Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference
Edited by Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen
 Published by Research Publishing, Singapore.
doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P3804-cd

An Integrated Uncertainty Quantification and Optimization for solving the 2025 NASA-DNV Challenge

Roberto Rocchetta, Lorenzo Nespoli, Vasco Medici

Università della Svizzera Italiana, SUPSI, Mendrisio, CH.
E-mail: [roberto.rocchetta],[lorenzo.nespoli],[vasco.medici]@supsi.ch

Yu Chen

Department of Mechanical and Aerospace Engineering, University of Liverpool, UK.
E-mail: yuchen2@liverpool.ac.uk

Marco de Angelis, Dawid Ochnio, Edoardo Patelli, Ewan Smith

University of Strathclyde, Glasgow, UK.
E-mail: [marco.de-angelis],[dawid.ochnio.2020],[edoardo.patelli],[ewan.smith]@strath.ac.uk

This paper presents a methodological framework for tackling the NASA and DNV Challenge on Optimization Under Uncertainty. The challenge requires designing and calibrating an uncertainty model using limited empirical data and optimizing design variables under uncertainty. We propose an integrated approach based on Bayesian experimental design, emulators, efficient computational tools, and advanced calibration techniques. Parametric and non-parametric uncertainty models are compared, calibrated using strategies incorporating likelihood-free KNN and discrepancy-based filtering methods, imprecise probability and likelihood-based ABC inference using Transitional Markov chain Monte Carlo. Uncertainty-based optimization is also performed by different approaches, including grid search, genetic algorithms, and two-level stochastic optimization using Bayesian techniques supported by surrogate models. The framework refines the uncertainty model by systematically updating the distributions and selecting optimal experimental conditions to enhance learning efficiency. Our results highlight the efficacy of the approach in balancing performance, reliability, and risk-constrained objectives that are generally applicable in UQ-driven decision-making problems.

Keywords: Uncertainty Quantification, Bayesian Optimization, Approximate Bayesian Computation, Surrogate Modelling, Experimental Design, Variational-Auto-encoders, Robust Risk-Constrained Optimization.

1. Introduction

The challenge problem, detailed in Agrell et al. (2025), concerns the behaviour of a system

$$\mathcal{Y}^*(x_a, x_e^*, x_c, s) : \mathbb{R}^8 \rightarrow \mathbb{R}^{T \times n_y} \tag{1}$$

where inputs include: $x_a \in [0,1]^3$ aleatoric variables with unknown distribution, epistemic parameters $x_e \in [0,1]^2$, unknown but fixed, control variables $x_c \in [0,1]^3$, and a seed $s \in \mathbb{N}$ for the random number generator. The input vector $X = (x_a, x_e, x_c, s)$ applies to a simulation model $\mathcal{Y}(X)$ that approximates the true system $\mathcal{Y}^*$, produces $n_y = 6$ time dependent outputs, evaluated over $T = 60$ timesteps. Responses are indexed by $I_1 = \{1,2,3\}$ if performance-related and by $I_2 = \{4,5,6\}$ if reliability-related.

Two main problems must be tackled: 1) model calibration and uncertainty quantification; and 2) robust design under uncertainty. The former problem focuses on the definition of an uncertainty model, (f_a, E), where f_a is the density function for x_a and E a set supposedly containing the true x_e^*, while problem 2 focuses on the identification of optimal designs compromising between reliability, risk, and performance. The challenge allowed $n_q = 10$ queries of the model by selecting $x_{c,q}$ and returning $K = 100$ samples, forming an empirical dataset, $\mathcal{D}_q^{\text{emp}} = \{\{y_k\}_{k=1}^K, x_{c,q}\} \in \mathbb{R}^{T \times n_y \times K+3}$ in addition to an initial dataset $\mathcal{D}_0^{\text{emp}}$.

The worst-case performance for given x_c is defined as follows:

$$\underline{J}(x_c) = \min_{x_e \in E} \quad \mathop{\mathbb{E}}_{x_a \sim f_a} [j(X)] \tag{2}$$

where $j(X) = \sum_{t=1}^{T} \sum_{i \in I_1} y_i(X,t)\,\Delta t$ is the total performance and $\Delta t = 0.01695$. The worst-case probability of failure with a limit state function $h = \min_{i \in \{1,2,3\}} g_i(X)$ is defined as follows:

$$\overline{P_f}(x_c) = \max_{x_e \in E} \mathbb{P}\left[h < 0\right], \tag{3}$$

and the worst-case tail expectation (risk) as:

$$\underline{R_f}(x_c) = \min_{x_e \in E} \mathbb{E}\left[h \mid h \leq 0\right]. \tag{4}$$

Note that the worst-case risk is a lower bound because this tail expectation is always negative. A detailed description of the challenge can be found in the challenge's github[a].

