Available online at www.eccomasproceedia.org
Eccomas Proceedia UNCECOMP (2025) 56-71

ECCOMAS
Proceedia

UNCECOMP 2025
6th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Science and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Rhodes Island, Greece, 15-18 June 2025

BAYESIAN ROBUST DESIGN OPTIMIZATION: AN EXTREME VALUE APPROACH

Augustin Persoons[1], Pascal Lafon[1], and David Moens[2,3]

[1]University of Technology of Troyes
12 Rue marie Curie, Troyes 10010, France
e-mail: augustin.persoons@utt.fr

[2] KU Leuven, Department of Mechanical Engineering, LMSD Div.
Jan De Nayerlaan 5, 2860, Sint-Katelijne-Waver, Belgium

[3] Flandersmake@KULeuv en

Abstract. *This work deals with the problem of robust design optimization under interval uncertainty. The proposed approach falls in the framework of Bayesian optimization techniques relying on Gaussian process regression models as a surrogate of the expensive performance function. The main difficulty resides in the problem at hand being two nested optimization problems. A direct, nested implementation of Bayesian optimization methods results in sub-optimal constraints in the selected calibration data points. The proposed method consists in approximating the stochasticity of the optimal of the inner problem using extreme value distributions. This approach allows us to consider an alternative stochastic process associated directly with the outer optimization problem and on which Bayesian optimization can be applied. While this approach does not circumvent the nested nature of the problem, the Bayesian optimization is limited to the outer problem and the generated calibration dataset is therefore not locked to any specific nested shape. The methods exhibits promising performances on the simple examples it has been tested on.*

Keywords: Bayesian optimization, Robust design optimization, Interval uncertainty, Extreme value distribution.

ISSN:2623-3339
Peer-review under responsibility of the organizing committee of UNCECOMP 2025.
doi: 10.7712/120225.12340.21353

1 Introduction and statement of the problem

Consider an engineering design optimization problem involving a system which depends on two sets on variables. First a set of design parameters $x \in \mathbb{R}^{n_x}$ that can be chosen in a deterministic manner within a design space I_x . A second set of uncertain parameters $u \in \mathbb{R}^{n_u}$ that will take an arbitrary value in an interval I_u. The performance y of the system can be evaluated by the means of numerical model g such that $\forall (x, u) \in (I_x, I_u),\ y = g(x, u)$. Such a problem can for instance occur when limited data of the uncertainty variables is available and no distribution can be reasonably inferred from them.

This work focuses on design optimization in the context of an objective function formulated as a robustness metric. The notion of robustness, often credited to Taguchi [1], refers to a system's ability to maintain stable performance despite variations in operating conditions. Over time, various definition of robustness metrics have been developed to suit different contexts, depending largely on the type of uncertainty being addressed. Some approaches rely on stochastic formulations, such as those based on statistical moments [2, 3, 4], while others adopt deterministic perspectives [5, 6, 7, 8]. The deterministic approach, which is particularly appropriate for handling interval uncertainties, is the one adopted in this study.

This work will focus on only one aspect of the robustness optimization problem, which is the minimization of the variation of performance. The second problem that will not be considered is the optimization of the nominal performance, in a similar fashion as deterministic design optimization. The firs one is considered the most difficul and the proposed approach is a firs step toward a weighted multi-objective optimization method.

The goal will therefore be to fin the optimal design x^* minimizing the variation in performance propagated from the uncertain variables. The optimal design can then be define as:

$$x^* = \underset{x \in I_x}{\operatorname{argmin}} \left(\max_{u \in I_u} g(x, u) - \min_{u \in I_u} g(x, u) \right) \tag{1}$$

A robustness indicator Δy can be conveniently define as:

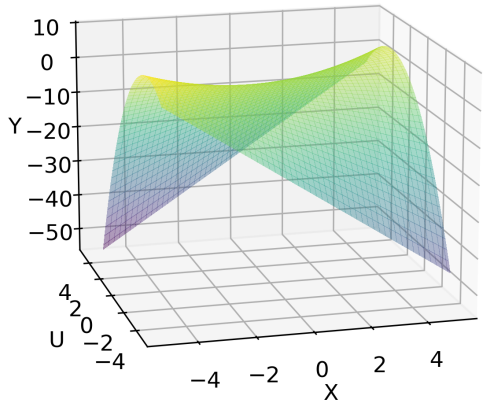
$$\forall x \in I_x,\ \Delta y(x) = \max_{u \in I_u} g(x, u) - \min_{u \in I_u} g(x, u) \tag{2}$$

The problem is illustrated on a saddleback-like 2-D function in Figure1. Figure1a illustrates the performance surface as a function of the design x and uncertain u variables. Figure1b illustrates the projection of the surface on the x space, highlighting the performance amplitude associated with each design as the vertical amplitude of the blue area. Figure1c finall illustrates the robustness indicator Δ_y associated with each design as well as the optimal design x^*.

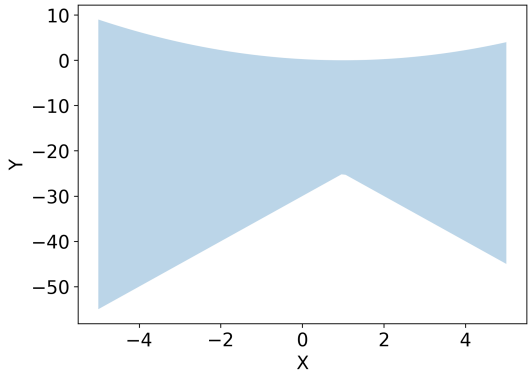
In this work, the Bayesian optimization framework is firs briefl introduced and the diffi culties in applying it to the problem at hand is highlighted. The heuristic approach based on extreme value distribution that is proposed in this work is then be detailed as well as the algorithm associated with it. The performance of the method is then illustrated on two analytical functions and a finit element model and the results are discussed. The paper finall concludes with a discussion, perspectives for future developments currently lacking and a link to the GitLab repository of the method is provided with a brief description of its content.

2 Bayesian optimization approach

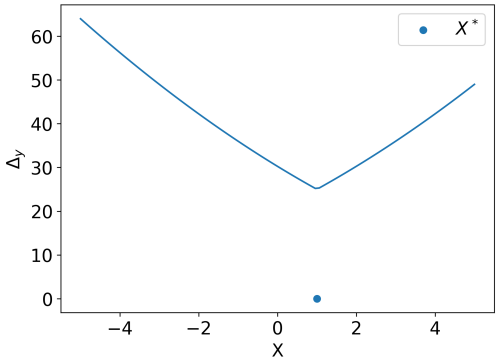
Bayesian optimization refers to a class of numerical optimization methods aimed tackling optimization problems where the objective function is computationally costly to evaluate (e.g.



