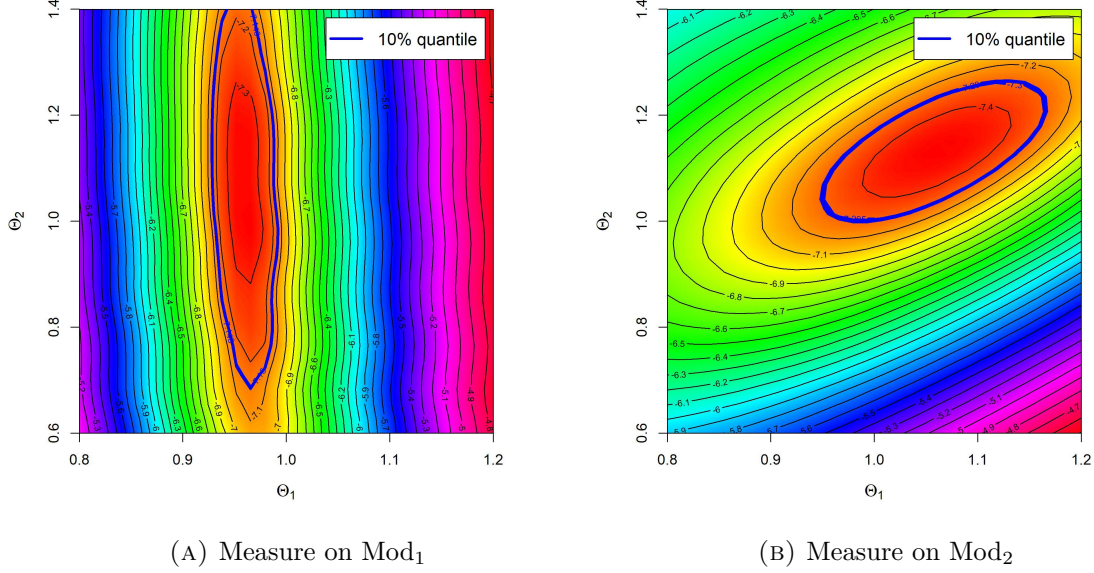


FIGURE 5.2. Representation of the logarithm of dissimilarity measures $\text{Meas}_1^{\text{Mod}}(\Theta)$ (left) and $\text{Meas}_2^{\text{Mod}}(\Theta)$ (right), on a grid of size 30 by 30 for Θ on $[0.8, 1.2] \times [0.6, 1.4]$.

The vector of thresholds $\mathbf{T}$ has been chosen as a quantile of order 10% defined using the 30×30 grid (see Figure 5.2), i.e., $\mathbf{T} = (-7.148, -7.296)$. Solving the pre-calibration problem with this formulation based on the two main modes is detailed in the following section.

5.3. Solving the pre-calibration problem

Enrichment is conducted over 30 iterations starting from an initial LHS Maximin DoE of size 5, employing the three methodologies ('Alternating Scal', 'Pareto Scal', and 'Vect') introduced in Section 3, as well as the two baseline methods ('Random' and 'Sobol'). Implementation choices for surrogate models and enrichment criteria are similar to those presented in Section 3.3. The search bounds of the GP hyperparameters have been adjusted to the $\mathbb{X}$ design space by scaling the bounds selected in Section 4 according to the ranges of each component of $\mathbb{X}$: $\theta_1 \in [0.04, 250]$, $\theta_2 \in [0.08, 400]$. Partial relative errors Err_i and their sum Err_{sum} (Equations (4.1) and (4.3) from Section 4) are estimated from a 30×30 grid by comparing with the true evaluations of the black-box function $\mathbf{g}$ (Figure 5.2).

For an initial DoE, sequential designs and partial excursion set estimates after 30 iterations are shown in Figure 5.3 for the different enrichment strategies. In this example, all three proposed enrichment strategies show notable performance, with estimation of partial excursion sets after 30 iterations all being visually very satisfactory. These results are significantly better than the baseline methods. Tests with other initial experimental designs produced similar results.

Partial relative errors and their sum are shown in Figure 5.4 for 10 different initial DoEs. Overall, the three proposed strategies significantly outperform the two baseline methods, while among the

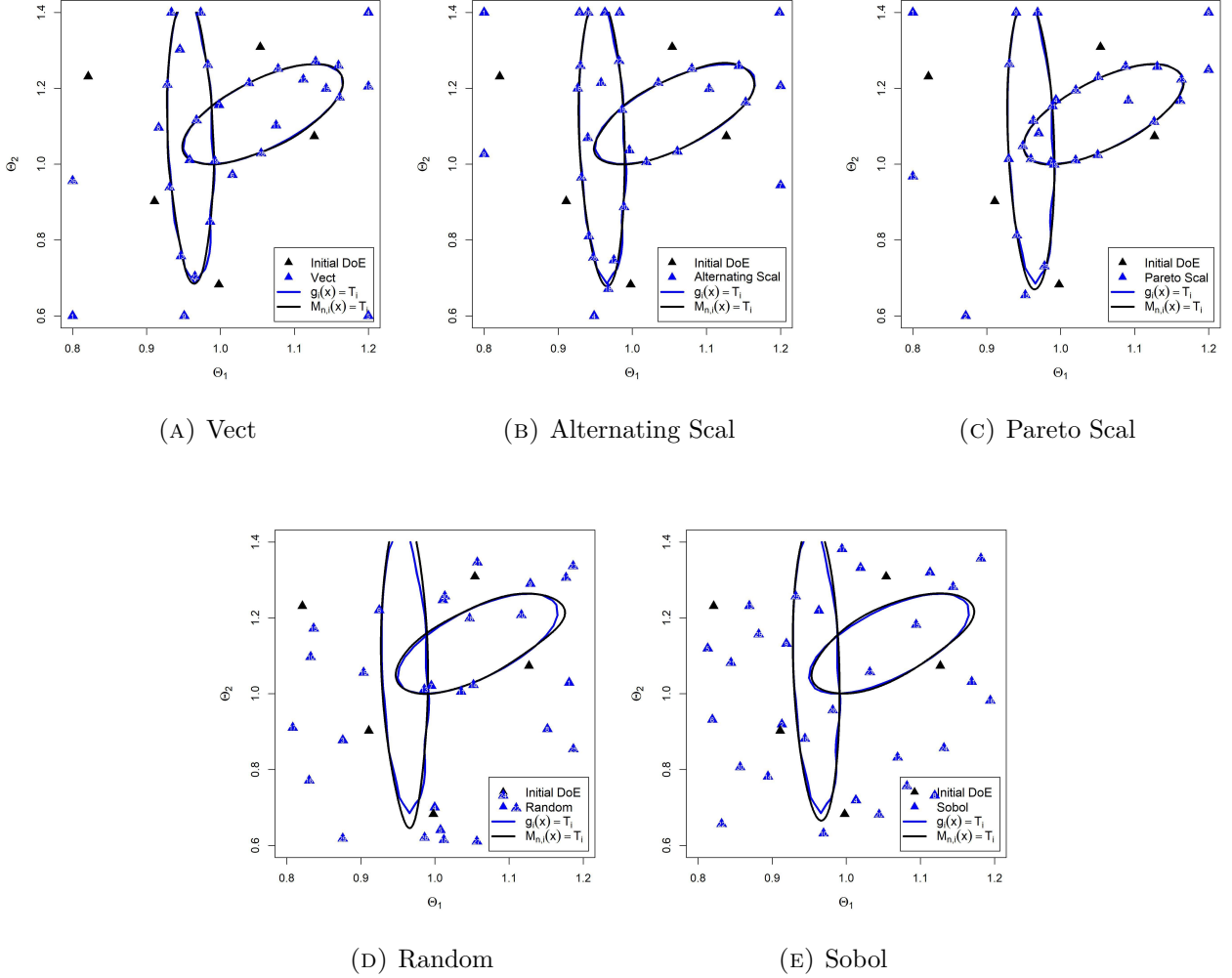


FIGURE 5.3. Representation of the enrichment for an initial LHS Maximin DoE of size 5 after 30 iterations for the different enrichment criteria, for the pre-calibration using only the two main modes with $\mathbf{T} = (-7.148, -7.296)$. Boundaries of partial excursion sets are overlaid, calculated from a 30×30 grid.

proposed strategies, the results are comparable, with slightly better performances for Alternating Scal for partial error Err_2 and for Pareto Scal for Err_1 . The plateaus observed in Figures 5.4A and 5.4B for relative errors below 0.1 underline the limits of using a grid size 30×30 to estimate these errors.