2. Proposed approach

The main goal of Problem 1 is to find an uncertainty model (UM), (f_a, E) where f_a is a PDF for the aleatory variables x_a and E is a set bounding the epistemic vector x_e, while Problem 2 focuses on the identification of optimal x_c designs compromising between reliability, performance and risk. The general workflow is as follows.

(1) $q \leftarrow 0$
(2) Calibrate UM using $\mathcal{D}_q^{\text{emp}}$
(3) Optimization of x_c
(4) Optimal design of experiment
(5) Multi-loop uncertainty propagation
(6) $q \leftarrow q + 1$ and repeat from point 2

We proposed using different techniques for cross-checking and to increase confidence in the results.

2.1. *Uncertainty models and calibration*

The uncertainty models considered are:

$\boldsymbol{UM}_{\text{knn}}$ consisting of a non-parametric kernel density estimator (KDE) for f_a and a convex hull for E providing tractability for robust optimization, e.g., Gorissen et al. (2015). A KNN-based calibration strategy and a filtering approach are proposed to identify suitable input realizations.

$\boldsymbol{UM}_{\text{peel}}$ is built with an imprecise model inversion strategy, see de Angelis et al. (2021), that is used to calibrate the model and to produce posterior samples of f_a and E. The model is non-parametric, prior- and likelihood free and returns a credal set of probability mixtures that bound the target distribution. Posterior samples are generated from an optimal mixture distribution within the credal set.

$\boldsymbol{UM}_{\text{beta}}$ is a second-order parametric model with Beta marginals $\mathcal{B}(x_{a_i}; \theta)$ and a Gaussian copula parametrised by ρ to model the stochastic dependence. This model augments the epistemic space to $e = \{\theta, \rho, x_e\}$, see e.g. Gray et al. (2022). A bivariate imprecise structure can be derived from f_a by representing epistemic uncertainty in the parameters through intervals.

Staircase density functions (Crespo et al., 2018) have also been explored, showing promising preliminary results but due to space constraints are not discussed further in this paper.

The calibration strategies explored are:

K-nearest neighbourhoods (KNN): This data-driven calibration and a filtering approach is based on a discrepancy vector $\delta = \delta(x_a, x_e, x_c)$, i.e. difference between output features in the empirical data and the simulated model response. The selected features are the j, g_i and additional features obtained using a conditional variation auto encoder. We initialise non-informative prior distributions $f(x_a, x_e) \sim U([0,1]^5)$ for both aleatory variables and epistemic parameters. Then, a data set $\mathcal{D}^{\text{sim}}$ is generated from it and used within the KNN filtration process as follows:

$$\mathcal{D}_q^{knn} \leftarrow \arg \min_{(x_a, x_e) \in \mathcal{D}^{\text{sim}}} \|\delta(x_e, x_a, x_c)\|_2, \tag{5}$$

where for each sample in the empirical $\mathcal{D}_q^{\text{emp}}$, k nearest neighbourhoods are obtained by minimising the discrepancy on a reduced set of outputs including $j(X), g_i(X)$ and additional features. The procedure yields $k \times K$ input pairs, i.e., $\mathcal{D}_q^{knn} = \{(x_a^{(i)}, x_e^{(i)})\}_{i=1}^{k \times K}$. The data in $\mathcal{D}_q^{knn}$ is then used to fit the UM_{knn} by aggregating samples for the query steps $q = 0, 1, 2....$ KDE is fitted on the x_a samples, and the convex hull is constructed over the intersection of the filtered x_e points. The epistemic points are selected based on a discrepancy vector $\delta_q(x_e) = r^{\text{sim}}(x_e) - r_q^{\text{emp}}$, where the

[a] https://github.com/dnv-opensource/UQ-Challenge-2025/

ratios $r = (\frac{y_4}{y_5}, \frac{y_4}{y_6}, \frac{y_5}{y_6})$ are used for this because invariant with respect to x_a, s, making them very suitable for isolating the effects of x_c, x_e. An η-level set is then defined for each $q = 1, .., n_q$ as

$$E_{\eta,q} = \{x_e : ||\delta_q(x_e)||_\infty \leq \eta\}, \quad (6)$$

where η is a threshold discrepancy on the infinity norm. A sampling algorithm will approximate the set (6) as $E^{\text{hull}}_{\eta,q}$, which includes all samples x_e whose outputs are closest to the empirical data at step q. Then, a intersection hull is computed as $E^{hull}_{\eta} = \cap_q E^{hull}_{\eta,q}$ by aggregating sets from all the available empirical datasets.