(a) Illustrative behavior surface



(b) Projection of the output performance on the design axis



(c) Robustness metric Δy as a function of the design x

Figure 1: Illustration of a dummy robust design optimization problem and its solution

a numerical model). Their principle consists in replacing the objective function with a surrogate $\hat{g}$ that is less computationally expensive to evaluate, see e.g. [9]. The tradeoff of such surrogate-based methods is the extra approximation introduced by the prediction error of the surrogate. This error is related to the quality of the surrogate calibration and, in turn, to the size of the calibration dataset. The goal of efficien surrogate methods is to defin a semi-optimal calibration dataset as a compromise between the the number of evaluation of the objective function and the error induced to the quantity of interest. Popular methods in the literature use a so-called active-learning approach aiming a building a near-optimal design of experiment iteratively [10, 11, 12]. They take advantage of the stochastic nature of the prediction provided by Gaussian process regression models (GPR), using the variance of the surrogate as an indicator of the local prediction error. The next optimal location to add to the dataset is the one identifie (usually through a heuristic) as the location where the local error contributes the most to the error of the quantity of interest.

One of the most popular Bayesian optimization method is the efficien global optimization methods (EGO) [13]. Which proposed a heuristic for selecting the coordinates to refin in the shape of the expected improvement. Let us defin y_{min}, the scalar being the current best

estimate of the minimum of the objective function.

$$y_{min} = \min_{x} \hat{g}(x) \tag{3}$$

The expected improvement is define as:

$$\begin{aligned} \forall x \in I_x, EI(x) &= E\left[max(y_{min} - \hat{g}(x)\,,\,0)\right] \\ &= F(y_{min} \mid x) E(y_{min} - \hat{g}(x) \mid \hat{y}_y(x) \leq \hat{g}_{y\,min}) \\ &= \int_{-\infty}^{y_{min}} (y_{min} - \tau)\, f(\tau \mid x)\, d\tau \end{aligned} \tag{4}$$

with F and f respectively the cumulative and probability density functions of $\hat{g}(x)$. Following the EGO algorithm, the optimal design coordinate of the next point to add the calibration dataset x^* is define as:

$$x^* = \underset{x \in I_x}{\operatorname{argmax}}\, EI(x) \tag{5}$$

Several approaches are possible to apply Bayesian optimization in solving the nested problem define in equation1. A firs approach could be to solve the outer problem as a deterministic one and the inner problems with Bayesian optimization. The main limitation is that the choice of outer location investigated by the optimization algorithm locks the method into significan calibration effort to turn this location into a deterministically known value, before jumping to another one. This imposes a grid-like shape to the generated calibration data and significan inefficien y.

Some work have been proposed in the Bayesian optimization literature for related min-max problems where two nested optimization problems need to be solved. Authors have proposed a straightforward approach consisting in implementing two nested EGO algorithms considering two distinct GPRs [14]. A more promising framework is to solve directly the outer problem with Bayesian optimization, the difficult then becomes to propagate the inner uncertainty to this outer problem. Authors have proposed approaches to do so through a worst-case scenario based on the GPR statistical moments [15] or through confidenc intervals [16, 17]. This work proposes an new approach falling into this category.

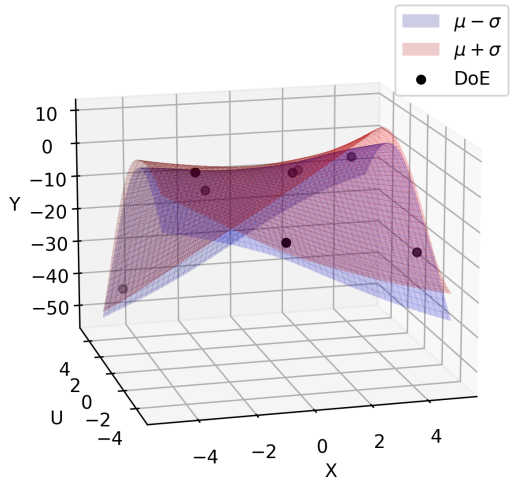
Lets us defin the optimization problem considering a GPR surrogate $\hat{g}$ of the performance. The surrogate counterpart of the robustness indicator Δy define in equation 2 then becomes:

$$\forall x \in I_x,\ \Delta\hat{y}(x) = \max_{u \in I_u} \hat{g}(x,u) - \min_{u \in I_u} \hat{g}(x,u) \tag{6}$$

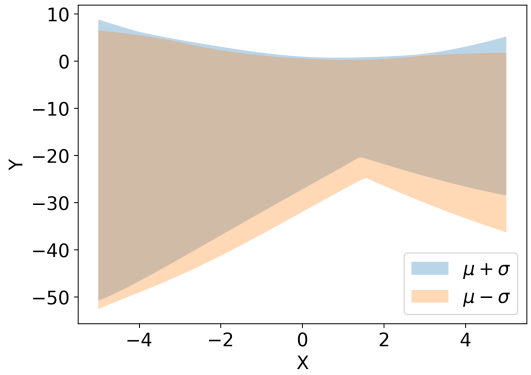
Let us then defin $\bar{\hat{g}}$ and $\underline{\hat{g}}$ the two stochastic processes define as:

$$\forall x \in I_x, \begin{cases} \bar{\hat{g}}(x) = \max\limits_{u \in I_u} \hat{g}(x,u) \\ \underline{\hat{g}}(x) = \min\limits_{u \in I_u} \hat{g}(x,u) \end{cases} \tag{7}$$

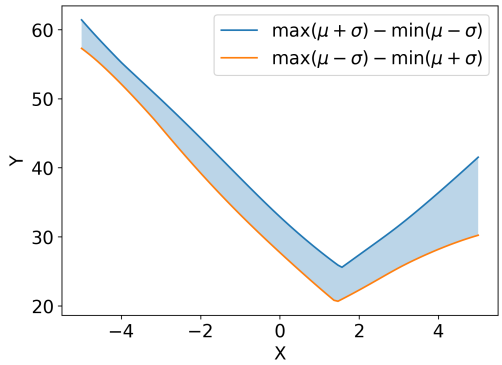
The robustness indicator associated with any design $\Delta\hat{y}(x)$ is a random variable, define as the difference between the two extrema of the Gaussian stochastic process $\hat{g}(x,u)$, $u \in I_u$. Its stochastic nature is illustrated in Figure 2 using the same dummy function as presented in Figure 1. Figure 2a illustrates the two performance surfaces associated with the mean prediction of the GPR plus and minus its standard deviation. When projected on the design space as in Figure 2b these surfaces highlight the stochastic nature of the performance amplitude. Finally,



(a) Kriging prediction surface $\mu_{x,u}$ and the confi dence interval surfaces $\mu_{x,u} \pm \sigma_{x,u}$



(b) Projection of Kriging estimated performances on the design axis



(c) Stochastic robustness metric $\hat{\Delta}y$ and its confi dence interval as a function of the design

Figure 2: Illustration of a dummy robust design optimization problem associated with a GPR surrogate of the system

the variance of the robustness metric $\Delta\hat{y}$ is illustrated in Figure 2c by representing its upper and lower bounds associated with the two surfaces.

