Advanced methodology for uncertainty propagation in computer experiments with large number of inputs

Bertrand Iooss, Amandine Marrel

HAL Id: hal-01907198

https://hal.archives-ouvertes.fr/hal-01907198

Submitted on 28 Oct 2018

Advanced methodology for uncertainty propagation in computer experiments with large number of inputs

Bertrand Iooss,[a,b,*] and Amandine Marrel,[c]

[a]*EDF R&D, Département PRISME*
6 Quai Watier, 78401 Chatou, France

[b]*Institut de Mathématiques de Toulouse*
31062 Toulouse, France

[c]*CEA, DEN, DER, SESI, LEMS*
13108 Saint-Paul-lez-Durance, France

[*]Email: bertrand.iooss@edf.fr

October 28, 2018

Abstract

Complex computer codes are often too time expensive to be directly used to perform uncertainty propagation or sensitivity analysis. A solution to cope with this problem consists in replacing the cpu-time expensive computer model by a cpu inexpensive mathematical function, called metamodel. Among the metamodels classically used in computer experiments, the Gaussian process (Gp) model has shown strong capabilities to solve practical problems. However, in case of high dimensional experiments (with typically several tens of inputs), the Gp metamodel building process remains difficult. To face this limitation, we propose a methodology which combines several advanced statistical tools: initial space-filling design, screening to identify the non-influential inputs, Gaussian process metamodel building with the group of influential inputs as explanatory variables. The residual effect of the group of non-influential inputs is captured by another Gaussian process metamodel. Then, a sensitivity analysis based on variance decomposition and an uncertainty propagation (here 95%-quantile estimation) can be performed through the joint Gp metamodel. The efficiency of the methodology is illustrated on a thermal-hydraulic calculation case simulating accidental scenario in a Pressurized Water Reactor. More precisely, a Loss Of Coolant Accident is considered, which takes into account a double-ended guillotine break with a specific size piping rupture.

Keywords — Gaussian process, Metamodel, Quantile, Screening, Sobol' indices

I. INTRODUCTION

Best-estimate computer codes are increasingly used to estimate safety margins in nuclear accident management analysis instead of conservative procedures [1, 2]. In this context, it is essential to evaluate the accuracy of the numerical model results, whose uncertainties come mainly from the lack of knowledge of the underlying physics and the model input parameters. The so-called Best Estimate Plus Uncertainty (BEPU) methods were then developed and introduced in safety analyses, especially for thermal hydraulic issues, and even more precisely for the large-break loss-of-coolant accident [3, 4, 5]. Its man principles rely on a probabilistic modeling of the model input uncertainties, on Monte Carlo sampling for running the thermal-hydraulic computer code on sets of inputs, and on the application of statistical tools (based for example on order statistics as the Wilks' formula) to infer high quantiles of the output variables of interest [6, 7, 8, 9, 10].

Not restricted to the nuclear engineering domain, the BEPU approach is more largely known as the uncertainty analysis framework [11]. Quantitative assessment of the uncertainties tainting the results of computer simulations is indeed a major topic of interest in both industrial and scientific communities. One of the key issues in such studies is to get information about the output when the numerical simulations are expensive to run. For example, one often faces up with cpu time consuming numerical models and, in such cases, uncertainty propagation, sensitivity analysis, optimization processing and system robustness analysis become difficult tasks using such models. In order to circumvent this problem, a widely accepted method consists in replacing cpu-time expensive computer models by cpu inexpensive mathematical functions (called metamodels) based, e.g., on polynomials, neural networks, or Gaussian processes [12]. This metamodel is built from a set of computer code simulations and must be as representative as possible of the code in the variation domain of its uncertain parameters, with good prediction capabilities. The use of metamodels has been extensively applied in engineering issues as it provides a multi-objective tool [13]: once estimated, the metamodel can be used to perform global sensitivity analysis, as well as uncertainty propagation, optimization, or calibration studies. In BEPU-kind analyses, several works [14, 15, 16] have introduced the use of metamodels and shown how this technique can help estimate quantiles or probability of failure in thermal-hydraulic calculations.

However, the metamodeling technique is known to be relevant when simulated phenomena are related to a small number of random input variables (see [17] for example). In case of high

dimensional numerical experiments (with typically several tens of inputs), and depending on the complexity of the underlying numerical model, the metamodel building process remains difficult, or even impracticable. For example, the Gaussian process (Gp) model [18] which has shown strong capabilities to solve practical problems, has some caveats when dealing with high dimensional problems. The main difficulty relies on the estimation of Gp hyperparameters. Manipulating predefined or well-adapted Gp kernels (as in [19, 20]) is a current research way, while several authors propose to couple the estimation procedure with variable selection techniques [21, 22, 23].