Data-peeling algorithm $\boldsymbol{UM}_{\text{peel}}$. A stack of nested rectangular sets is built on the observations. The observations at the boundary (support vectors) are progressively removed or "peeled away" to build the stack of nested sets; see, e.g. de Angelis et al. (2021). Each rectangular set is assigned an enclosing lower probability using scenario theory, i.e. solving a chance-constrained program. The inverse problem is then addressed using uniform samples in $[0, 1]^5$. Upon propagation, uniform samples are assigned membership to each of the enclosing sets. The resulting stack of nested sets in the input space is a calibrated consonant Dempster-Shafer structure whose basic beliefs are set equal to the scenario lower probabilities. Posterior samples are finally generated from a mixture distribution within the credal set identified by the calibrated consonant structure.

Approximate Bayesian Computation (ABC) see e.g. Beaumont (2019), are well established methods often used to approximate posterior distributions using a likelihood, i.e., by comparing simulated data with measurements using stochastic distances. For the calibration, we consider ABC approaches combining transitional Markov Chain Monte Carlo (TMCMC) (Ching and Chen, 2007)with a standard Pseudo-Likelihood:

$$\mathcal{L}(x_e|x_c) \propto \exp\left(-\frac{\delta(x_e, x_a, x_c)}{\epsilon}\right) \quad (7)$$

and with approximate likelihood obtained from a latent-space model Lee et al. (2024):

$$\mathcal{L}(x_e|x_c) \propto \int_Z \frac{q_\phi(\mathbf{z}|\mathbf{X}_{obs}) \cdot q_\phi(\mathbf{z}|\theta)}{p(\mathbf{z})} d\mathbf{z}. \quad (8)$$

This calibration returns $\boldsymbol{UM}_{\text{beta}}$.

2.2. *Optimisation*

Challengers must find a design x_c that maximizes the worst-case J while satisfying stringent reliability ($\epsilon = 10^{-3}, 10^{-4}$) and risk ($\beta = -300$) constraints. The selection of the control points are obtained with the following strategies:

Grid-search approach is used to explore the 3-dimensional design space, where n_{grid} is the number of designs in the grid and $\mathcal{P}$ is a grid of solutions. The set of candidates has been evaluated using the provided simulator and a Neural Network based surrogate model and fitness scores computed, see e.g. equation (2). For the performance assessment, a nested loop MC strategy is used as presented next.

Nested stochastic optimisation approach: The multi-objective optimisation problem is defined on the objective bounds as shown in Eqs. (2-4). Bayesian optimisation is adopted in the outer loop to find the next most promising design $x_{c,q}$ to query the real system, also with the aid of an acquisition function. A Gaussian process is directly mapping towards the bounds (the lower bound in this example), $\underline{J}(x_c^{(i)}) \approx \mathcal{GP}(x_c^{(i)})$. The inner loop is based on genetic algorithm (GA) which is employed to search for the bounds of objective functions subject to the epistemic uncertainty $e \in E$, as denoted in Eq. (2).

Conservative chance constraint consists in lowering safety limits by a defined quantity V and checking $P_{f,V} = \mathbb{P}[h - V < 0]$. If a design meets these stricter conditions, it is guaranteed to satisfy the original constraints, though possibly with reduced performance. This greatly simplified the (reliability) comparison of different x_c since reduced number of samples are required to estimate P_f.

2.3. *Optimal Design of Experiment (DOE) and data acquisition*

Two approaches identify experimental design candidates $x_{c,q}$ by solving an optimal DOE problem $x^*_{c,q} = \arg\max \alpha_{DOE}(x_c)$, where $\alpha_{DOE}(x_c)$ is an acquisition function. The first approach maximises a weighted combination of expected infor-

mation gain and performance scores, as defined by Equations (2) to (3). This balances both optimisation and information gain criteria, e.g., Li et al. (2024). The second approach uses Bayesian optimization for second-level optimization. A key advantage of Bayesian optimization is its acquisition function, $\alpha(x_{c,q})$, which identifies the most promising query points for evaluation on the real system. In this analysis, we tend to be more explorative and parametrise acquisition functions to discover the uncertain x_c space for more informative refinement of the UM in the calibration stage.