A major difficult of implementing a Bayesian optimization procedure to minimize Δy resides in estimating the statistics of $\Delta\hat{y}(x)$ with an affordable computationburden. As previously mentioned, several methods have been proposed to do so, through worst-case scenario or confidenc intervals approaches. This work does not aim at estimating precisely the statistics of the robustness indicator, instead a heuristic is developed based on the statistics of extreme value distributions.

3 A heuristic based on extreme value distributions

It should be stressed that estimating the statistics of $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$ is a very challenging task. It consists in estimating, for any design x, the statistics of both extrema of a non-stationary Gaussian process (the GPR predictive posterior) over the domain I_u. This difficult and the proposed way of addressing it are further discussed in section 3.2. Another difficult resides

in estimating the local contribution of the GP variance to these statistics. They result from an aggregated contribution from all Gaussian random variables $g(x,u)$ and differentiating the contribution of each is not trivial. In essence we are interested in a quantity related to the maximum of the partial derivative of the expected improvement with respect to the variance of $\hat{g}(x,u)$, i.e. $\forall x,\ \max_u \frac{\partial EI(x|\hat{g})}{\partial var(\hat{g}(x,u))}$.

Therefore, estimating the statistics of the extrema of the of GP isn't enough, we also need to estimate their sensitivity to the local variance of the GP. In the general case of non-stationary GPs, one has to resort to numerical estimation for those statistics, which would be computationally intractable for the derivation and optimization at hand. This work proposes a heuristic of the local contribution based on the extreme value distributions associated with the Gaussian random variables $\hat{g}(x,u)$. This heuristic is aimed at being informative enough to semi-optimally guide the calibration of the surrogate while introducing a manageable computation burden.

3.1 Identifying semi-optimal coordinates in the uncertain sub-space

Let us consider, for a given design x, a Monte-Carlo population of n trajectories $\{\hat{g}_i\}_{i=1..n}$ of the GP $\hat{g}(x,u),\ u \in I_u$. We can define two Monte-Carlo populations of scalars $\{\underline{\hat{y}}_i\}_{i=1..n}$ and $\{\bar{\hat{y}}_i\}_{i=1..n}$ of the two extrema random variables $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$ in the shape:

$$\forall i \in [1,n],\ \begin{cases} \bar{\hat{y}}_i = \max\limits_u \hat{g}_i(x,u) \\ \underline{\hat{y}}_i = \min\limits_u \hat{g}_i(x,u) \end{cases} \tag{8}$$

The goal is then to identify the locations $\bar{u}^*$ and $\underline{u}^*$ for which the local random variables $\hat{g}(x,\bar{u}^*)$ and $\hat{g}(x,\underline{u}^*)$ contribute the most to the statistics of $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$ respectively. It is proposed to consider the random variables $\bar{u}$ and $\underline{u}$ defined as the arguments of the Gaussian process extrema:

$$\forall x \in I_x,\ \begin{cases} \bar{u}(x) = \operatorname{argmax}\limits_u \hat{g}(x,u) \\ \underline{u}(x) = \operatorname{argmin}\limits_u \hat{g}(x,u) \end{cases} \tag{9}$$

an intuitive way of defining $\bar{u}^*$ and $\underline{u}^*$ could be to consider the modes of the distributions of $\bar{u}$ and $\underline{u}$ respectively. This work does not aim at providing an analytical or numerical estimation of those modes, it is instead proposed to define a simple heuristic to providing to a satisfactory near-optimal and guiding the calibration of the surrogate efficientl . To introduce it, let us consider, for any u the Monte-Carlo population of performance values $\{\hat{g}_i(x,u)\}_{i=1..n}$. The extrema of this population are realizations of the random variables $\bar{\hat{g}}(x,u)$ and $\underline{\hat{g}}(x,u)$ defined respectively as the maximum and minimum of n independent identically distributed Gaussian random variables $\hat{g}(x,u)$. From the extreme value theory we know that $\bar{\hat{g}}(x,u)$ and $\underline{\hat{g}}(x,u)$ follow Gumbel distributions which parameters relate directly to mean μ and standard deviation σ of $\hat{g}(x,u)$:

$$\begin{aligned} \forall \{x,u\} \in I_x * I_u\,,\ \bar{\hat{g}}(x,u) = \max\{\hat{g}_i(x,u)\}_{i\in[1,n]} &\sim \text{Gumbel}(\bar{\nu},\bar{\beta}) \\ \bar{\nu}\,(x,u,n) &= \Phi^{-1}\left(1-\frac{1}{n}\right)\sigma + \mu \\ \bar{\beta}\,(x,u,n) &= \Phi^{-1}\left(1-\frac{1}{n}e^{-1}\right) - \bar{\nu} \end{aligned} \tag{10}$$

and:

$$\begin{aligned}
\forall\{x,u\} \in I_x * I_u\,,\ \underline{\hat{g}}(x,u) &= \min\{\hat{g}_i(x,u)\}_{i\in[1,n]} \sim \text{Gumbel}(\underline{\nu},\underline{\beta}) \\
\underline{\nu}\,(x,u,n) &= \Phi^{-1}\left(\frac{1}{n}\right)\sigma + \mu \\
\underline{\beta}\,(x,u,n) &= -\Phi^{-1}\left(\frac{1}{n}e^{-1}\right) + \underline{\nu}
\end{aligned} \tag{11}$$

with Φ^{-1} the inverse of the standard Gaussian cumulative distribution function. It is then proposed to use the expected value of $\bar{\hat{g}}(x,u)$ and $\underline{\hat{g}}(x,u)$ as heuristics for the value of the density of $\bar{u}$ and $\underline{u}$ at the location u. This choice can be seen as considering the expected value of the extremum of the population of trajectories, evaluated at the location u, as indicative of the likelihood of u being the argument of the extremum a trajectory over the domain I_u. The definition of $\bar{u}^*$ and $\underline{u}^*$ then naturally become:

$$\forall x \in I_x \begin{cases} \bar{u}^* = \underset{u\in I_u}{\text{argmax}}\, E\left(\bar{\hat{g}}(x,u)\right) \\ \underline{u}^* = \underset{u\in I_u}{\text{argmin}}\, E\left(\underline{\hat{g}}(x,u)\right) \end{cases} \tag{12}$$

or

$$\forall x \in I_x \begin{cases} \bar{u}^*(x) = \underset{u\in I_u}{\text{argmax}}\left(\bar{\nu}\,(x,u,n) + \gamma\bar{\beta}\,(x,u,n)\right) \\ \underline{u}^*(x) = \underset{u\in I_u}{\text{argmin}}\left(\underline{\nu}\,(x,u,n) - \gamma\underline{\beta}\,(x,u,n)\right) \end{cases} \tag{13}$$

with γ the Euler's constant. The two semi-optimal locations can then estimated by solving two optimization problems following equation 14 and involving one evaluation of the GPR per iteration each. It is finall proposed to only evaluate the performance function at one of those locations, i.e. the one associated with the highest GPR variance.

3.2 Identifying semi-optimal coordinates in the design sub-space

As introduced in section 2, the goal is to solve the outer optimization problem (i.e. the one define over the design sub-space) with the EGO algorithm. The challenge then resides in estimating the expected improvement associated with any design coordinate x. The outer-optimization problem can be framed as the minimization of the stochastic process $\Delta\hat{y}$ define in equation 6. Evaluating the expected improvement requires the distribution of $\Delta\hat{y}(x)$, which implies knowing the distribution of $\bar{\hat{g}}(x)$ and $\underline{\hat{g}}(x)$. Unfortunately estimating the distribution of the extrema of non-stationary Gaussian processes is notoriously difficul and require numerous numerical estimations [18], the most notorious approach being based on the Rice series [19]. It is therefore proposed to defin another heuristic for this distribution, deriving from the one introduced in section 3.1.

The principle is, for any design x, to reduce the Gaussian process $\hat{g}(x)$ to two independent random variables, i.e. the Gaussian random variables $\hat{g}(x,\bar{u}^*(x))$ and $\hat{g}(x,\underline{u}^*(x))$. It consists in reducing the continuous Gaussian process to the two random variables that contribute the most to the process extrema distributions. Doing so provides us with a simple indicator of $\Delta\hat{y}(x)$ as the difference of the two Gumbel distributions described in equations eqs. (10) and (11) at the locations $\bar{u}^*(x)$ and $\underline{u}^*(x)$ respectively:

$$\forall x \in I_x,\ \Delta\hat{y}(x) = \max\{\hat{g}_i(x, \bar{u}^*(x))\}_{i\in[1,n]} - \min\{\hat{g}_i(x, \underline{u}^*(x))\}_{i\in[1,n]} \tag{14}$$

Since the two Gumbel distributions are known, the distribution $f_{\Delta_{\hat{y}}}$ of $\Delta\hat{y}(x)$ can be estimated numerically by convolution. The expected improvement, as introduced in equation 4 can finall be estimated by numerical integration using the distribution of the the heuristic define above:

$$\forall x \in I_x, EI^{EV}(x) = \int_{-\infty}^{\Delta_{min}} (\Delta_{min} - \tau)\, f_{\hat{\Delta}_y^{EV}}(\tau \mid x)\, d\tau \tag{15}$$

With $f_{\hat{\Delta}_y}$ the distribution of the difference of the two Gumbel-distributed random variables at the locations $\bar{u}^*(x)$ and $\underline{u}^*(x)$, and Δ_{min} the current best estimate of the minimum of $\Delta_{\hat{y}}$

3.3 The active learning algorithm

The active learning algorithm employed in this work is based on the EGO algorithm, with the primary difference being that EGO is used here to identify the minimum of the (non-Gaussian) stochastic process $\hat{\Delta}_y$, as define in Equation 6. While the expected improvement criterion admits a closed-form expression in the case of a Gaussian process, an additional step is required when adapting it to robust design optimization. In this context, the additional step involves solving two optimization problems to determine the coordinates $\bar{u}(x, n)$ and $\underline{u}(x, n)$ in the uncertainty space, as outlined in Section 3.1. The full algorithm, including this extension, is summarized in the fl wchart shown in Figure 3, where the modification to the standard EGO approach are highlighted in blue.

Several components of the algorithm are treated as hyperparameters. These include the choice of kernel for the Gaussian Process Regression (GPR), the number of virtual Monte Carlo samples n introduced in Section 3.1, the strategy and size of the initial design of experiments, and the optimization methods used to identify $\bar{u}(x, n)$, $\underline{u}(x, n)$, and x^*. The present study reports preliminary results from ongoing research, and the sensitivity of the algorithm's performance to these parameters has not yet been explored. As an initial step, these choices were made arbitrarily to examine convergence behavior and performance. Specificall , a squared exponential kernel was selected, the virtual Monte Carlo population size was set to $n = 100$, and the initial design of experiments was generated using a Latin Hypercube sampling strategy with a size equal to three times the problem dimension. The optimization problems were solved using an interior point algorithm, complemented by a 10-starts distributed search strategy to improve the chances of reaching global optima.

The stopping criterion for the algorithm, in this early stage, is limited to a fi ed simulation budget: the number of simulations is not allowed to exceed a predefine threshold. Future work will explore more adaptive and efficien stopping criteria.