In this paper, following the latter technique, we propose a rigorous and robust method for building a Gp metamodel with a high-dimensional vector of inputs. Moreover, concerning the practical use of this metamodel, our final goal is twofold: to be able to interpret the relationships between the model inputs and outputs with a quantitative sensitivity analysis, and to have a reliable high-level quantile estimation method which does not require additional runs of the computer code.

In what follows, the system under study is generically denoted

$$Y = g\left(X_1, \ldots, X_d\right) \tag{1}$$

where $g(\cdot)$ is the numerical model (also called the computer code), whose output Y and input parameters $X_1, \ldots, X_d$ belong to some measurable spaces $\mathcal{Y}$ and $\mathcal{X}_1, \ldots, \mathcal{X}_d$ respectively. $\mathbf{X} = (X_1, \ldots, X_d)$ is the input vector and we suppose that $\mathcal{X} = \prod_{k=1}^{d} \mathcal{X}_k \subset \mathbb{R}^d$ and $\mathcal{Y} \subset \mathbb{R}$. For a given value of the vector of inputs $\mathbf{x} = (x_1, \ldots, x_d) \in \mathbb{R}^d$, a simulation run of the code yields an observed value $y = g(\mathbf{x})$.

To meet our aforementioned objectives, we propose a sequential methodology which combines several relevant statistical techniques. Our approach consists in four steps:

1. **Step 1: Design of experiments.** Knowing the variation domain of the input variables, a design of n numerical experiments is firstly performed and yields n model output values. The obtained sample of inputs/outputs constitutes the learning sample. The goal is here to explore, the most efficiently and with a reasonable simulation budget, the domain of uncertain parameters $\mathcal{X}$ and get as much information as possible about the behavior of the simulator output Y. For this, we use a space-filling design (SFD) of experiments, providing a full coverage of the high-dimensional input space [12, 23].

2. **Step 2: Preliminary screening.** From the learning sample, a screening technique is performed in order to identify the Primary Influential Inputs (PII) on the model output variability and rank them by decreasing influence. To achieve it, we use dependence measures with associated statistical tests. These measures have several advantages: they can be directly estimated on a SFD, the sensitivity indices that they provide are interpretable quantitatively and their efficiency for screening purpose has been recently illustrated by [24, 25, 26].

3. **Step 3: Building and validation of joint metamodel.** From the learning sample, a metamodel is built to fit the simulator output Y. For this, we propose to use a joint Gp metamodel [27], by considering only the PII as the explanatory inputs while the other inputs (screened as non significantly influential) are considered as a global stochastic (*i.e. unknown*) input. Moreover, we use a sequential process to build the joint Gp where the ranked PII are successively included as explanatory inputs in the metamodel (ranking from Step 2). At each iteration, a first Gp model [22], only depending on the current set of explanatory inputs, is built to approximate the mean component. The residual effect of the other inputs is captured using a second Gp model, also function of the explanatory inputs, which approximates the variance component. The accuracy and prediction capabilities of the joint metamodel are controlled on a test sample or by cross-validation.

4. **Step 4: Use of the metamodel for sensitivity analysis and uncertainty propagation.** A quantitative sensitivity analysis (Step 4A) and an uncertainty propagation (Step 4B) are performed using the joint metamodel instead of the computer model, leading to a large gain of computation time [28, 27]. In this work, we are particularly interested in estimating variance-based sensitivity indices (namely Sobol' indices) and the 95%-quantile of model output.

The paper is organized as follows. In the following section, the nuclear test case which constitutes the guideline application of the paper is described. It consists in a thermal-hydraulic calculation case which simulates an accidental scenario in a nuclear Pressurized Water Reactor. Then, each step of the above statistical methodology is detailed in a dedicated section and illustrated as the same time on the use-case. A last section concludes the work.