2.4. *Surrogate models*

A **feed-forward neural network** (FFNN) was trained on a synthetic dataset of one million uniformly sampled inputs $X = (x_e, x_a, x_c) \in [0,1]^8$ and random integer seeds, with corresponding model responses y. The model was trained to predict just the expectation operator in (2) and the reliability functions (g_1, g_2, g_3); this reduces the dimensionality of the output while being functional to the calibration/optimization process. After hyperparameter tuning, the final FFNN model (with 4 layers, 512 neurons per layer, and a learning rate of 0.005) was compared to a linear regressor. A normalised Root-mean-square deviation (nRMSE) measures the quality of FFNN models:

$$\text{nRMSE}(y, \hat{y}) = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}}{\frac{1}{n}\sum_{i=1}^{n}|y_i|} \qquad (9)$$

Specifically, FFNN achieved a nRMSE of 0.05 for estimating J, compared to 0.3 for the linear model. For estimating g_3—the most challenging among the g_i—the FFNN had a nRMSE of 0.12, while the linear model achieved 0.5.

Variational auto-encoder (VAE) and Conditional VAEs, have also been used to estimate Eq. (8) and for features extraction in combination with the KNN calibration algorithm (5). Specifically, enhanced performance is observed by focusing on the most informative encoded output features, conditioned on the given $x_{c,q}$.

3. Results - problem 1

3.1. *Preliminary analysis*

One-at-a-time sensitivity analysis allowed to identify a pattern of behaviour for x_{a1} and x_{a2}, that is shown in Table 1. It is found that for $x_{a1} = 0$ all model outputs are zeros irrespectively of the values of the other parameters. Whatever the value of (x_e, x_c), an increase of x_{a1} produces an increase in the mean of y_4, y_5 and y_6, while $x_{a2} = 0$ sets the variance of the model output to zero.

Table 1. Effect of increasing x_{a1} ($x_{a2} = 0$) on the mean model output of y_4, y_5, y_6.

x_{a1}	mean(y_4)	mean(y_5)	mean(y_6)	var($\forall y$)
0.0	0.0	0.0	0.0	0.0
0.2	138.33	171.22	77.81	0.0
0.6	1821.45	913.54	851.99	0.0
1.0	6620.74	4663.74	1979.36	0.0

3.2. *Calibration*

For the $\boldsymbol{UM}_{\text{knn}}$, using the approach described in Section 2.1 and progressively reducing the η threshold a KDE distribution for f_a and convex hull set for E are identified. These are computed filtering 50,000 simulations, uniformly sampled in (x_e, x_a), and conditioned on $(x_{c,q})$. Figure 1 shows the individual hull sets, and the intersection hull at step $q = 7$. Table 2 shows bounding boxes for different α levels, with no points remaining for very small thresholds α, possibly due to a discrepancy between local and remote simulators. This approach efficiently constrains epistemic uncertainty without probabilistic assumptions. Figure 2 shows a data sets obtained from the KNN calibration procedure for the first 4 empirical data sets. The procedure yields a best guess $x_e = (0.334, 0.596, 0.378)$ for a low η.

Table 2. The intersection hull E_η as intervals projected on the 3 epistemic components at stage $q = 7$.

η	x_{e1}	x_{e2}	x_{e3}
0.1	[0.084, 0.650]	[0.000, 0.891]	[0.000, 0.999]
0.02	[0.151, 0.399]	[0.445, 0.687]	[0.003, 0.810]
0.01	[0.221, 0.377]	[0.478, 0.639]	[0.007, 0.630]

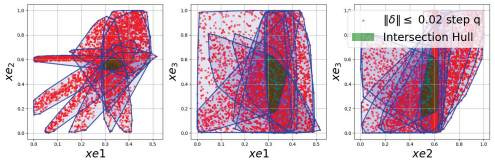

Fig. 1. Filtration process on the epistemic set at stage $q = 7$ and $\eta = 0.02$.