It is also interesting to compare the strategies in terms of hyperparameter estimation (Figure 5.5). We refer to Equations (3.9), (3.10) and (3.12) for hyperparameter description of MOGP. For SOGP (scalar strategies), $\theta_1^{(j)}$ and $\theta_2^{(j)}$ corresponds to the length scales for the j^{th} output component. These plots highlight significant differences in range between the two output components for SOGP models using scalar strategies, consistent with variations of the black-box function observed in Figure 5.2. The hyperparameter evolution for the two baseline methods (not shown here) yields very similar figures to those obtained for Alternating Scal and Pareto Scal strategies, as they all use the same SOGP models. This heterogeneity between output components cannot be captured by MOGP model (Vect strategy)

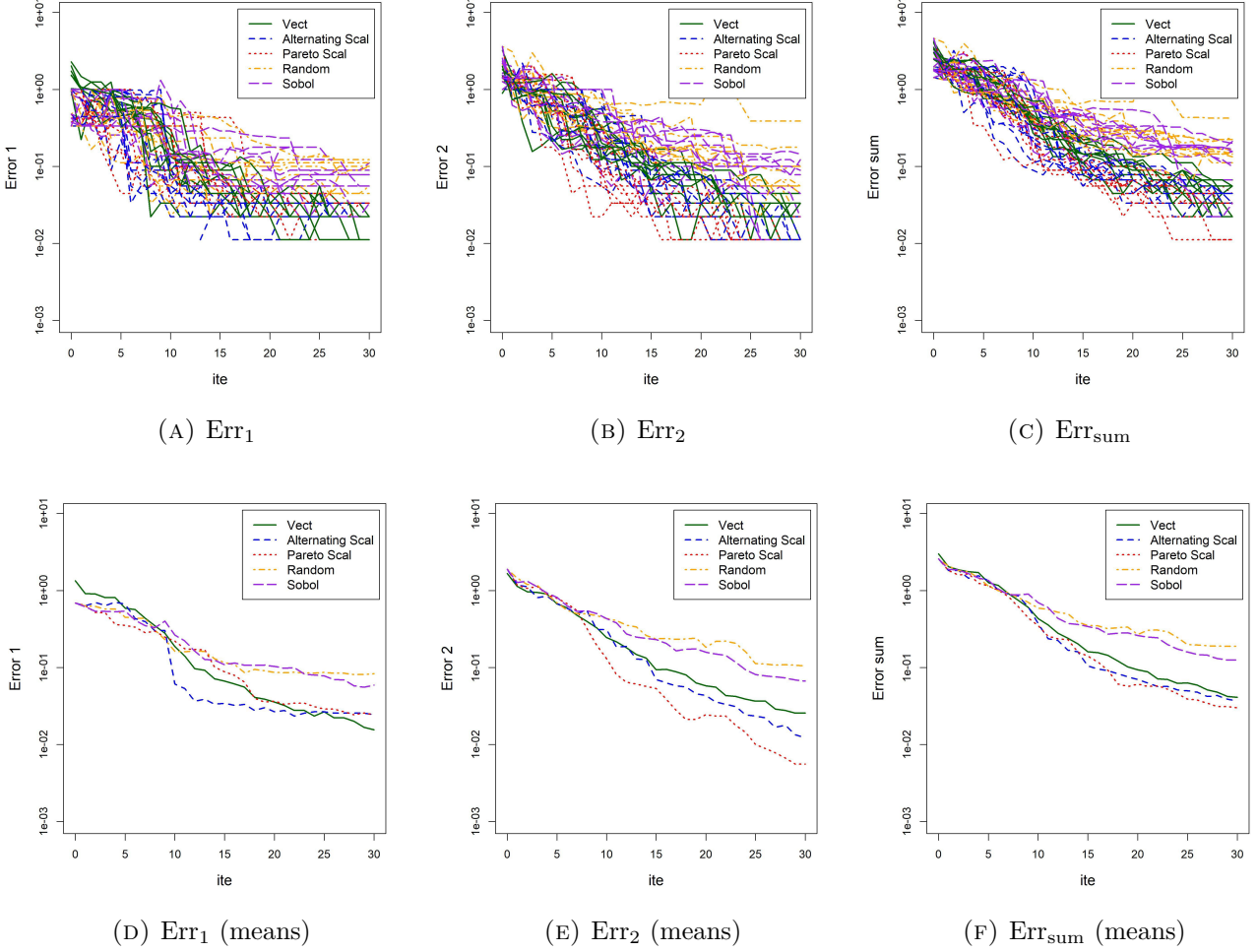


FIGURE 5.4. Line plots and means of partial relative errors and of their sum with the number of iterations when estimating excursion sets of the pre-calibration using only the two main modes with $\mathbf{T} = (-7.148, -7.296)$. The five enrichment strategies Vect, Alternating Scal, Pareto Scal, Random and Sobol are performed from an initial DoE of size 5 and with 30 enrichment iterations. Means are evaluated over 10 LHS Maximin initial DoEs randomly chosen.

as the separable covariance model imposes a common range for both output components. This specific constraint of MOGP explains that the performance of Vect strategy is slightly behind, underlining the limitation. It is also relevant to note that the θ_{kOut} parameter of MOGP (Figure 5.5F) is close to $\pi/2$, indicating a low correlation between the variables.

In summary, the performance of the three proposed strategies evaluated on this industrial application provides very satisfactory estimates of the excursion sets after 30 iterations, significantly outperforming the two baseline methods. The coarse grid of black-box function evaluations is not sufficiently precise to differentiate significantly the evolution of errors among the three proposed strategies. Overall, Vect method proves to be as effective as the two scalar strategies, although the latter benefit from a differentiated estimation of hyperparameters for each output component. These results open up