II. THERMAL-HYDRAULIC TEST CASE

Our use-case consists in thermal-hydraulic computer experiments, typically used in support of regulatory work and nuclear power plant design and operation. Indeed, some safety analysis considers the so-called "Loss Of Coolant Accident" which takes into account a double-ended guillotine break with a specific size piping rupture. The test case under study is a simplified one, as regards both physical phenomena and dimensions of the system, with respect to a realistic modeling of the reactor. The numerical model is based on code CATHARE 2 (V2.5_3mod3.1) which simulates the time evolution of physical quantities during a thermal-hydraulic transient. It models a test carried out on the Japanese mock-up "Large Scale Test Facility" in the framework of the OECD/ROSA-2 project, and which is representative of an Intermediate Break Loss Of Coolant Accident (IBLOCA) [29]. This mock-up represents a reduced scale Westinghouse PWR (1/1 ratio in height and 1/48 in volume), with two loops instead of the four loops on the actual reactor and an electric powered heating core (10 MWe), see Figure 1. It operates at the same pressure and temperature values as the reference PWR. The simulated accidental transient is an IBLOCA with a break on the cold leg and no safety injection on the broken leg.
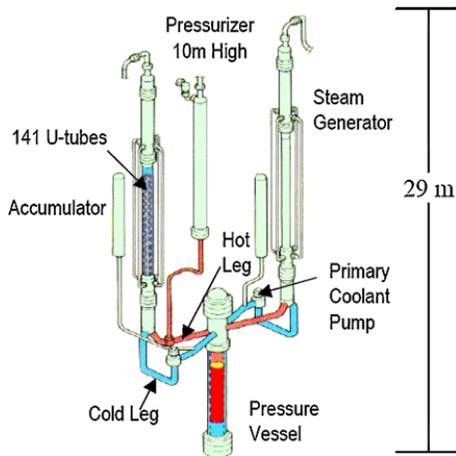

Fig. 1. Hot and cold legs of the Large Scale Test Facility.

In this use-case, $d = 27$ scalar input variables of CATHARE code are uncertain, defined by their probability density function (uniform, log-uniform, normal or log-normal). They correspond to various system parameters as, boundary conditions, some critical flow rates, interfacial friction

coefficients, condensation coefficients, heat transfer coefficients, etc. Only uncertainties related to physical parameters are considered here and no uncertainty on scenario variables (initial state of the reactor before the transient) is taken into account. Table I gives more details about these uncertain inputs and their probability density functions (pdf). The nature of these uncertainties appears to be epistemic since they come from a lack of knowledge on the true value of these parameters. The output variable of interest is a single scalar which is the maximal peak cladding temperature (PCT) during the accident transient (as an example, see the peak in Figure 2). This quantity is derived from the physical outputs provided by CATHARE code.

TABLE I
List of the 27 uncertain input parameters (with the type of their pdf) and associated physical models in CATHARE code. HTC means Heat Transfer Coefficient. For the pdf, U (resp. LU, N and LN) stands for Uniform (resp. Log-Uniform, Normal and Log-Normal).

| Type of inputs	Name of inputs	pdf	Physical models
Heat transfer	SDNBCO (X_1)	N	Departure from nucleate boiling
in the core	STMFSCO (X_2)	U	Minimum film stable temperature
	SCVC9 (X_3)	LN	HTC for steam convection
	SHCFV (X_4)	LN	Wall-fluid HTC
	SHBOI (X_5)	N	HTC for film boiling
Heat transfer in the steam	SGVHCFV (X_6)	LU	HTC forced wall-steam convection
generators (SG) U-tube	SPPCH (X_7)	N	Liquid-interface HTC for film condensation
Wall-steam friction in core	SP1CGZ (X_8)	LU	
Interfacial friction	STBAEBU (X_9)	LN	SG outlet plena and crossover legs together
	STOIBC1 (X_{10})	LN	Hot legs (horizontal part)
	STOIBC2 (X_{11})	LN	Bend of the hot legs
	STOIBC3 (X_{12})	LN	SG inlet plena
	STOIDC (X_{13})	LN	Downcomer
	STOICO (X_{14})	LN	Core
	SBRUP (X_{15})	LN	Upper plenum
	SBRLP (X_{16})	LN	Lower plenum
	SBRUH (X_{17})	LN	Upper head
Condensation	SQLEDC (X_{18})	LN	Downcomer
	SQLEBI (X_{19})	U	Cold leg (intact)
	SQLEBR (X_{20})	U	Cold leg (broken)
	SCOQBI (X_{27})	U	Jet
Break flow	QLEFBR (X_{21})	LN	Flashing (undersaturated)
	CLFBR (X_{22})	N	Wall-liquid friction (undersaturated)
	DQLEFBR (X_{23})	N	Flashing delay (undersaturated)
	SQLEFBR (X_{24})	LN	Flashing (saturated)
	SCLFBR (X_{25})	N	Wall-liquid friction (saturated)
	STOIFBR (X_{26})	LN	Global interfacial friction (saturated)
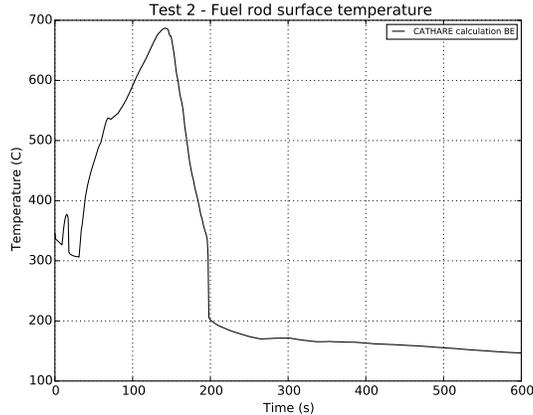
Fig. 2. Physical simulation output of the model with CATHARE code: maximal rod cladding temperature during the transient.