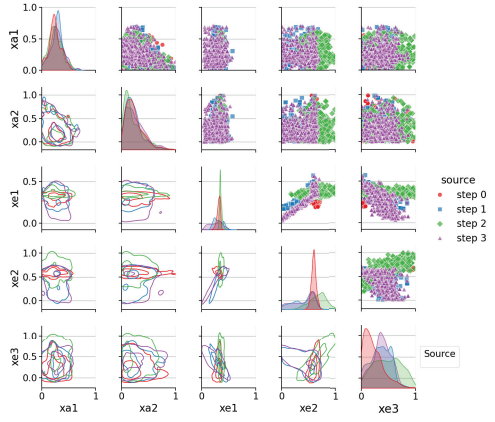

Fig. 2. An example of KNN calibration results on $\mathcal{D}_q^{\text{emp}}$ at stage $q = 0, 1, 2, 3$.

For $\boldsymbol{UM}_{\text{beta}}$, the parameters of the imprecise Beta model, *i.e.* θ and ρ, are updated sequentially, where the posterior PDF serves as the prior for the next with Hamiltonian Monte Carlo. A 95% highest density interval is used to derive the bound the parameter while a uniform prior employed for the updating. At $q = 10$, the resulting posterior bivariate p-box ($\boldsymbol{UM}_{\text{beta}}$) for x_a is $x_{a1} \sim \mathcal{B}([2.44, 2.45], [6.64, 6.69])$, $X_{a2} \sim \mathcal{B}([1.22, 1.23], [4.17, 4.19])$, and $\rho = [-0.21, -0.10]$. Fig. 3 shows the predictive p-box from the calibrated models for the y component and the empirical data for $q = 10$ and at $t = 20$ showing the ability to enclose the observed data.

3.3. *Comparison*

Estimation of the tightest prediction interval bounds by given α values are shown in Table 3.

Figure 4 compares the set E obtained for three UM. For $\boldsymbol{UM}_{\text{beta}}$, the set E is selected by slicing

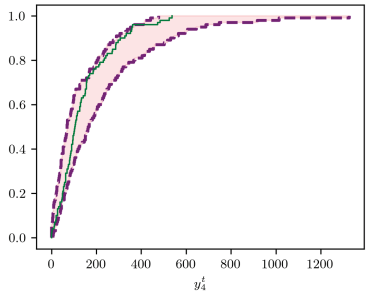

Fig. 3. Predictive p-box at $t = 20$ from $\boldsymbol{UM}_{\text{beta}}$ (area) and the empirical data for $x_{c,10}$ (in green).

Table 3. "Prediction intervals" of response y_i obtained using different UMs for Problem 1.

$\boldsymbol{UM}$	Output	$\alpha = 0.95$	$\alpha = 0.999$
	y_1, y_2, y_3	[0.00 3.35]	[0.00 3.35]
peel	y_4	[0.05, 605.8]	[0.052, 739.1]
peel	y_5	[0.037, 437.9]	[0.037, 533.0]
peel	y_6	[0.020, 233.7]	[0.020, 284.0]
knn	y_4	[0.0, 835.9]	[0.0, 1056.4]
knn	y_5	[0.0, 407.5]	[0.0, 518.9]
knn	y_6	[0.0, 345.2]	[0.0, 440.9]
beta	y_4	[20.2, 1197.6]	[0.6, 1659.5]
beta	y_5	[13.2, 777.2]	[0.4, 1084.1]
beta	y_6	[6.6, 389.3]	[0.2, 542.4]

the posterior based on high density interval as described in Section 3.2 , resulting in a box-shaped region (shown in green). For the two other models, the orange and blue markers indicate $E_{.035}^{hull}$ obtained for a threshold $\eta = 0.035$ and a smaller $E_{.005}^{hull}$, respectively. Note that the axes are much tighter than the initial interval $[0, 1]$, hence all the calibration approaches successfully reduced the epistemic uncertainty substantially. Also, the calibrated predictions are all consistent as they show a non-empty intersection, likely containing the true x_e^*.

4. Results - problem 2

We performed a grid search over the design space $x_c \in [0, 1]^3$, evaluating the worst-case $\underline{J}$, failure probability $\overline{P_f}$, and risk $\underline{R_f}$ according to the definition provided in the challenge description, the calibrated UMs and using the FFNN surrogate model to speed up the computation of the worst-

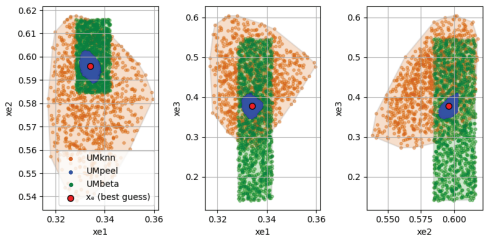

Fig. 4. Sets E from different UMs, uniform samples within the sets and best epistemic guess.

case bounds. Figure 6 shows the results obtained from a dense grid search with $n_{\text{grid}} = 30^3$ points. Each row presents 2D projections of x_e. From left to right, the columns show the worst-case risk, the worst-case performance, and the worst-case failure probability. The designs at each query step are indicated as red dots and labelled accordingly. The design $x_{c,7}$ has very high reliability (deep blue area in the right column), at step $x_{c,1}$ high performance low reliability, at step $q = 5$ compromise solution (reliable and high objective function). The corresponding controls $x_c^\star$ are in Table 4.