4 Examples and results

4.1 A benchmark function with local minima

The performance of the proposed approach is firs demonstrated using a two-dimensional test function characterized by the presence of local minima. This example is define through the following performance function g:

$$\forall x, u \in I_x, I_u, ; g(x) = xu - \sin(x)u^2 + x^2 \tag{16}$$

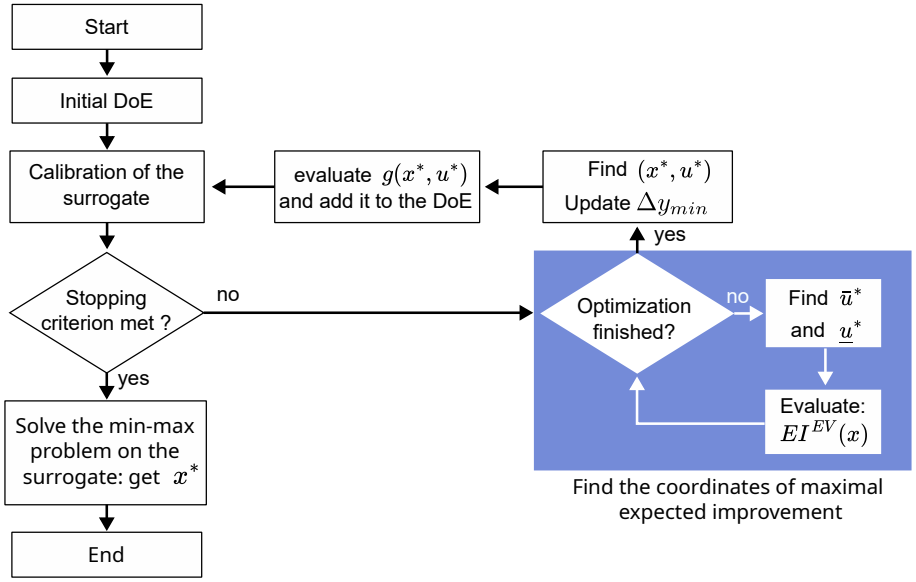
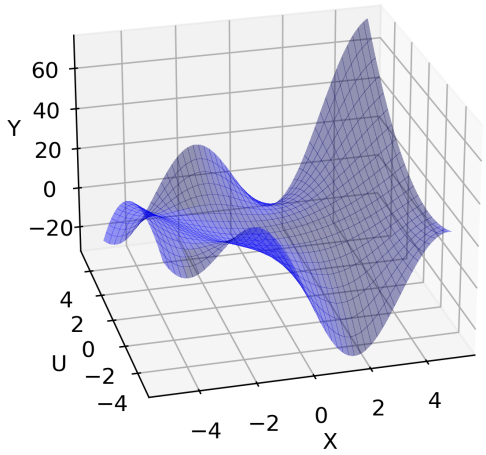
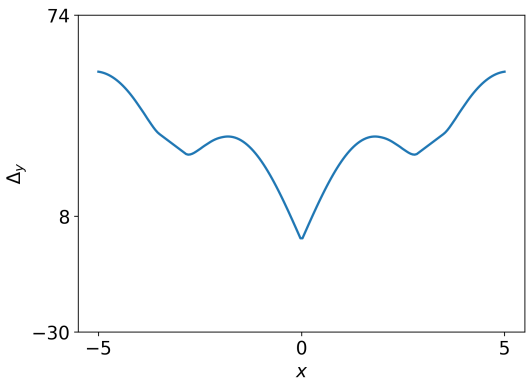


Figure 3: Flowchart of the proposed robust optimization algorithm.

The design variable x belongs to the interval $I_x = [-5, , 5]$, and the uncertain variable u is define over $I_u = [-5, 5]$. The known optimal design is located at $x^* = 0$. Although the function is relatively simple, it yields a highly non-linear robustness function with multiple local minima and non-differentiability at the optimum. This test case has previously been used in the same context to evaluate a method based on confidenc intervals, referred to as AK-RULOK, which was introduced by the authors in [16]. In that study, AK-RULOK converged to the optimal design after 30 iterations, resulting in an error of approximately 0.02 in the design coordinate. It will serve here as a baseline for comparison with the current method.



(a) Behavior surface of the firs example function



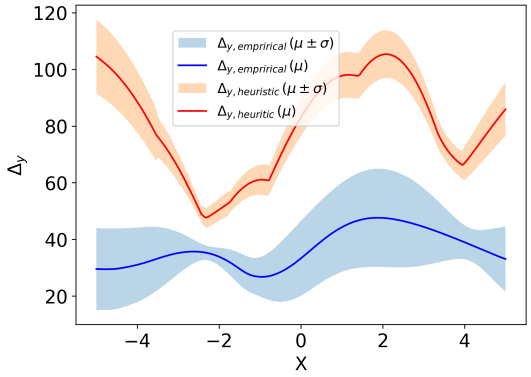
(b) Surface of the robustness metric Δ_y for the firs example function

Figure 4: Illustration of the behavior and optimal solution of the firs example function

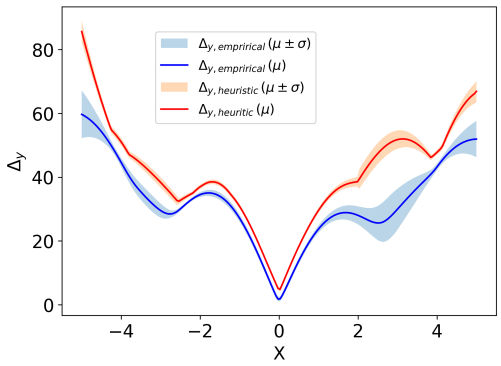
The results are firs presented in Figure 5c, which displays the convergence of the optimal

design point over the course of the optimization and the convergence amplitude over 15 repetitions of the method. The plot indicates that the algorithm successfully converges to the correct optimal value, with convergence observed after approximately 21 iterations, or 27 evaluations of the performance function, with an error interval of $[-7.4e^{-2}, 4.0e^{-2}]$. Those results suggest that the proposed approach can achieve similar or better performance than existing methods in the literature.

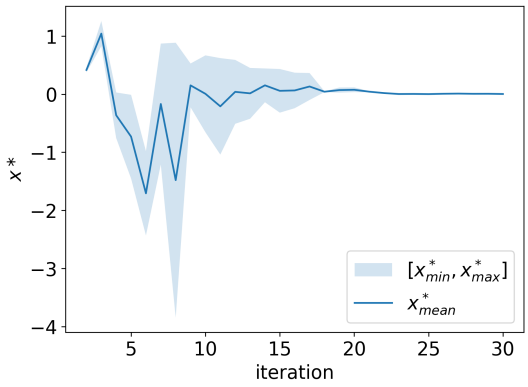
Further insight into the convergence behavior is provided in Figures 5a, 5b, which depict the evolution of the heuristic robustness metrics $\hat{\Delta}_y^{EV}$ introduced in Section 3.2. The heuristic is plotted with its mean surface and its plus/minus standard deviation envelope and is compared to empirical counterparts obtained with a Monte-Carlo population of GPR realizations. These figure correspond to iterations 2, and 20, respectively, representing different stages of convergence. Although the indicator appears to be biased relative to the robustness function estimated via the empirical GPR model, it successfully captures the overall trends of the system behavior. It should also be mentioned that a constant bias does not affect the efficien y of the method since the heuristic is only used comparatively to other heuristic values. Importantly, both the heuristic $\hat{\Delta}_y^{EV}$ and the GPR-estimated robustness function $\hat{\Delta}_y$ ultimately converge toward the same deterministic profil in the region of interest.