Our objective with this use-case is to provide a good metamodel for sensitivity analysis, uncertainty propagation and, more generally, safety studies. Indeed, the cpu-time cost of one simulation is too important to directly perform all the statistical analysis which are required in a safety study and for which many simulations are needed. To overcome this limitation, an accurate metamodel, built from a reduced number of direct code calculations, will make it possible to develop a more complete and robust safety demonstration.

III. STEP 1: DESIGN OF EXPERIMENTS

This initial step of sampling consists of defining a design of n experiments for the inputs and performing the corresponding runs with the numerical model g. The obtained sample of inputs/outputs will constitute the learning sample on which the metamodel will then be fitted. The objective is therefore to investigate, most efficiently and with few simulations, the whole variation domain of the uncertain parameters in order to build a predictive metamodel which approximates as accurately as possible the output Y.

For this, we propose to use a space-filling design (SFD) of a n of experiments, this kind of design providing a full coverage of the high-dimensional input space [12]. Among SFD types, a Latin Hypercube Sample (LHS, [30])) with optimal space-filling and good projection properties [23] would be well adapted. In particular, [12, 31] have shown the importance of ensuring good low-

8

order sub-projection properties. Maximum projection designs [32] or low-centered L^2 discrepancy LHS [33] are then particularly well-suited.

Mathematically, the experimental design corresponds to a n-size sample $\left\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\right\}$ which is performed on the model (or code) g. This yields n model output values denoted $\left\{y^{(1)}, \ldots, y^{(n)}\right\}$ with $y^{(i)} = g(\mathbf{x}^{(i)})$. The obtained learning sample is denoted (X_s, Y_s) with $X_s = \left[\mathbf{x}^{(1)T}, \ldots, \mathbf{x}^{(n)T}\right]^T$ and $Y_s = \left[y^{(1)}, \ldots, y^{(n)}\right]^T$. Then, the goal is to build an approximating metamodel of g from the n-sample (X_s, Y_s).

The number n of simulations is a compromise between the CPU time required for each simulation and the number of input parameters. For uncertainty propagation and metamodel-building purpose, some thumb rules propose to choose n at least as large as 10 times the dimension d of the input vector [34, 22].

To build the metamodel for the IBLOCA test case, $n = 500$ CATHARE simulations are performed following a space-filling LHS built in dimension $d = 27$. The histogram of the obtained values for the output of interest, namely the PCT, is given by Figure 3 (temperature is in °C). A kernel density estimator [35] of the data is also added on the plot to provide an estimator of the probability density function.
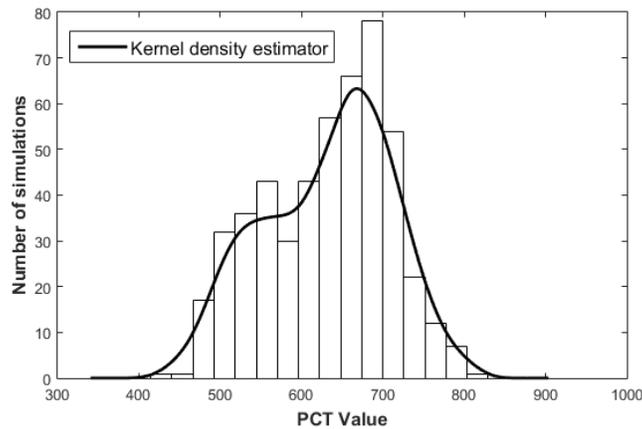

Fig. 3. Histogram of the PCT from the learning sample of $n = 500$ simulations.