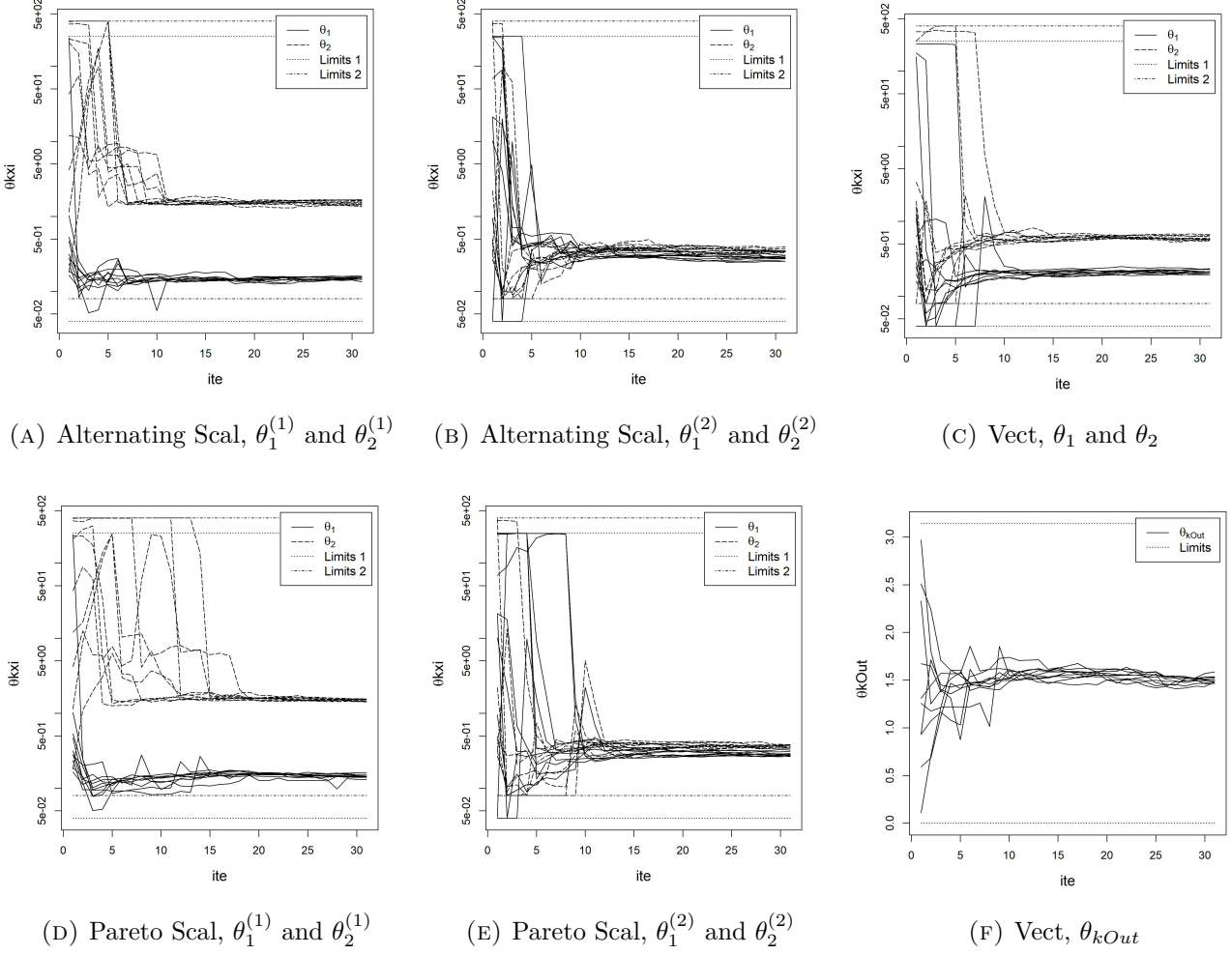


FIGURE 5.5. Evolution of $\theta_i^{(1)}$ and $\theta_i^{(2)}$ hyperparameters of SOGP models associated with Alternating Scal and Pareto Scal, and of the θ_i and θ_{kOut} hyperparameters of MOGP model associated with Vect, as part of an enrichment of initial 10 DoE LHS Maximin of size 5 with 30 iterations, for pre-calibration using only the two principal modes with $\mathbf{T} = (-7.148, -7.296)$.

promising prospects for Vect, notably with more advanced models of MOGPs, non-separable, capable of incorporating component-specific ranges.

Conclusion

We have developed three strategies for simultaneously estimating the partial excursion sets of vector output models, applied to isotopic data. Two approaches are based on scalar models with dedicated enrichment criteria (Alternating Scal and Pareto Scal), while the third uses a MOGP model exploiting correlation between outputs in the enrichment criterion. These three proposed strategies are compared against two baseline methods (random sampling and Sobol' sequence-based sampling) to assess their

performance. Tested on two and four dimensional functions with two output components, the three proposed strategies significantly reduce partial relative errors, consistently outperforming the baseline methods. In 2D, the baseline methods remain competitive when performance is evaluated via data profiles, benefiting from their low computational cost. However, this competitiveness vanishes in higher dimensions as the cost gap with the three proposed strategies decreases while the accuracy gap remains substantial. Analysis of the results shows that Vect strategy offers a good compromise, particularly in 4D, although it is more computationally expensive. For computationally cheap simulators, Alternating Scal strategy is more efficient, but can fail in specific cases, unlike the more robust Vect and Pareto Scal strategies (for instance, if a threshold T_i is far outside the range of the i^{th} output over $\mathbb{X}$, then $\Gamma_i^* = \{\mathbf{x} \in \mathbb{X} : g_i(\mathbf{x}) \leq T_i\}$ is almost empty or almost equal to $\mathbb{X}$, making this component trivial while another remains difficult to estimate). Interestingly, in such failure cases, Alternating Scal can even underperform the baseline methods. One way of overcoming the difficulty associated with the Alternating strategy would be to adopt an alternating criterion that would automatically detect the correct EFF_i to optimize.

The sequential enrichment strategies for the DoE presented in Section 3 have been applied to the pre-calibration of a wind turbine simulator. This simulator predicts vibration frequencies and deformation modes under the effect of wind, from input parameters such as stiffness coefficients. The aim is to identify input parameters that match the experimental data obtained by Operational Modal Analysis (OMA). Simulator evaluations have been limited by the absence of real OMA and therefore perform on synthetic data we generated. The integration of experimental data remains an essential challenge for concrete implementation. The sequential enrichment strategies developed were applied to the simultaneous estimation of partial excursion sets associated with the two main deformation modes of a pre-calibration problem.

The obtained results confirm the effectiveness of the three proposed strategies compared to the baseline methods, despite certain limitation of the MOGP model used for Vect strategy, notably due to a simplified correlation model between outputs. To improve accuracy, more complex and adaptive correlation structures should be explored to better model interactions between output components. Although our tests are limited to two output components, these strategies remain theoretically generalizable to a higher number of components. However, their extension will require algorithmic developments, focused on computational time optimization, to ensure their applicability to larger numbers of simulator outputs. Regarding scalability, the methods are applicable to higher-dimensional input spaces, though surrogate model fitting and acquisition optimization become more expensive as dimension increases; dimension reduction techniques such as KPLS [10] could mitigate these challenges. For the number of outputs, Alternating Scal extends naturally with linear cost, Pareto Scal faces multi-objective optimization challenges, and Vect strategy requires generalizing Proposition 3.1 for orthonant probability computation in dimension $p > 2$ (TVPACK is suited to $p \leq 3$) and adapting the MOGP coregionalization structure.

In practice, the choice between the three enrichment strategies depends primarily on the simulator evaluation time and the characteristics of the outputs. For cheap simulators (seconds to minutes), prefer Alternating Scal strategy due to its low enrichment overhead, though it can fail when partial excursion sets have unbalanced estimation difficulty, i.e., when one partial set is easy to recover while another still requires boundary-focused exploration. For expensive simulators (hours), prefer Vect strategy as it exploits inter-output correlations to reduce the number of costly evaluations, despite higher enrichment computation time. When a balanced, adaptive compromise is desired without strong prior knowledge, Pareto Scal strategy is a robust compromise.

Validated in an industrial setting, these methods open up prospects for other sectors requiring high-performance models with multiple outputs such as healthcare, automotive and aerospace, opening up new application prospects.

Appendix A. Proof of the integral formulation for Vector Bichon criterion

Proposition A.1. Noting $F_{v_{\mathbf{x}}}$ the cumulative distribution function of $v_{\mathbf{x}} := \min_i \left(\frac{|T_i - \xi_i(\mathbf{x})|}{\sigma_{n,i}(\mathbf{x})} \right) \Big| \mathcal{E}_n$, we have

$$\text{VEFF}(\mathbf{x}) = \det(\Sigma_n(\mathbf{x}))^{\frac{1}{2p}} \int_0^\kappa F_{v_{\mathbf{x}}}(t) dt. \quad (\text{A.1})$$

Furthermore, when $p = 2$, for every $\mathbf{x} \in \mathbb{X}$ and every $t \in \mathbb{R}$,

$$F_{v_{\mathbf{x}}}(t) = \sum_{i=1}^2 \left(\phi(t + \alpha_i) - \phi(-t + \alpha_i) \right) - \mathbb{P}((U_1, U_2) \in [\alpha_1 - t, \alpha_1 + t] \times [\alpha_2 - t, \alpha_2 + t]), \quad (\text{A.2})$$

with $\alpha_i := \frac{T_i - M_{n,i}(\mathbf{x})}{\sigma_{n,i}(\mathbf{x})}$, ϕ c.d.f of $\mathcal{N}(0, 1)$, $(U_1, U_2) \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$ and ρ the correlation coefficient between $\xi_{n,1}(\mathbf{x})$ and $\xi_{n,2}(\mathbf{x})$.

Lemma A.2. Let $\kappa > 0$ and Y be a non negative real random variable. Then we have

$$\mathbb{E}[(\kappa - Y)^+] = \int_0^\kappa F_Y(t) dt, \quad (\text{A.3})$$

where $(\cdot)^+$ denotes the positive part function and F_Y the cumulative distribution function of Y .