(a) Robustness surface metric estimated by the GP at iteration 3



(b) Robustness surface metric estimated by the GP at iteration 20



(c) Convergence of the optimal robust design coordinates compared to the true optimal

Figure 5: Illustration of the convergence of the algorithm in terms of optimal solution and robustness function approximation on the firs example

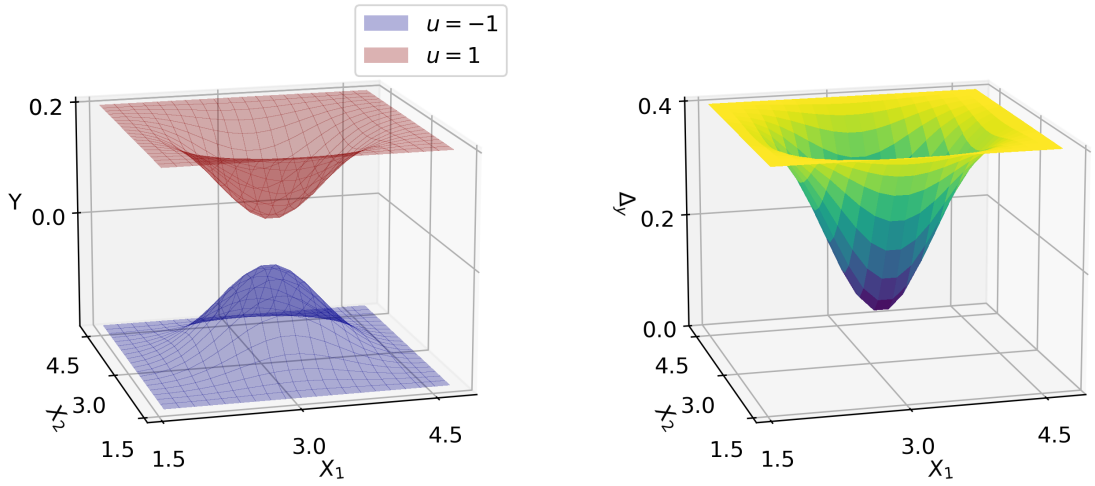
4.2 A benchmark function in dimension three

The second example involves a three-dimensional function derived from the Easom function, as define in Equation 17. In this case, the design space is two-dimensional, with the boundaries for both variables given by $I_{x_1} = I_{x_2} = [\frac{\pi}{2}, \frac{3\pi}{2}]$. The uncertain variable u is define over the interval $I_u = [-1, 1]$. Within this domain, the function is monotonic with respect to the variable u, which simplifie the interpretation of the resulting behavior surfaces. The function's shape and the corresponding optimal design coordinates are shown in Figures 6a and 6b, respectively.

$$\begin{aligned} &\forall\{x, u\} \in \{I_x.^2, I_u\}, \\ &g(x) = -cos(x_1)cos(x_2)sin\left(\frac{u}{2\pi}\right) e^{-(x_1-x_1^*)^2+(x_2-x_2^*)^2} + \frac{u}{5} \end{aligned} \tag{17}$$

with $x^* = \{x_1^*, x_2^*\} = \{\pi, \pi\}$ the optimal design.

In order to illustrate the repeatability of the convergence, the proposed method has been repeated fiftee times over forty iterations each.



(a) Behavior surface of the second example function considering two values of u

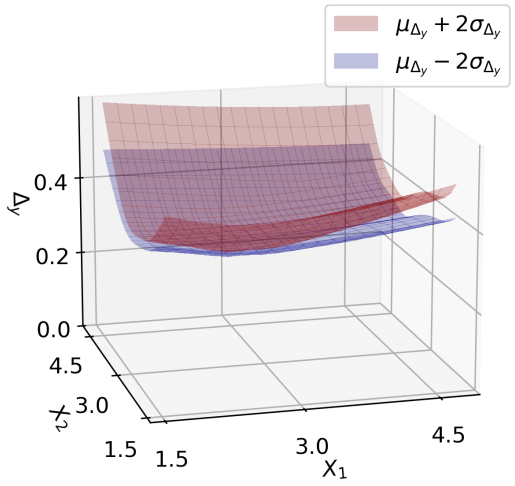
(b) Surface of the robustness metric Δ_y for the second example function

Figure 6: Illustration of the behavior and optimal solution of the second example function

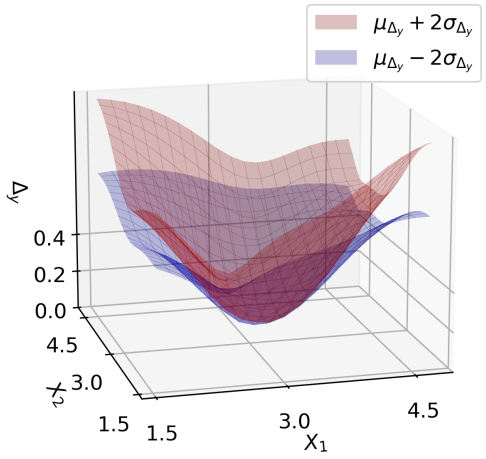
The firs results presented in Figures 7a and 7b represent the heuristic $\hat{\Delta}_y$ surface at an early iteration and a late one. The second figu e show the convergence of the heuristic metric to the actual Δ_y surface (see Figure 6b) around the optimum. The second result presented in Figure 7c is the convergence of the method over forty iterations. Both the convergence of x_1^* and x_2^* are represented in blue and orange respectively. The color surfaces represent the min and max of both coordinates over fiftee repetitions of the method. The figur highlight the fast convergence of the method with the variance over fiftee applications being visibly negligible after thirty iterations (i.e. 39 evaluations of the performance function).