Remark III.1 Note that the input values are sampled following their prior distributions defined on their variation ranges. Indeed, as we are not ensured to be able to build a sufficiently accurate metamodel, we prefer sample the inputs following the probabilistic distributions in order to have

at least a probabilized sample of the uncertain output, on which statistical characteristics could be estimated. Moreover, as explained in the next section, dependence measures can be directly estimated on this sample, providing first usable results of sensitivity analysis.

From the learning sample, some scatterplots of the PCT with respect to some well-chosen inputs are displayed in Figure 4. An additional local regression using weighted linear least squares and a first degree polynomial model (moving average filter) is added on each scatterplot to extract a possible tendency.
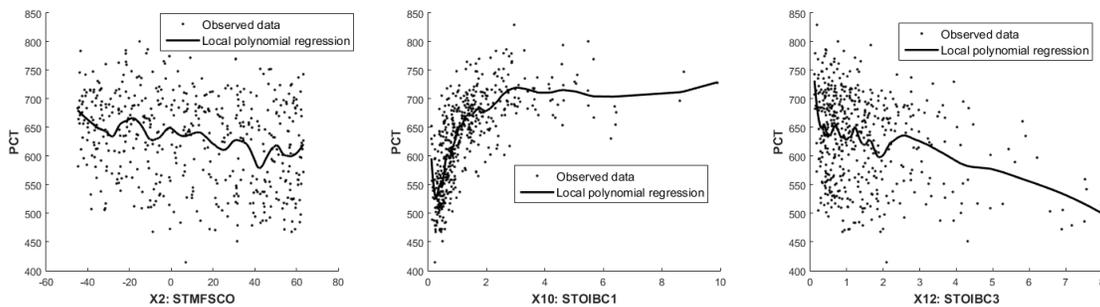

Fig. 4. Scatterplots with local polynomial regression of PCT according to several inputs, from the learning sample of $n = 500$ simulations.

IV. STEP 2: PRELIMINARY SCREENING BASED ON DEPENDENCE MEASURE

From the learning sample, a screening technique is performed in order to identify the primary influential inputs (PII) on the variability of model output. It has been recently shown that screening based on dependence measures [24, 25] or on derivative-based global sensitivity measures [36, 37] are very efficient methods which can be directly applied on a SFD. One of their great interest is that, additionally to their screening job, the sensitivity indices that they provide can be quantitatively interpreted and used to order the PII by decreasing influence, paving the way for a sequential building of metamodel.

In the considered IBLOCA test case, the adjoint model is not available and the derivatives of the model output are therefore not computed because of their costs. The screening step will then be based only on dependence measures and more precisely on HSIC indices, directly estimated from the learning sample.

10

IV.A. Screening based on HSIC dependence measure

Recently, [24] and [25] have proposed to use dependence measures for screening purpose, by applying them directly on a SFD. These sensitivity indices are not the classical ones based on the decomposition of output variance (see [38] for a global review). They consider higher order information about the output behavior in order to provide more detailed information. Among them, the Hilbert-Schmidt independence criterion (HSIC) introduced by [39] builds upon kernel-based approaches for detecting dependence, and more particularly on cross-covariance operators in reproducing kernel Hilbert spaces (RKHS).

If we consider two RKHS $\mathcal{F}_k$ and $\mathcal{G}$ of functions $\mathcal{X}_k \to \mathbb{R}$ and $\mathcal{Y} \to \mathbb{R}$ respectively, the crossed-covariance $C_{X_k,Y}$ operator associated to the joint probabilistic distribution of (X_k, Y) is the linear operator defined for every $f_{X_k} \in \mathcal{F}_k$ and $g_Y \in \mathcal{G}$ by:

$$\langle f_{X_k}, C_{X_k,Y} g_Y \rangle_{\mathcal{F}_k} = \mathrm{Cov}\left(f_{X_k}, g_Y\right). \tag{2}$$

$C_{X_k,Y}$ generalizes the covariance matrix by representing higher order correlations between X_k and Y through nonlinear kernels. The HSIC criterion is then defined by the Hilbert-Schmidt norm of the cross-covariance operator:

$$\mathrm{HSIC}(X_k, Y)_{\mathcal{F}_k, \mathcal{G}} = \|C_k\|_{HS}^2. \tag{3}$$

From this, [24] introduces a normalized version of the HSIC which provides a sensitivity index of X_k, lying in $[0, 1]$:

$$R_{\mathrm{HSIC},k}^2 = \frac{\mathrm{HSIC}(X_k, Y)}{\sqrt{\mathrm{HSIC}(X_k, X_k)\mathrm{HSIC}(Y, Y)}}. \tag{4}$$

The closer to one the $R_{\mathrm{HSIC},k}^2$, the stronger the dependence between X_k and Y. In practice, [39] propose a Monte Carlo estimator of $\mathrm{HSIC}(X_k, Y)$ and a plug-in estimator can be deduced for $R_{\mathrm{HSIC},k}^2$. Note that Gaussian kernel functions with empirical estimations of the variance parameter are used in our application (see [39] for more details).