Proof.

$$\begin{aligned} \mathbb{E}[(\kappa - Y)^+] &= \mathbb{E}[(\kappa - Y)\mathbf{1}_{\kappa - Y \geq 0}] \\ &= \kappa \mathbb{E}[\mathbf{1}_{Y \leq \kappa}] - \mathbb{E}[Y\mathbf{1}_{Y \leq \kappa}] \\ &= \kappa F_Y(\kappa) - \int_0^{+\infty} \mathbb{P}[Y\mathbf{1}_{Y \leq \kappa} \geq t] dt \quad (\text{because } Y\mathbf{1}_{Y \leq \kappa} \geq 0) \\ &= \kappa F_Y(\kappa) - \int_0^\kappa \mathbb{P}[Y\mathbf{1}_{Y \leq \kappa} \geq t] dt \\ &= \kappa F_Y(\kappa) - \int_0^\kappa \mathbb{P}[t \leq Y \leq \kappa] dt \\ &= \kappa F_Y(\kappa) - \int_0^\kappa (F_Y(\kappa) - F_Y(t)) dt \\ &= \int_0^\kappa F_Y(t) dt. \end{aligned} \quad \blacksquare$$

Proof of Proposition A.1. The integral formulation of Equation (A.1) follows directly from Lemma A.2 applied with $Y = v_{\mathbf{x}}$ (see initial formulation of VEFF in Equation (3.5)).

To calculate $F_{v_{\mathbf{x}}}$, we note $\mathbb{P}_n := \mathbb{P}(\cdot | \mathcal{E}_n)$. Let $t \in \mathbb{R}$,

$$\begin{aligned} F_{v_{\mathbf{x}}}(t) &= \mathbb{P}_n \left[\min_i \left(\frac{|T_i - \xi_i(\mathbf{x})|}{\sigma_{n,i}(\mathbf{x})} \right) \leq t \right] \\ &= \underbrace{\mathbb{P}_n \left(\frac{|T_1 - \xi_1(\mathbf{x})|}{\sigma_{n,1}(\mathbf{x})} \leq t \right) + \mathbb{P}_n \left(\frac{|T_2 - \xi_2(\mathbf{x})|}{\sigma_{n,2}(\mathbf{x})} \leq t \right)}_{\textcircled{1}} \\ &\quad - \underbrace{\mathbb{P}_n \left(\left\{ \frac{|T_1 - \xi_1(\mathbf{x})|}{\sigma_{n,1}(\mathbf{x})} \leq t \right\} \cap \left\{ \frac{|T_2 - \xi_2(\mathbf{x})|}{\sigma_{n,2}(\mathbf{x})} \leq t \right\} \right)}_{\textcircled{2}}. \end{aligned} \quad (\text{A.4})$$

The marginal distributions of $\boldsymbol{\xi}(\mathbf{x}) := (\xi_1(\mathbf{x}), \xi_2(\mathbf{x}))^\top$ verify

$$\xi_i(\mathbf{x}) \sim \mathcal{N}(M_{n,i}(\mathbf{x}), \sigma_{n,i}(\mathbf{x})^2), \forall i \in \{1, 2\},$$

so for i fixed,

$$\begin{aligned} \mathbb{P}_n\left(\frac{|T_i - \xi_i(\mathbf{x})|}{\sigma_{n,i}(\mathbf{x})} \leq t\right) &= \mathbb{P}_n\left(-t \leq \frac{\xi_i(\mathbf{x}) - T_i}{\sigma_{n,i}(\mathbf{x})} \leq t\right) \\ &= \mathbb{P}_n\left(-t + \frac{T_i - M_{n,i}(\mathbf{x})}{\sigma_{n,i}(\mathbf{x})} \leq \frac{\xi_i(\mathbf{x}) - M_{n,i}(\mathbf{x})}{\sigma_{n,i}(\mathbf{x})} \leq t + \frac{T_i - M_{n,i}(\mathbf{x})}{\sigma_{n,i}(\mathbf{x})}\right) \\ &= \phi(t + \alpha_i) - \phi(-t + \alpha_i), \end{aligned}$$

with $\alpha_i := \frac{T_i - M_{n,i}(\mathbf{x})}{\sigma_{n,i}(\mathbf{x})}$ and ϕ c.d.f. of $\mathcal{N}(0, 1)$. Thus,

$$\textcircled{1} = \sum_{i=1}^2 \left(\phi(t + \alpha_i) - \phi(-t + \alpha_i) \right). \quad (\text{A.5})$$

Next,

$$\textcircled{2} = \mathbb{P}_n\left(\left\{\frac{|T_1 - \xi_1(\mathbf{x})|}{\sigma_{n,1}(\mathbf{x})} \leq t\right\} \cap \left\{\frac{|T_2 - \xi_2(\mathbf{x})|}{\sigma_{n,2}(\mathbf{x})} \leq t\right\}\right) \quad (\text{A.6})$$

$$\begin{aligned} &= \mathbb{P}_n\left((\xi_1(\mathbf{x}), \xi_2(\mathbf{x})) \in [T_1 \pm t \sigma_{n,1}(\mathbf{x})] \times [T_2 \pm t \sigma_{n,2}(\mathbf{x})]\right) \\ &= \mathbb{P}_n\left(\left(\frac{\xi_1(\mathbf{x}) - M_{n,1}(\mathbf{x})}{\sigma_{n,1}(\mathbf{x})}, \frac{\xi_2(\mathbf{x}) - M_{n,2}(\mathbf{x})}{\sigma_{n,2}(\mathbf{x})}\right) \in [\alpha_1 \pm t] \times [\alpha_2 \pm t]\right) \\ &= \mathbb{P}((U_1, U_2) \in [\alpha_1 \pm t] \times [\alpha_2 \pm t]), \end{aligned} \quad (\text{A.7})$$

with $(U_1, U_2) \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$ and ρ the correlation coefficient between $\xi_1(\mathbf{x})$ and $\xi_2(\mathbf{x})$.

By reinjecting $\textcircled{1}$ and $\textcircled{2}$ from Equations (A.5) and (A.7) into Equation (A.4), we obtain the expected result. $\blacksquare$

Appendix B. Estimation of $H : (\alpha_1, \alpha_2, \rho) \mapsto \int_0^\kappa F_{v_{\mathbf{x}}}(t) dt$

In this appendix, we present the way in which, for a fixed $\mathbf{x}$, the quantity $\int_0^\kappa F_{v_{\mathbf{x}}}(t) dt$, required to implement Vector Bichon criterion, is estimated (see Proposition 3.1, Equation (3.7)). For implementation, κ is set to 1, but retained here for generality. According to Equation (3.8), the quantity $F_{v_{\mathbf{x}}}$ (and therefore its integral between 0 and κ) depend only on α_i and ρ . We present the interpolation method used to estimate the function

$$H : \begin{cases} \mathbb{R} \times \mathbb{R} \times [-1, 1] & \longrightarrow \mathbb{R} \\ (\alpha_1, \alpha_2, \rho) & \longmapsto \int_0^\kappa F_{v_{\mathbf{x}}}(t) dt \end{cases}.$$

The first part focuses on estimating H for fixed $(\alpha_1, \alpha_2, \rho) \in \mathbb{R}^2 \times [0, 1]$ using Gauss–Legendre quadrature at interpolation points. The second part explores the symmetries of H to optimize computation time. The third part deals with interpolation on $[-50, 50]^2 \times [-1, 1]$, while interpolation outside this domain is addressed in the final part.

Estimation of H for $(\alpha_1, \alpha_2, \rho)$ fixed using Gaussian quadrature method

According to Proposition 3.1,

$$H(\alpha_1, \alpha_2, \rho) := \int_0^\kappa F_{v_{\mathbf{x}}}(t) dt \quad (\text{B.1})$$

and for every $\mathbf{x} \in \mathbb{X}$ and every $t \in \mathbb{R}$,

$$F_{v_{\mathbf{x}}}(t) = \sum_{i=1}^2 \left(\phi(t + \alpha_i) - \phi(-t + \alpha_i) \right) - \mathbb{P}((U_1, U_2) \in [\alpha_1 \pm t] \times [\alpha_2 \pm t]), \quad (\text{B.2})$$

with ϕ c.d.f of $\mathcal{N}(0, 1)$ and $(U_1, U_2) \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$. The calculation of $F_{v_{\mathbf{x}}}(t)$ in (B.2) involves an orthant probability, in the particular case of the bivariate centered normal distribution (see for example [4] for a recent review on orthant probabilities). This orthant is obtained using the *pvtnorm* function in the *mvtnorm* package in **R** language. The calculation method used is ‘*TVPACK*’ from [31], which is particularly suited to the 2 and 3 dimensions.