With the refinement of the uncertainty models at step $q = 10$, the nested stochastic optimisation approach has a narrower E to search. In the second level, the optimiser guides the search for optimal control, while in the first level, the genetic algorithm is set to explore the epistemic E space to find the endpoints of the objectives. In solving the $\epsilon-$ constrained and risk-based design, an additional penalty term is added where in compliance with the constraints. To ease the comparison and discussion, we will focus our selection for problem 2 on the 10 designs $x_{c,q}$. Hence, the results of this optimisation are not reported here and will be added to future extensions.

For the selection of the 5 optimal designs, we decided to limit our selection to the 11 designs for which empirical data are available. This facilitated the comparison among different models as shown in Fig. 5. Hence, the design $q = 1$ (see Table 4) is the chosen performance-based design because it achieved the highest worst-case performance score regardless of reliability. The reliability-based design is $q = 7$ due to its superior performance in minimizing the worst-case failure probability with the CP bound on the empirical data equal to 0.0362, and the lowest failure probability (0.020) under the condition of $V = 500$. This design is also favourably located in a low-probability region, as demonstrated by the contour plot of Fig. 6 (right column). Furthermore, within the chance-constrained framework, designs $q = 9$ and $q = 10$ have been selected representing different levels of conservativeness. Design $q = 9$ is less conservative, offering a balanced compromise between performance and reliability. Design $q = 5$ is the risk-based design, selected taking into account the potential risks while ensuring competitive performance. The risk-based design is chosen balancing high performance with risk reduction.

4.1. *Comparison*

Table 4 summarizes the results for the $x_{c,q}$ along with the corresponding empirical estimates of performance objective, system failure probability, and confidence bounds for a 0.95 level, i.e., concentration bounds on expectations and Clopper-Pearson upper bound on P_f. Table 5 present the propagated intervals for $\boldsymbol{UM}_{\text{knn}}$, estimated upper bounds on P_f and $P_{f,V}$ are reported. It is worth noting that the failure probability bounds appear to adequately capture the empirical estimates across the cases considered. However, in some instances, the empirical mean of the performance metric lies outside the predicted bounds, an observation that will be further investigated in future work.

5. Conclusions

An integrated framework for uncertainty quantification and design under uncertainty has been presented. The framework relies on different approaches for cross-checking the results and gain confidence in the proposed solutions. Surrogate models are essential to the framework given the greedy algorithms for Bayesian optimisation and double-loop uncertainty propagation. The deployment of multiple calibration strategies has significantly strengthen our confidence in the calibrated

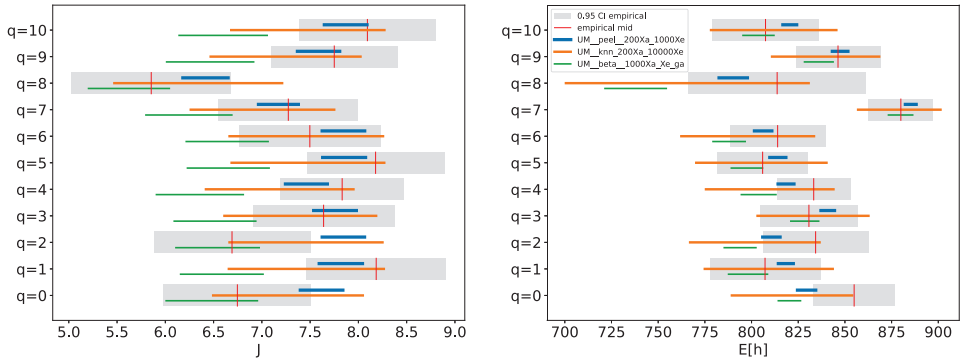

Fig. 5. Comparison between $\boldsymbol{UM}_{\text{peel}}$ ($E^{hull}_{.02}$), $\boldsymbol{UM}_{\text{knn}}$ ($E^{hull}_{.035}$) with 200 aleatory 10^4 epistemic samples and $\boldsymbol{UM}_{\text{beta}}$ with 10^3 aleatory samples. Results of the first two are evaluated using the simulator while the third evaluated using surrogate. Intervals for $\mathbb{E}[j]$ and $\mathbb{E}[h]$ for $q = 0, .., 10$ are superimposed to the 95% confidence intervals on the empirical (queried) data displayed with the gray shaded areas.