Then, from the estimated R_{HSIC}^2, independence tests can be performed for a screening purpose. The objective is to separate the inputs into two sub-groups, the significant ones and the non-significant ones. For a given input X_k, statistical HSIC-based tests aims at testing the null

hypothesis "$\mathcal{H}_0^{(k)}$: X_k and Y are independent", against its alternative "$\mathcal{H}_1^{(k)}$: X_k and Y are dependent". The significance level[a] of these tests is hereinafter noted α. Several statistical hypothesis tests are available: asymptotic versions (for large sample size), spectral extensions and bootstrap versions for non-asymptotic case (case of small sample size). All these tests are described and compared in [25]; a guidance to use them for a screening purpose is also proposed.

So, at the end of this HSIC-based screening step, the inputs are clustered into two subgroups, PII and non-influential inputs, and the PII are ordered by decreasing R_{HSIC}^2.This order will be used for the sequential metamodel building in step 3.

IV.B. Application on IBLOCA test case

From the learning sample of $n = 500$ simulations, R_{HSIC}^2 dependence measures are estimated and bootstrap independence tests with $\alpha = 0.1$ are performed. The independence hypothesis $\mathcal{H}_0$ is rejected for eleven inputs, which are now designated as PII (Primary Influential Inputs). These inputs are given by Table II with their estimated R_{HSIC}^2. Ordering them by decreasing R_{HSIC}^2 reveals the predominance influence of the interfacial friction coefficient in the hot legs (X_{10}), followed by X_2, X_{12} and X_{22}. Then seven parameters have a lower influence: X_{15}, X_{13}, X_9, X_5, X_{14}, X_{26} and X_{27}.

Note that the estimated HSIC and the results of significant tests are relatively stable when the learning sample size varies from $n = 300$ to $n = 500$. Only two or three selected variables with a very low HSIC (R_{HSIC}^2 around 0.01) can differ. This confirms the robustness of the HSIC indices and the associated significance tests for qualitative ranking and screening purpose.

In the next steps, only the eleven PII are considered as explanatory variables in the joint metamodel and will be successively included in the building process. The other sixteen variables will be joined in a so-called *uncontrollable* parameter.

TABLE II
HSIC-based sensitivity indices R_{HSIC}^2 for the PII (Primary Influential Inputs), identified by independence test (Step 2).

PII	X_{10}	X_2	X_{12}	X_{22}	X_{15}	X_{13}	X_9	X_5	X_{14}	X_{26}	X_{27}
R_{HSIC}^2	0.39	0.04	0.02	0.02	0.01	0.01	0.01	0.01	0.01	0.01	0.01

---

[a]The significance level of a statistical hypothesis test is the rate of the type I error which corresponds to the rejection of the null hypothesis $\mathcal{H}_0$ when it is true.

V.   STEP 3: JOINT GP METAMODEL WITH SEQUENTIAL BUILDING PRO-
CESS

Among all the metamodel-based solutions (polynomials, splines, neural networks, etc.), we focus our attention on the Gaussian process (Gp) regression, which extends the kriging principles of geostatistics to computer experiments by considering the correlation between two responses of a computer code depending on the distance between input variables. The Gp-based metamodel presents some real advantages compared to other metamodels: exact interpolation property, simple analytical formulations of the predictor, availability of the mean squared error of the predictions and the proved capabilities for modeling of numerical simulators (see [40], [18] or [22]). The reader can refer to [41] for a detailed review on Gp metamodel.

However, for its application to complex industrial problems, developing a robust implementation methodology is required. Indeed, it often implies the estimation of several hyperparameters involved in the covariance function of the Gp (e.g. usual case of anisotropic stationary covariance function). Therefore, some difficulties can arise from the parameter estimation procedure (instability, high number of hyperparameters, see [22] for example). To tackle this issue, we propose a progressive estimation procedure based on the result of the previous screening step and using a joint Gp approach [27]. The interest of the previous screening step becomes twofold: to reduce the input space on which each component of the joint Gp is built (reduction to the PII space), and rank the selected explanatory inputs (i.e. the PII) in order to enable a sequential Gp-building process. It is expected that these two uses of screening results could significantly make the joint Gp building easier and more efficient.