For $(\alpha_1, \alpha_2, \rho)$ fixed, the estimation of $H(\alpha_1, \alpha_2, \rho) := \int_0^\kappa F_{v_{\mathbf{x}}}(t) dt$ is done using a Gauss–Legendre quadrature method (see [2]) between 0 and κ with the package *gaussquad* and $n_{\text{quad}} = 10$ quadrature points. With ten quadrature points, the relative error is 10^{-16} and increasing the number of points does not improve accuracy due to limitations in estimating $F_{v_{\mathbf{x}}}$. The results of estimating H as a function of α_1 and α_2 and for different values of ρ are shown in Figure B.1 (with $\kappa = 1$). The function exhibits a certain regularity as a function of α_1 , α_2 and ρ . Also, when ρ is close to 0 the maximum of the function H seems to be located around $(\alpha_1, \alpha_2) = (0, 0)$, whereas when $|\rho| > 0.5$ it appears that there are two maxima, symmetrically positioned relatively to point $(0, 0)$. This is consistent since, for example, if $\rho \simeq 1$ the random variables are highly correlated and so a α_1 and a α_2 small in $|\cdot|$ but of opposite signs provide more information than $\alpha_1 = \alpha_2 = 0$.

Study of the symmetries of the H function

The H function has several symmetries, which is beneficial in terms of reducing the computation time for the interpolation presented next. Let $m \in \mathbb{R}^2$ and $\Sigma \in \mathbb{R}^{2 \times 2}$, let $\mathcal{N}(m, \Sigma)$ be the bivariate normal distribution with mean m and covariance matrix Σ . By definition of H ,

$$H(\alpha_1, \alpha_2, \alpha) = \int_0^\kappa \mathbb{P}(\min(|X_1|, |X_2|) \leq t) dt, \quad (\text{B.3})$$

with $(X_1, X_2) \sim \mathcal{N}\left(\begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$. With these notations, $(X_1, -X_2) \sim \mathcal{N}\left(\begin{pmatrix} \alpha_1 \\ -\alpha_2 \end{pmatrix}, \begin{pmatrix} 1 & -\rho \\ -\rho & 1 \end{pmatrix}\right)$ and $(-X_1, -X_2) \sim \mathcal{N}\left(\begin{pmatrix} -\alpha_1 \\ -\alpha_2 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right)$, which allows us to demonstrate the two following symmetries on the function H :

$$\forall \rho \in [-1, 1], \forall (\alpha_1, \alpha_2) \in \mathbb{R}^2, H(\alpha_1, \alpha_2, \rho) = H(\alpha_1, -\alpha_2, -\rho); \quad (\text{B.4})$$

and

$$\forall \rho \in [-1, 1], \forall (\alpha_1, \alpha_2) \in \mathbb{R}^2, H(\alpha_1, \alpha_2, \rho) = H(-\alpha_1, -\alpha_2, \rho). \quad (\text{B.5})$$

Equation (B.4) represents, for a fixed α_1 , a central symmetry with respect to point $(\alpha_2, \rho) = (0, 0)$, which restricts our study to $\rho \in [0, 1]$. Furthermore, Equation (B.5), for a fixed ρ , exhibits central symmetry with respect to the point $(\alpha_1, \alpha_2) := (0, 0)$, thus restricting our study to $\alpha_1 \in \mathbb{R}^+$.

In summary, the symmetries of Equations (B.4) and (B.5) make it possible to restrict the interpolation of H function to the set $\mathbb{R}^+ \times \mathbb{R} \times [0, 1]$. In numerical terms, it makes sense to leave a margin equivalent to the interpolation step size at the boundaries ($\alpha_1 = 0$ and $\rho = 0$) to ensure consistent regularity of the interpolation function. We also note the presence of two axial symmetries with respect to the lines $\alpha_2 = \alpha_1$ and $\alpha_2 = -\alpha_1$ (see Figure B.1). However, these symmetries are not taken into account to reduce the computation time required for interpolation, as their implementation would be too complex.

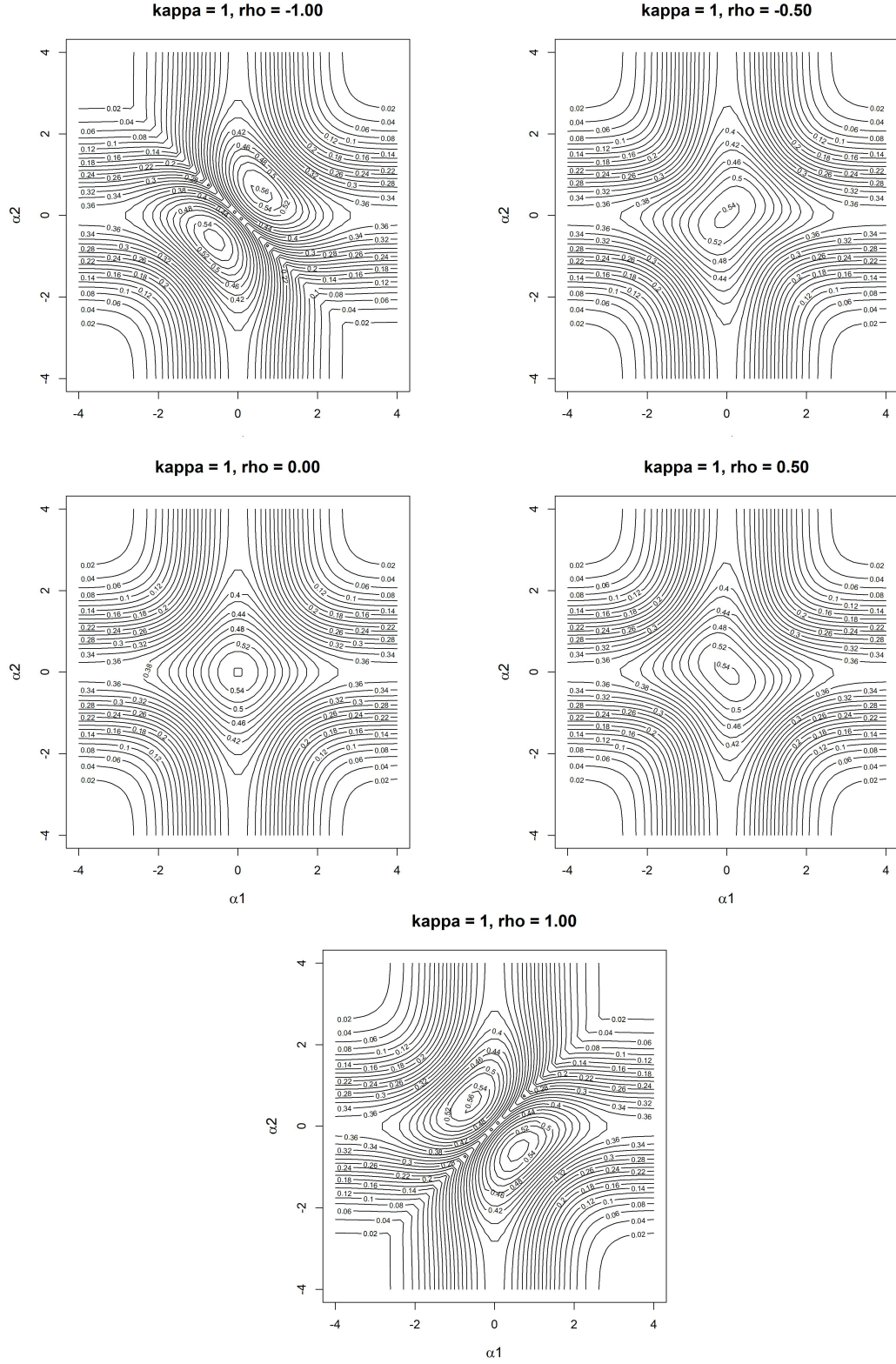


FIGURE B.1. Representation of the level sets of H (with $\kappa = 1$) on a grid of size 50×50 for $\alpha := (\alpha_1, \alpha_2) \in [-4, 4]$, and for different values of ρ : $\rho = -1$, $\rho = -0.5$, $\rho = 0$, $\rho = 0.5$, and $\rho = 1$.