4.3 A forming process finite element example

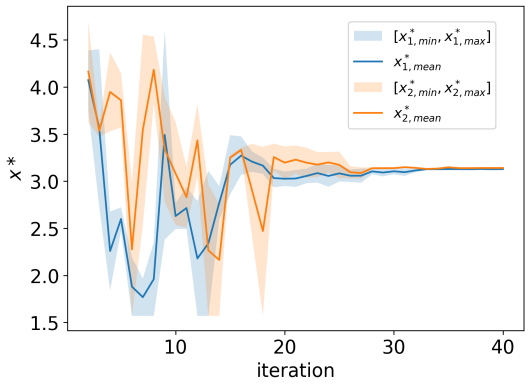
The last example is a 2D finit element model of a forming process simulating the plastic deformation of a metal sheet (DP780 steel with 1.4 mm thickness) held in contact with the die



(a) Robustness surface metric estimated by the GP at iteration 3



(b) Robustness surface metric estimated by the GP at iteration 20



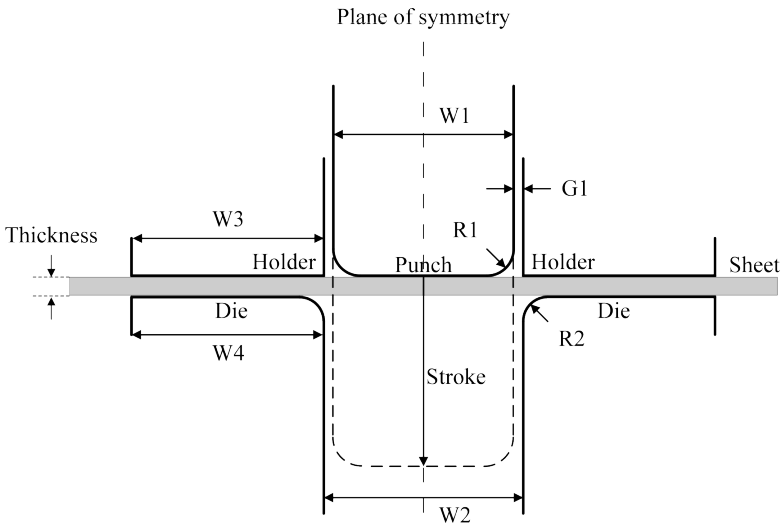
(c) Convergence of the optimal robust design coordinates compared to the true optimal

Figure 7: Illustration of the convergence of the algorithm in terms of optimal solution and robustness surface approximation on the second example

with a holder and deformed in a "U" shape by the punch, see [20]. The simulation includes the spring back after release of the elastic deformation and we are interested in the shape of the profil after spring back. The formed profil is considered dependent on only to variables R_2 and R_1, the radius of the die and punch edge rounding respectively. The Punch radius is considered as an interval uncertainty with the interval $I_{R1} = [2, 5]$. The Die radius is considered as the design parameter with a design space $I_{R1} = [1, 10]$.

The performance metric is define at the root mean square error between the nodal coordinates of the profile before and after spring back (see example in Figure 9).

The RDO problem is then to fin the die radius resulting in the most consistent profil after spring back, despite variation of the punch radius. The convergence of the algorithm is presented in Figure 10a with a fast convergence to the fina solution. The convergence plot exhibits a noisy behavior with the algorithm alternating between two solutions indicating the presence of another local minimum. Figure 10b illustrates ten profile after spring back taken form ten linearly spaced values of the die radius between $2mm$ and $5mm$, considering the optimal punch



Parameters	W1	W2	W3	W4	R1	R2	G1	Stroke
Dimensions	50.0	54.0	89.0	89.0	[2, 5]		2.0	71.8

(Unit: mm)

Figure 8: Schematic representation and dimensions of the forming process

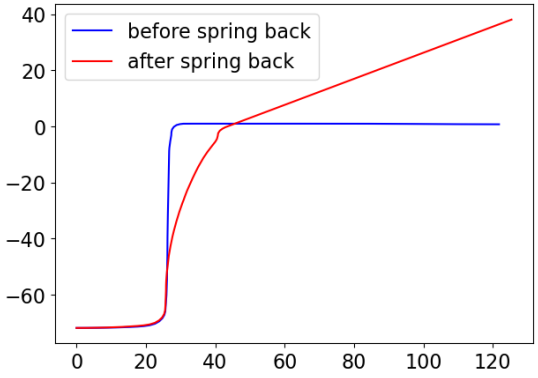
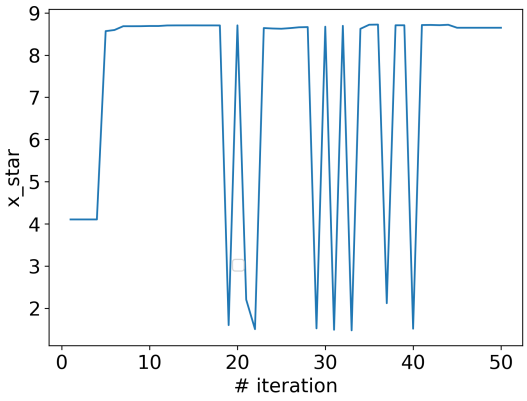


Figure 9: Illustration of the formed sheet profil before and after spring back

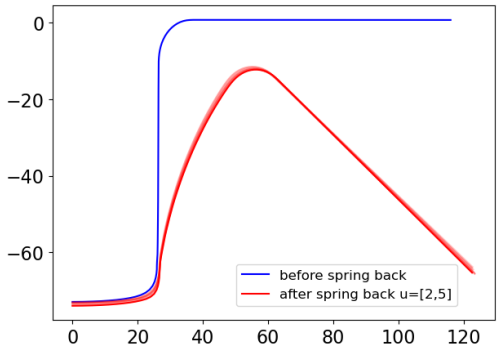
radius found by the algorithm. The proposed solution exhibits a very high spring back and its compliance with the desired shape is quite poor. However the profile are very consistent with an extremely limited influenc form the variations of the die radius. This results highlights the importance of extending this method in a multi-objective setting to fin a tradeoff between the value of the error and the amplitude of ts variations. Nevertheless, the identifie solution is very satisfactory with respect to the assigned objective, which is only to generate a profil which is robust to the variations of R_1.

5 Conclusions

This paper introduces a novel learning function along with an adaptive calibration algorithm specificall designed for robust design optimization under interval uncertainties. The core objective is to defin a heuristic distribution that guides the calibration of the Gaussian Process Regression (GPR) model through the estimation of the expected improvement function. This heuristic is intended to be computationally efficient reflect ve of the GPR-based robustness metric, and capable of highlighting the local influenc of each design variable.



(a) conergence of the algorithm over the finit element example



(b) Illustration of the solution found by the algorithm with (red) ten profile after spring back corresponding to ten values of $R_1 \in [2, 5]$

Figure 10: Illustration of results from the finit element example

The proposed approach involves generating a virtual Monte Carlo population and, for each design point, constructing the heuristic by taking the convolution of the two extreme value distributions derived from the most extreme expected values within the Monte Carlo sample.