V.A.   Sequential Gp-building process based on successive inclusion of PII as explanatory variables

At the end of the screening step, the PII are ordered by decreasing influence (decreasing R^2_{HSIC}). They are successively included as explanatory inputs in the Gp metamodel while the other inputs (the remaining PII and the other non-PII inputs) are joined in a single macro-parameter which is considered as an uncontrollable parameter (i.e. a stochastic parameter, notion detailed in section V.B). Thus, at the jth iteration, a joint Gp metamodel is built with, as explanatory inputs, the j first ordered PII. The definition and building procedure of a joint Gp is fully described in

13

[27] and summarized in the next subsection.

However, building a Gp or a joint Gp involves to perform a numerical optimization in order to estimate all the parameters of the metamodel (covariance hyperparameters and variance parameter). As we usually consider in computer experiments anisotropic (stationary) covariance, the number of hyperparameters linearly increases with the number of inputs. In order to improve the robustness of the optimization process and deal with a large number of inputs, the estimated hyperparameters obtained at the $(j-1)^{th}$ iteration are used, as starting points for the optimization algorithm. This procedure is repeated until the inclusion of all the PII. Note that this sequential estimation process is directly adapted from the one proposed by [22].

V.B. Joint Gp metamodeling to handle the group of non-PII inputs

In the framework of stochastic computer codes, [28] proposed to model the mean and dispersion of the code output by two interlinked Generalized Linear Models (GLM), called "joint GLM". This approach has been extended by [27] to several nonparametric models and the best results on several tests are obtained with two interlinked Gp models, called "joint Gp". In this case, the stochastic input is considered as an uncontrollable parameter denoted $\mathbf{X}_\varepsilon$ (i.e. governed by a seed variable).

We extend this approach to a group of non-explanatory variables. More precisely, the input variables $\mathbf{X} = (X_1, \ldots, X_d)$ are divided in two subgroups: the explanatory ones denoted $\mathbf{X_{exp}}$ and the others denoted $\mathbf{X}_\varepsilon$. The output is thus defined by $y = g(\mathbf{X_{exp}}, \mathbf{X}_\varepsilon)$. Under this hypothesis, the joint metamodeling approach yields building two metamodels, one for the mean Y_m and another for the dispersion component Y_d:

$$Y_m(\mathbf{X_{exp}}) = \mathbb{E}(Y|\mathbf{X_{exp}}) \tag{5}$$

$$Y_d(\mathbf{X_{exp}}) = \text{Var}(Y|\mathbf{X_{exp}}) = \mathbb{E}\left[(Y - Y_m(\mathbf{X_{exp}}))^2|\mathbf{X_{exp}}\right]. \tag{6}$$

To fit these mean and dispersion components, we propose to use the methodology proposed by [27]. To summarize, it consists in the following steps. First, an initial Gp denoted $Gp_{m,1}$ is estimated for the mean component with homoscedastic nugget effect. A nugget effect is required to relax the interpolation property of the Gp metamodel, which would yield zero residuals for the

whole learning sample. Then, a second Gp, denoted $Gp_{v,1}$, is built for the dispersion component with, here also, an homoscedastic nugget effect. $Gp_{v,1}$ is fitted on the squared residuals from the predictor of $Gp_{m,1}$. Its predictor is considered as an estimator of the dispersion component. The predictor of $Gp_{v,1}$ provides an estimation of the dispersion at each point, which is considered as the value of the heteroscedastic nugget effect. The homoscedastic hypothesis is so removed and a new Gp, denoted $Gp_{m,2}$, is fitted on data, with the estimated heteroscedastic nugget. Finally, the Gp on the dispersion component is updated from $Gp_{m,2}$ following the same methodology as for $Gp_{v,1}$.

Remark V.1 Note that some parametric choices are made for all the Gp metamodels: a constant trend and a Matérn stationary anisotropic covariance are chosen. All the hyperparameters (covariance parameters) and the nugget effect (when homoscedastic hypothesis is done) are estimated by maximum likelihood optimization process.