Interpolation of H on $[-50, 50]^2 \times [-1, 1]$

Recall that our objective is to estimate the function $H : (\alpha_1, \alpha_2, \rho) \mapsto \int_0^\kappa F_{v_x}(t) dt$ over its entire domain, in order to reduce the computation time needed to evaluate Vector Bichon criterion. With this in mind, and given the observed regularity of H (see Figure B.1), we opt for an interpolation method, using the *ipol* routine from the *chebpol* package [30], while exploiting the symmetries of H previously highlighted.

The interpolation methods compared are: multilinear interpolation (see, for example, [61]), the Floater–Hormann method [24] which relies on the construction of pole-free barycentric rational interpolants, and the Stalker method, a spline-based method proposed in the *chebpol* package that preserves monotonicity and extrema. The Stalker method preserves any existing monotonicity or, failing that, limits overshoot, i.e., the difference between the extrema values of the interpolation points and the extrema values of the interpolation function (see [30]).

The type of grid used for interpolation plays a crucial role in the quality of the interpolation process. The distribution of grid points along the ρ component is chosen on a regular basis. For the α_1 and α_2 components, we compare a regular grid with an irregular grid obtained by a cubic transformation on each component (see the example in Figure B.2 for $(\alpha_1, \alpha_2) \in [-1, 1]^2$). The use of an irregular grid based on a cubic transformation aims to concentrate more points in the region of interest of the H function. This region of interest corresponds to high values of H , i.e., values of (α_1, α_2) not too far to the axes $\alpha_1 = 0$ and $\alpha_2 = 0$ (see Figure B.1).

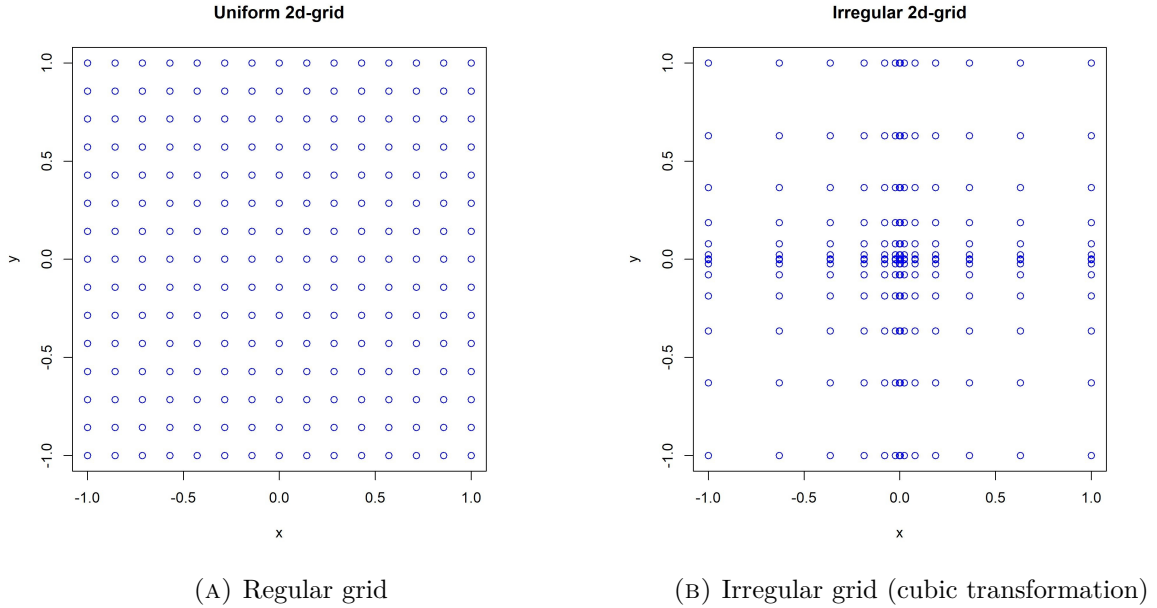


FIGURE B.2. Comparison between a regular grid and an irregular grid resulting from a cubic transformation on $[-1, 1]^2$, with 15 points per dimension.

The various interpolation methods are evaluated on $[-50, 50]^2 \times [-1, 1]$, using the two proposed grids (regular and cubic irregular), with 200 interpolation points per dimension. The results in term of interpolation errors are computed over two different sets of points: 10^5 random points selected in the area of interest $[-5, 5]^2 \times [-1, 1]$ (cf. Table B.1) and 10^5 random points selected in the global interpolation area $([-50, 50]^2 \times [-1, 1])$ (cf. Table B.2). These results indicate that the Stalker (cubic)

and Floater–Hormann (regular) methods perform best. However, the Floater–Hormann method turns out to be excessively expensive when evaluating the interpolation function at a specific point, as shown in Table B.3. Consequently, the interpolation method chosen for the H function is the Stalker (cubic) method.

Error Method	Mean	Median	$q_{0.05}$	$q_{0.95}$	Maximum
Multilinear R	$2.75e^{-3}$	$2.20e^{-3}$	$7.66e^{-5}$	$7.34e^{-3}$	$4.15e^{-2}$
Floater–Hormann R	$3.65e^{-5}$	<u>$5.27e^{-8}$</u>	<u>$9.00e^{-10}$</u>	$9.88e^{-5}$	$2.22e^{-2}$
Stalker R	$8.60e^{-4}$	$3.54e^{-4}$	$1.44e^{-5}$	$4.30e^{-3}$	$2.90e^{-2}$
Multilinear C	$2.17e^{-4}$	$1.40e^{-4}$	$8.61e^{-6}$	$6.25e^{-4}$	<u>$3.81e^{-3}$</u>
Floater–Hormann C	$7.82e^{15}$	$2.94e^{13}$	$1.03e^{10}$	$4.34e^{15}$	$2.00e^{19}$
Stalker C	<u>$1.03e^{-5}$</u>	$7.47e^{-6}$	$4.95e^{-7}$	<u>$2.70e^{-5}$</u>	$5.93e^{-3}$

TABLE B.1. Absolute errors of different interpolation methods chosen with 200 interpolation points per dimension and for 10^5 points chosen randomly on $[-5, 5]^2 \times [-1, 1]$. R refers to a regular grid and C to a cubic grid.

Error Method	Mean	Median	$q_{0.05}$	$q_{0.95}$	Maximum
Multilinear R	$3.34e^{-4}$	0	0	$2.74e^{-3}$	$2.68e^{-2}$
Floater–Hormann R	<u>$1.30e^{-6}$</u>	$3.00e^{-10}$	0	<u>$1.35e^{-6}$</u>	$7.80e^{-3}$
Stalker R	$1.63e^{-4}$	0	0	$5.02e^{-4}$	$1.89e^{-2}$
Multilinear C	$2.62e^{-5}$	0	0	$1.98e^{-4}$	$1.35e^{-3}$
Floater–Hormann C	$1.33e^{21}$	$1.53e^{18}$	$5.37e^{14}$	$3.57e^{20}$	$9.83e^{24}$
Stalker C	$1.36e^{-6}$	0	0	$1.08e^{-5}$	<u>$1.24e^{-4}$</u>

TABLE B.2. Absolute errors of different interpolation methods chosen with 200 interpolation points per dimension and for 10^5 points chosen randomly on $[-50, 50]^2 \times [-1, 1]$. R refers to a regular grid and C to a cubic grid.

Interpolation of H outside $[-50, 50] \times [-1, 1]$

By default, the *chebpol* package extends the interpolation function beyond the domain boundary using multilinear interpolation. Tests on points outside or on the boundary show very low errors, particularly when the values of the H function are high. Given these results, we retain the default multilinear interpolation method for extrapolation.

Method \ Grid	Multilinear	Floater–Hormann	Stalker	True evaluation
regular	$2.29e^{-2}s$	3min 12s	$8.95e^{-2}s$	3min 29s
cubic	$6.95e^{-3}s$	2min 55s	$1.86e^{-2}s$	3min 29s

TABLE B.3. Comparison of interpolation function evaluation computation times for 10^4 randomly selected points, between the different interpolation methods selected.