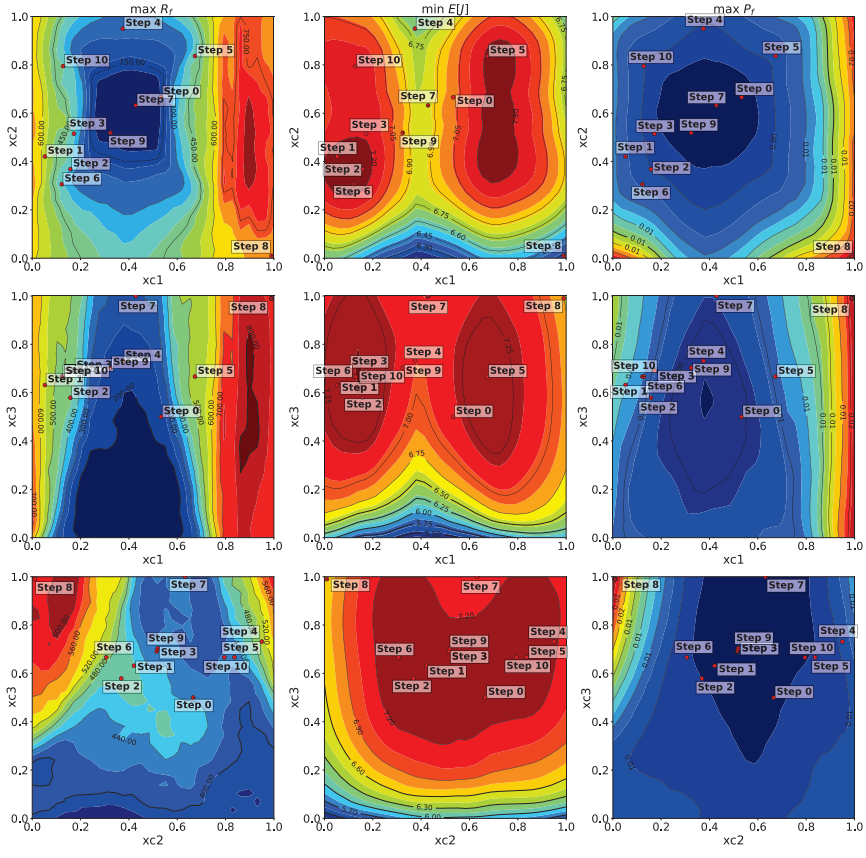

Fig. 6. Contour plots of worst case scores (risk in column 1, performance column 2 and probability of failure column 3) as function of the x_c. The locations of the designs $x_{c,q}$, steps $q = 0, 1, 2, .., 10$, are also presented.

Table 4. Main statistics for the empirical data. All intervals are 95% confidence intervals. The table shows the expected performance $\hat{J}$, reliability response $\hat{\mathbb{E}}[h]$, estimated failure probability $\hat{P}_f$, its 99% Clopper-Pearson (CP) upper bound, and ratios r. CC indicates a chance constrained design.