To validate the method, it is applied to two analytical benchmark problems and a fini e element example as a proof of concept. In both analytical cases, the algorithm converges efficientl and accurately to the correct optimal design, demonstrating performance that is comparable to or slightly better than existing methods. On the finit element example, the algorithm identifie a solution which is poor in terms of industrial compliance but which does satisfie very well the robustness criterion. These initial results are promising and form a strong foundation for further investigation. Future work will include a detailed sensitivity analysis of key hyperparameters such as the size of the virtual Monte Carlo population as well as performance evaluations on a more comprehensive and reproducible benchmark set.

6 A GitLab repository

The recent work on this method focused on proposing a python version of the method shared on a GitLab repository under the address https://git.utt.fr/persoons/RDO The project has two branches using the SMT [21] and GPJax [22] toolboxes for Gaussian process calibration. The project includes:

- a module python fil *module.py* including all the functions of the method, including the estimation of the expected improvement as well as eight predefine benchmark functions
- a jupyter notebook *demo.ipynb* illustrating step by step the problem at hand and the convergence of the method.
- a jupyter notebook *demo_EI.ipynb* illustrating the calculation of the expected improvement heuristic with fast Fourier transform and graphically comparing it to empirical histograms drawn from a Monte-Carlo population.
- a python fi e *results_generation.py* allowing to run repeated applications of the method and used to generate the results presented in this paper.

REFERENCES

[1] G. Taguchi. *Quality engineering (Taguchi methods) for the development of electronic circuit technology*, IEEE Transactions on Reliability, vol. 44, no. 2, pp. 225–229, Jun. 1995.

[2] R. Jin, X. Du, W. Chen, and W. Chen, *The use of metamodeling techniques for optimization under uncertainty*, Structural and Multidisciplinary Optimization, vol. 25, no. 2, pp. 99–116, Jun. 2003

[3] J. Janusevskis and R. Le Riche. *"Simultaneous kriging-based estimation and optimization of mean response*, Journal of Global Optimization, vol. 55, no. 2, pp. 313–33, Feb. 2013.

[4] B. J. Williams, T. J. Santner, and W. I. Notz, *Sequential design of computer experiments to minimize integrated response functions*, Statistica Sinica, vol. 10, pp. 1133–1152, 2000.

[5] H.-G. Beyer and B. Sendhoff, *Robust optimization – A comprehensive survey*, Computer Methods in Applied Mechanics and Engineering, vol. 196, no. 33-34, pp. 3190–3218, Jul. 2007.

[6] C. Zang, M. Friswell, and J. Mottershead, *A review of robust optimal design and its application in dynamics*, Computers & Structures, vol. 83, no. 4-5, pp. 315–326, Jan. 2005.

[7] D. Bertsimas, D. B. Brown, and C. Caramanis, *Theory and Applications of Robust Optimization*, IAM Review, vol. 53, no. 3, pp. 464–501, Jan. 2011.

[8] B. Rustem and M. Howe, *Algorithms for worst-case design and applications to risk management*, Princeton, N.J. ; Oxford: Princeton University Press, 2002

[9] Greenhill, Stewart, Santu Rana, Sunil Gupta, Pratibha Vellanki, et Svetha Venkatesh, *Bayesian Optimization for Adaptive Experimental Design: A Review* , IEEE Access 8 (2020): 13937-48. https://doi.org/10.1109/ACCESS.2020.2966228.

[10] Gramacy, Robert B., *Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences*, Boca Raton, Florida: Chapman Hall/CRC, 2020.

[11] Jiang, Ping, Qi Zhou, et Xinyu Shao, *Surrogate Model-Based Engineering Design and Optimization. Springer Tracts in Mechanical Engineering*, Singapore: Springer Singapore, 2020. https://doi.org/10.1007/978-981-15-0731-1.

[12] Moustapha, M., S. Marelli, et B. Sudret, *A Generalized Framework for Active Learning Reliability: Survey and Benchmark*, arXiv:2106.01713 , 3 juin 2021.

[13] Jones, Donald R, et Matthias Schonlau, *Efficient Global Optimization of Expensive Black-Box Functions*, Journal of Global Optimization 13 (1998): 455 92

[14] Marzat, Julien, Eric Walter, et Hélène Piet-Lahanier, *A New Expected-Improvement Algorithm for Continuous Minimax Optimization*, Journal of Global Optimization 64, no 4 (avril 2016): 785 802. https://doi.org/10.1007/s10898-015-0344-x.

[15] Rehman, Samee ur, Matthijs Langelaar, et Fred van Keulen, *Efficient Kriging-Based Robust Optimization of Unconstrained Problems* , Journal of Computational Science 5, no 6 (novembre 2014): 872 81. https://doi.org/10.1016/j.jocs.2014.04.005.

[16] Mierlo, Conradus van, Augustin Persoons, Matthias G. R. Faes, et David Moen, *Robust Design Optimisation under Lack-of-Knowledge Uncertainty* , Computers & Structures 275 (15 janvier 2023): 106910. https://doi.org/10.1016/j.compstruc.2022.106910.

[17] Mierlo, Conradus van, Augustin Persoons, Matthias G. R. Faes, et David Moen, *Robust Design Optimization of Expensive Stochastic Simulators Under Lack-of-Knowledge*, ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg 9, no 2 (1 juin 2023): 021205. https://doi.org/10.1115/1.4056950.

[18] Azaïs, Jean-Marc, et Li-Vang Lozada-Chang, *A Toolbox on the Distribution of the Maximum of Gaussian Process*, 2013. hal-00784874

[19] Lindgren, Georg., *Gaussian Integrals and Rice Series in Crossing Distributions—to Compute the Distribution of Maxima and Other Features of Gaussian Processes*, Statistical Science 34, no 1 (1 février 2019). https://doi.org/10.1214/18-STS662.

[20] H. Huh, K. Chung, SS. Han, et WJ. Chung, *Benchmark Study of the 8th International Conference and Workshop on Numerical Simulation of 3D Sheet Metal Forming Processes*, Proceedings of Numisheet 2011

[21] P. Saves and R. Lafage and N. Bartoli and Y. Diouane and J. Bussemaker and T. Lefebvre and J. T. Hwang and J. Morlier and J. R. R. A. Martins, *SMT 2.0: A Surrogate Modeling Toolbox with a focus on Hierarchical and Mixed Variables Gaussian Processes*, Advances in Engineering Software, 2024, vol. 188, https://doi.org/10.1016/j.advengsoft.2023.103571.

[22] Thomas Pinder and Daniel Dodd, *GPJax: A Gaussian Process Framework in JAX*, Journal of Open Source Software, 2022, vol. 7, https://doi.org/10.21105/joss.04455.