Appendix C. Convergence checks for the NSGA-II Pareto front

This appendix provides convergence indicators for the NSGA-II computation of the Pareto front associated with the multi-objective problem in Equation (3.5). These indicators follow standard COCO benchmarking practice for multiobjective algorithms [36, 58]. For a fixed initial DoE and fixed random seeds, we compare the fronts obtained with the article parameters (150 generations, population size 100) to those obtained with `generations` and `popsiz` both multiplied by 2, at different moments of the enrichment (`iteration` = 0, 100, 200). The fronts are relatively close (slightly more dispersion at 100 and 200), which is sufficient for selecting a robust compromise point in our setting.

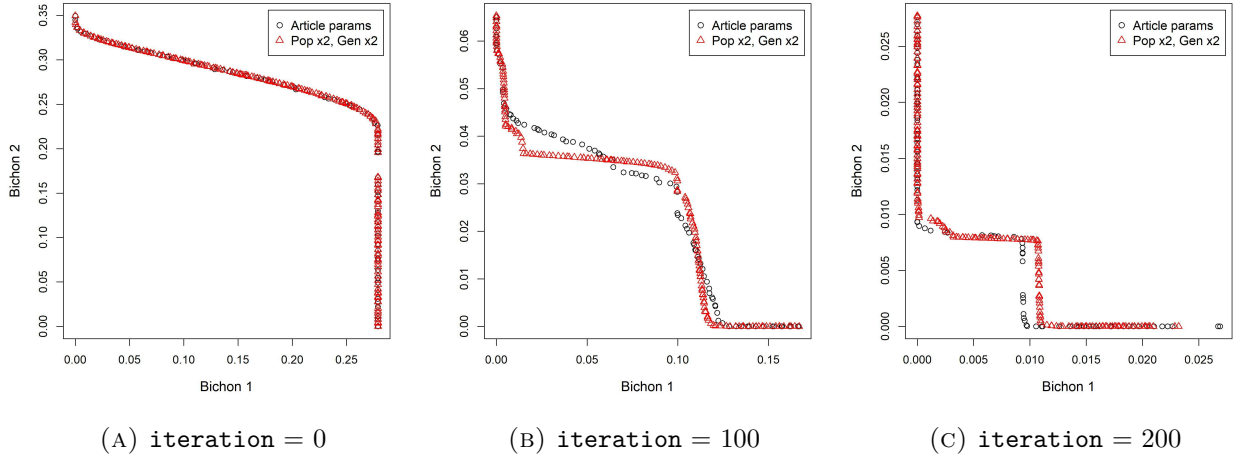


FIGURE C.1. Pareto fronts computed with NSGA-II for a fixed initial DoE and fixed seeds, using three different values of `iteration`. Article parameters: 150 generations, population size 100.

In addition, we summarize hypervolume statistics over 10 seeds for each configuration. The hypervolume indicator measures the volume of objective space dominated by the estimated Pareto front with respect to a fixed reference point [25]. The same hypervolume reference point is used across all cases. These statistics show limited variability across seeds, consistent with the qualitative front comparisons.

Case \ HV	Mean	SD	Min	Max
<code>iteration</code> = 0, Default	$7.958e^{-2}$	$1.326e^{-5}$	$7.956e^{-2}$	$7.960e^{-2}$
<code>iteration</code> = 0, Double	$7.978e^{-2}$	$6.733e^{-6}$	$7.977e^{-2}$	$7.980e^{-2}$
<code>iteration</code> = 100, Default	$4.863e^{-3}$	$4.065e^{-4}$	$4.022e^{-3}$	$5.200e^{-3}$
<code>iteration</code> = 100, Double	$4.583e^{-3}$	$6.411e^{-4}$	$3.948e^{-3}$	$5.425e^{-3}$
<code>iteration</code> = 200 Default	$1.046e^{-4}$	$1.717e^{-5}$	$7.696e^{-5}$	$1.275e^{-4}$
<code>iteration</code> = 200 Double	$1.140e^{-4}$	$1.842e^{-5}$	$8.914e^{-5}$	$1.421e^{-4}$

TABLE C.1. Hypervolume summary over 10 seeds for each configuration. The comparison includes the article parameters (Default) versus **generations** and **popsiz**e multiplied by 2 (Double). Mean, standard deviation (SD), minimum, and maximum are reported.

Acknowledgements

This research was funded by the joint Inria/IFPEN laboratory “HPC/AI/HPDA Convergence for the Energy Transition” (High-performance Computing / Artificial Intelligence / High-Performance Data Analysis). The authors would like to thank Nicolas Guy, Morgane Menz and Jean-Lou Pfister of IFP Énergies Nouvelles for their valuable contribution to the application relating to the calibration of the wind turbine numerical simulator, discussed in Section 5. The calculations presented in this document were carried out using the GRICAD infrastructure (<https://gricad.univ-grenoble-alpes.fr>), supported by Grenoble research communities. This work has benefited from a French government grant managed by the Agence Nationale de la Recherche under the France 2030 program, reference ANR-23-IACL-0006. We would also like to express our gratitude to the partners of the CIROQUO consortium (<https://doi.org/10.5281/zenodo.6581216>), as well as to the members of the [18] thesis jury for their constructive exchanges and helpful comments.

References

- [1] Nrel OpenFAST Documentation Release v2. 3.0, 2020. National Renewable Energy Laboratory: Golden, CO, USA.
- [2] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with formulas, graphs, and mathematical tables*, volume 55 of *National Bureau of Standards, Applied Mathematics Series*. U.S. Department of Commerce, 1966.
- [3] Mauricio A. Alvarez, Lorenzo Rosasco, Neil D. Lawrence, et al. Kernels for vector-valued functions: A review. *Found. Trends Mach. Learn.*, 4(3):195–266, 2012.
- [4] Dario Azzimonti and David Ginsbourger. Estimating orthant probabilities of high-dimensional Gaussian vectors with an application to set estimation. *J. Comput. Graph. Stat.*, 27(2):255–267, 2018.
- [5] Christian Bak, Frederik Zahle, Robert Bitsche, Taeseong Kim, Anders Yde, Lars Christian Henriksen, Morten Hartvig Hansen, Jose Pedro Albergaria Amaral Blasques, Mac Gaunaa, and Anand Natarajan. The DTU 10-MW reference wind turbine, 2013. Danish wind power research 2013.
- [6] Julien Bect, David Ginsbourger, Ling Li, Victor Picheny, and Emmanuel Vazquez. Sequential design of computer experiments for the estimation of a probability of failure. *Stat. Comput.*, 22(3):773–793, 2012.

- [7] Yoshua Bengio, Ian Goodfellow, and Aaron Courville. *Deep learning. Vol. 1*. MIT Press, 2017.
- [8] Barron J. Bichon, Michael S. Eldred, Laura Painton Swiler, Sandaran Mahadevan, and John M. McFarland. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA J.*, 46(10):2459–2468, 2008.
- [9] Edwin V. Bonilla, Kian Chai, and Christopher Williams. Multi-task Gaussian process prediction. In *Proceedings of the 21st International Conference on Neural Information Processing Systems*, pages 153–160. Curran Associates, Inc., 2007.
- [10] Mohamed Amine Bouhlel, Nathalie Bartoli, Abdelkader Otsmane, and Joseph Morlier. Improving kriging surrogates of high-dimensional design models by Partial Least Squares dimension reduction. *Struct. Multidiscip. Optim.*, 53(5):935–952, 2016.
- [11] Rune Brincker and Carlos Ventura. *Introduction to operational modal analysis*. John Wiley & Sons, 2015.
- [12] Ambroise Cadoret. *Analyse modale opérationnelle pour le suivi de santé structurelle des éoliennes*. PhD thesis, Université de Rennes, 2023.
- [13] Russel E. Caflisch. Monte carlo and quasi-monte carlo methods. *Acta Numer.*, 7:1–49, 1998.
- [14] Pierre Dagnelie. *Statistique théorique et appliquée: Les bases théoriques*. Les Presses Agronomiques de Gembloux, 1992.
- [15] Guillaume Damblin, Mathieu Couplet, and Bertrand Iooss. Numerical studies of space-filling designs: optimization of Latin Hypercube Samples and subprojection properties. *J. Simul.*, 7(4):276–289, 2013.
- [16] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.*, 6(2):182–197, 2002.
- [17] Yves Deville, David Ginsbourger, Olivier Roustant Contributors, Nicolas Durrande, and Maintainer Olivier Roustant. Package ‘kergp’, 2015.
- [18] Clément Duhamel. *Gaussian processes-based excursion set estimation for scalar or vector black box functions. Application to the calibration of a numerical wind turbine simulator*. PhD thesis, Université Grenoble Alpes, 2024.
- [19] Clément Duhamel, Céline Helbert, Miguel Munoz Zuniga, Clémentine Prieur, and Delphine Sinoquet. A SUR version of the Bichon criterion for excursion set estimation. *Stat. Comput.*, 33(2): article no. 41 (17 pages), 2023.
- [20] Delphine Dupuy, Céline Helbert, and Jessica Franco. DiceDesign and DiceEval: Two R packages for design and analysis of computer experiments. *J. Stat. Softw.*, 65(11):1–38, 2015.
- [21] Christophe Dutang and Petr Savicky. randtoolbox: Generating and testing random numbers, 2013. R package.
- [22] Benjamin Echard, Nicolas Gayton, and Maurice Lemaire. AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. *Struct. Saf.*, 33(2):145–154, 2011.
- [23] Mohamed Reda El Amri, Céline Helbert, Olivier Lepreux, Miguel Munoz Zuniga, Clémentine Prieur, and Delphine Sinoquet. Data-driven stochastic inversion via functional quantization. *Stat. Comput.*, 30:525–541, 2020.
- [24] Michael S. Floater and Kai Hormann. Barycentric rational interpolation with no poles and high rates of approximation. *Numer. Math.*, 107:315–331, 2007.
- [25] Carlos M. Fonseca, Luis Paquete, and Manuel Lopez-Ibanez. An improved dimension-sweep algorithm for the hypervolume indicator. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1157–1163. IEEE Press, 2006.
- [26] Trygve Olav Fossum, Cédric Travelletti, Jo Eidsvik, David Ginsbourger, and Kanna Rajan. Learning excursion sets of vector-valued Gaussian random fields for autonomous ocean sampling. *Ann. Appl. Stat.*, 15(2):597–618, 2021.