q	$x_{c,q}$	$\hat{J}$	$\hat{P}_f(\overline{P}_f^{CP})$	$\hat{\mathbb{E}}[h]$	$\frac{y_4}{y_5}, \frac{y_4}{y_6}, \frac{y_5}{y_6}$	Tag
0	(0.53 , 0.67 , 0.50)	[5.98, 7.51]	0/100 (0.0362)	[833.2, 876.7]	2.03, 2.41, 1.19	Baseline
1	(0.05 , 0.42 , 0.63)	[7.46, 8.91]	1/100 (0.0545)	[777.5, 836.9]	1.75, 3.43, 1.96	Performing
2	(0.16 , 0.37 , 0.58)	[5.88, 7.50]	0/100 (0.0362)	[806.1, 862.5]	1.43, 2.74, 1.92	-
3	(0.17 , 0.52 , 0.69)	[6.91, 8.37]	0/100 (0.0362)	[804.7, 856.7]	1.91, 3.17, 1.66	-
4	(0.37 , 0.95 , 0.73)	[7.19, 8.47]	0/100 (0.0362)	[813.5, 853.1]	0.75, 1.63, 2.18	-
5	(0.67 , 0.84 , 0.67)	[7.47, 8.89]	0/100 (0.0362)	[781.7, 830.1]	1.41, 2.96, 2.11	Risk-Con.
6	(0.12 , 0.31 , 0.67)	[6.76, 8.23]	0/100 (0.0362)	[788.4, 839.5]	1.27, 2.87, 2.26	-
7	(0.43 , 0.63 , 1.00)	[6.55, 7.99]	0/100 (0.0362)	[862.8, 897.2]	1.39, 2.60, 1.88	Reliable
8	(0.99 , 0.01 , 0.99)	[5.03, 6.68]	2/100 (0.0704)	[766.2, 861.2]	1.17, 4.63, 3.96	-
9	(0.32 , 0.52 , 0.70)	[7.09, 8.41]	0/100 (0.0362)	[823.7, 868.8]	1.54, 2.29, 1.48	CC $\epsilon = 10^{-4}$
10	(0.13 , 0.80 , 0.67)	[7.39, 8.80]	0/100 (0.0362)	[779.0, 835.9]	1.53, 3.06, 2.00	CC $\epsilon = 10^{-3}$

Table 5. Propagated intervals for $\boldsymbol{UM}_{\text{knn}}$, using $E^{hull}_{.035}$, 10^4 epistemic and 200 aleatoric samples.

q	$J \in$	$\overline{P}_f$	$\overline{P}_{f,500}$	$\mathbb{E} \in$
0	[6.49, 8.04]	0.005	0.075	[789.5, 853.8]
1	[6.65, 8.26]	0.020	0.075	[775.0, 836.5]
2	[6.66, 8.24]	0.010	0.085	[767.1, 836.4]
3	[6.61, 8.18]	0.010	0.085	[803.2, 862.6]
4	[6.42, 7.94]	0.010	0.050	[775.6, 843.9]
5	[6.68, 8.26]	0.010	0.075	[770.4, 840.1]
6	[6.66, 8.25]	0.010	0.085	[762.4, 833.5]
7	[6.26, 7.75]	0.000	0.020	[857.1, 901.2]
8	[5.47, 7.21]	0.075	0.140	[700.6, 830.6]
9	[6.46, 8.02]	0.000	0.045	[811.0, 868.4]
10	[6.68, 8.27]	0.015	0.075	[778.3, 845.4]

UMs. These include an efficient KNN method combining hulls and kernel-density estimation, a rigorous imprecise-probabilistic data peeling method, and a second-order Bayesian updating method. Future work will concentrate on electing the best design based on the whole set of models.

References

Agrell, N., L. G. Crespo, V. Flovik, E. Vanem, and S. Kenny (2025). The nasa and dnv challenge on optimization under uncertainty.

Beaumont, M. A. (2019). Approximate bayesian computation. *Annual Review of Statistics and Its Application 6*(Volume 6, 2019), 379–403.

Ching, J. and Y.-C. Chen (2007). Transitional Markov Chain Monte Carlo Method for Bayesian Model Updating, Model Class Selection, and Model Averaging. *Journal of Engineering Mechanics 133*(7), 816–832.

Crespo, L. G., S. P. Kenny, D. P. Giesy, and B. K. Stanford (2018). Random variables with moment-matching staircase density functions. *Applied Mathematical Modelling 64*, 196–213.

de Angelis, M., R. Rocchetta, A. Gray, and S. Ferson (2021). Constructing consonant predictive beliefs from data with scenario theory. In *International Symposium on Imprecise Probability: Theories and Applications*, pp. 357–360.

Gorissen, B. L., İ. Yanıkoğlu, and D. den Hertog (2015). A practical guide to robust optimization. *Omega 53*, 124–137.

Gray, A., A. Wimbush, M. de Angelis, P. Hristov, D. Calleja, E. Miralles-Dolz, and R. Rocchetta (2022). From inference to design: A comprehensive framework for uncertainty quantification in engineering with limited information. *Mechanical Systems and Signal Processing 165*, 108210.

Lee, S., T. Yaoyama, M. Kitahara, and T. Itoi (2024). Latent space-based stochastic model updating. *arXiv preprint arXiv:2410.03150*.

Li, F., R. Baptista, and Y. Marzouk (2024). Expected information gain estimation via density approximations: Sample allocation and dimension reduction.