- [27] Jerome H. Friedman. Greedy function approximation: a gradient boosting machine. *Ann. Stat.*, 29(5):1189–1232, 2001.
- [28] Zhi-Fang Fu and Jimin He. *Modal analysis*. Elsevier, 2001.
- [29] Clément Gauchy. *Uncertainty quantification methodology for seismic fragility curves of mechanical structures: Application to a piping system of a nuclear power plant*. PhD thesis, Institut Polytechnique de Paris, 2022.
- [30] Simen Gaure. Details of chebpol, 2013. <http://download.nust.na/pub3/cran/web/packages/chebpol/vignettes/chebpol.pdf>.
- [31] Alan Genz. Numerical computation of rectangular bivariate and trivariate normal and t probabilities. *Stat. Comput.*, 14:251–260, 2004.
- [32] David Ginsbourger. Sequential design of computer experiments. In *Wiley StatsRef: Statistics Reference Online*, pages 1–9. John Wiley & Sons, 2017.
- [33] David Ginsbourger, Delphine Dupuy, Anca Badea, Laurent Carraro, and Olivier Roustant. A note on the choice and the estimation of kriging models for the analysis of deterministic computer experiments. *Appl. Stochastic Models Bus. Ind.*, 25(2):115–131, 2009.
- [34] Pierre Goovaerts. *Geostatistics for natural resources evaluation*. Oxford University Press, 1997.
- [35] Nico S. Gorbach, Andrew An Bian, Benjamin Fischer, Stefan Bauer, and Joachim M. Buhmann. Model selection for gaussian process regression. In *Pattern Recognition: 39th German Conference, GCPR 2017, Basel, Switzerland, September 12–15, 2017, Proceedings 39*, pages 306–318. Springer, 2017.
- [36] Nikolaus Hansen, Anne Auger, Raymond Ros, Olaf Mersmann, Tea Tušar, and Dimo Brockhoff. Coco: A platform for comparing continuous optimizers in a black-box setting. *Optim. Methods Softw.*, 36(1):114–144, 2021.
- [37] Céline Helbert, Delphine Dupuy, and Laurent Carraro. Assessment of uncertainty in computer experiments from Universal to Bayesian Kriging. *Appl. Stochastic Models Bus. Ind.*, 25(2):99–113, 2009.
- [38] Jeffrey D. Helterbrand and Noel Cressie. Universal cokriging under intrinsic coregionalization. *Math. Geol.*, 26:205–226, 1994.
- [39] Ruichen Jin, Wei Chen, and Agus Sudjianto. On sequential sampling for global metamodeling in engineering design. In *Proceedings of the ASME 2002 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Volume 2: 28th Design Automation Conference*, pages 539–548. American Society of Mechanical Engineer, 2002.
- [40] Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *J. Glob. Optim.*, 13(4):455–492, 1998.
- [41] Haitao Liu, Jianfei Cai, and Yew-Soon Ong. Remarks on multi-output Gaussian process regression. *Knowledge-Based Syst.*, 144:102–121, 2018.
- [42] R. Timothy Marler and Jasbir S. Arora. Survey of multi-objective optimization methods for engineering. *Struct. Multidiscip. Optim.*, 26:369–395, 2004.
- [43] M. D. McKay, R. J. Beckman, and W. J. Conover. Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*, 21(2):239–245, 1979.
- [44] Jorge J. Moré and Stefan M. Wild. Benchmarking derivative-free optimization algorithms. *SIAM J. Optim.*, 20(1):172–191, 2009.
- [45] Harald Niederreiter. *Random number generation and quasi-Monte Carlo methods*. Society for Industrial and Applied Mathematics, 1992.
- [46] Anthony O’Hagan. Curve fitting and optimal design for prediction. *J. R. Stat. Soc., Ser. B, Stat. Methodol.*, 40(1):1–24, 1978.

- [47] Michael Osborne. Gaussian processes for prediction, 2007. Technical Report PARG-07-01, Robotics Research Group, Department of Engineering Science, University of Oxford.
- [48] Christopher Joseph Paciorek. *Nonstationary Gaussian processes for regression and spatial modelling*. PhD thesis, Carnegie Mellon University, 2003.
- [49] Julien Pelamatti, Rodolphe Le Riche, Céline Helbert, and Christophette Blanchet-Scalliet. Coupling and selecting constraints in Bayesian optimization under uncertainties. *Optim. Eng.*, 25(1):373–412, 2024.
- [50] Victor Picheny, David Ginsbourger, Olivier Roustant, Raphael T. Haftka, and Nam-Ho Kim. Adaptive designs of experiments for accurate approximation of a target region. *J. Mech. Des.*, 132(7): article no. 071008 (9 pages), 2010.
- [51] Victor Picheny, Tobias Wagner, and David Ginsbourger. A benchmark of kriging-based infill criteria for noisy optimization. *Struct. Multidiscip. Optim.*, 48(3):607–626, 2013.
- [52] Michael J. D. Powell. Direct search algorithms for optimization calculations. *Acta Numer.*, 7:287–336, 1998.
- [53] Carl Edward Rasmussen and Christopher Williams. *Gaussian processes for machine learning*. MIT Press, 2006.
- [54] Edwin Reynders. System identification methods for (operational) modal analysis: review and comparison. *Arch. Comput. Methods Eng.*, 19:51–124, 2012.
- [55] Olivier Roustant, David Ginsbourger, and Yves Deville. DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. <https://hal.science/hal-00495766/>, 2012.
- [56] Nidamarthi Srinivas and Kalyanmoy Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput.*, 2(3):221–248, 1994.
- [57] Heike Trautmann, Detlef Steuer, and Olaf Mersmann. Package ‘mco’, 2013.
- [58] Tea Tušar, Dimo Brockhoff, Nikolaus Hansen, and Anne Auger. Coco: the bi-objective black box optimization benchmarking (bbob-biobj) test suite. **This specific title doesn’t exist on the arxiv website. Many similar entries exist, but the closest isn’t of the same year nor the same authors. Which one it is?**, 2016.
- [59] Peter Van Overschee and Bart De Moor. *Subspace identification for linear systems: Theory–Implementation–Applications*. Springer, 2012.
- [60] T. L. Vincent and W. J. Grantham. *Optimality in Parametric Systems*. John Wiley & Sons, 1981.
- [61] Rick Wagner. Multi-linear interpolation, 2008. Beach Cities Robotics.
- [62] Dongbin Xiu and George Em Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(2):619–644, 2002.