V.C.   Assessment of metamodel accuracy

To evaluate the accuracy of a metamodel, we use the predictivity coefficient Q^2:

$$Q^2 = 1 - \frac{\sum_{i=1}^{n_{\text{test}}} \left(y^{(i)} - \hat{y}^{(i)} \right)^2}{\sum_{i=1}^{n_{\text{test}}} \left(y^{(i)} - \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} y^{(i)} \right)^2} \tag{7}$$

where $(x^{(i)})_{1 \leq i \leq n_{\text{test}}}$ is a test sample, $(y^{(i)})_{1 \leq i \leq n_{\text{test}}}$ are the corresponding observed outputs and $(\hat{y}^{(i)})_{1 \leq i \leq n_{\text{test}}}$ are the metamodel predictions. Q^2 corresponds to the coefficient of determination in prediction and can be computed on a test sample independent from the learning sample or by cross-validation on the learning sample. The closer to one the Q^2, the better the accuracy of the metamodel. In the framework of joint Gp-modeling, Q^2 criterion will be used to assess the accuracy of the mean part of the joint Gp, namely $Gp_{m,\bullet}$, whether $Gp_{m,\bullet}$ is a homoscedastic ($Gp_{m,1}$) or heteroscedastic Gp ($Gp_{m,2}$). This quantitative information can be supplemented with a plot of predicted against observed values ($\hat{y}^{(i)}$ with respect to $y^{(i)}$) or a quantile-quantile plot.

To evaluate the quality of the dispersion part of a joint metamodel (denoted $Gp_{v,\bullet}$), we use the graphical tool introduced in [27] to assess the accuracy of the confidence intervals predicted by a Gp metamodel. For a given Gp metamodel, It consists in evaluating the proportions of observations that lie within the α-theoretical confidence intervals which are built with the mean

squared error of Gp predictions (the whole Gp structure is used and not only the conditional mean). These proportions (i.e. the observed confidence intervals) can be visualized against the α-theoretical confidence intervals, for different values of α.

V.D. Application on IBLOCA test case

The joint Gp metamodel is built from the learning sample of $n = 500$: the eleven PII identified at the end of the the screening step are considered as the explanatory variables while the sixteen others are considered as the uncontrollable parameter. Gps on mean and dispersion components are built using the sequential building process described in section V.A where PII ordered by decreasing R^2_{HSIC} are successively included in Gp. Q^2 coefficient of mean component Gp_m is computed by cross validation at each iteration of the sequential building process. The results which are given by Table III show an increasing predictivity until its stabilization around 0.87, which illustrates the robustness of the Gp building process. The first four PII make the major contribution yielding a Q^2 around 0.8, the four following ones yield minor improvements (increase of 0.02 on average for each input) while the three last PII does not improve the Gp predictivity. Note that, in this application, these results remain unchanged whether we consider homoscedastic Gp ($Gp_{m,1}$) or heteroscedastic Gp ($Gp_{m,2}$), the heteroscedastic nugget effect not significantly improving the Gp predictor for the mean component. Thus, only 13% of the output variability remains here not explained by Gp_m, this includes both the inaccuracy of the Gp_m (part of Y_m not fitted by Gp) and the total effect of the uncontrollable parameter, i.e. the group of non-selected inputs.

TABLE III
Evolution of Gp_m metamodel predictivity during the sequential process building, for each new additional PII.

PII	X_{10}	X_2	X_{12}	X_{22}	X_{15}	X_{13}	X_9	X_5	X_{14}	X_{26}	X_{27}
Q^2	0.60	0.64	0.70	0.79	0.81	0.83	0.85	0.85	0.87	0.87	0.87

For this exercise, 600 remaining CATHARE simulations, different from the learning sample, are also available. As they are not used to build the Gp metamodel, this set of simulations will constitutes a test sample. The Q^2 computed on this test sample is $Q^2 = 0.90$ for both $Gp_{m,1}$ and $Gp_{m,2}$, which is consistent with the estimations by cross-validation.

Now, to assess the quality of dispersion part of the joint Gp, the predicted confidence intervals are compared with the theoretical ones (cf. Section V.C). Figure 5 gives the results obtained by $Gp_{m,1}$ and $Gp_{m,2}$ on the learning sample (by cross-validation) and the test sample, since available here. It clearly illustrates both the interest of considering a heteroscedatic nugget effect and the efficiency of using a joint Gp model to fit and predict this nugget. It can be seen that the joint Gp yields the most accurate confidence intervals in prediction, especially for the test sample. Indeed, all its points are close to the theoretical $y = x$ line (a deviation is only observed for the learning sample for the highest α), while the homoscedastic Gp tends to give too large confidence intervals. Thus, in this case, the heteroscedasticity hypothesis is justified and, consequently, the proposed joint Gp model is clearly more competitive than the simple one.
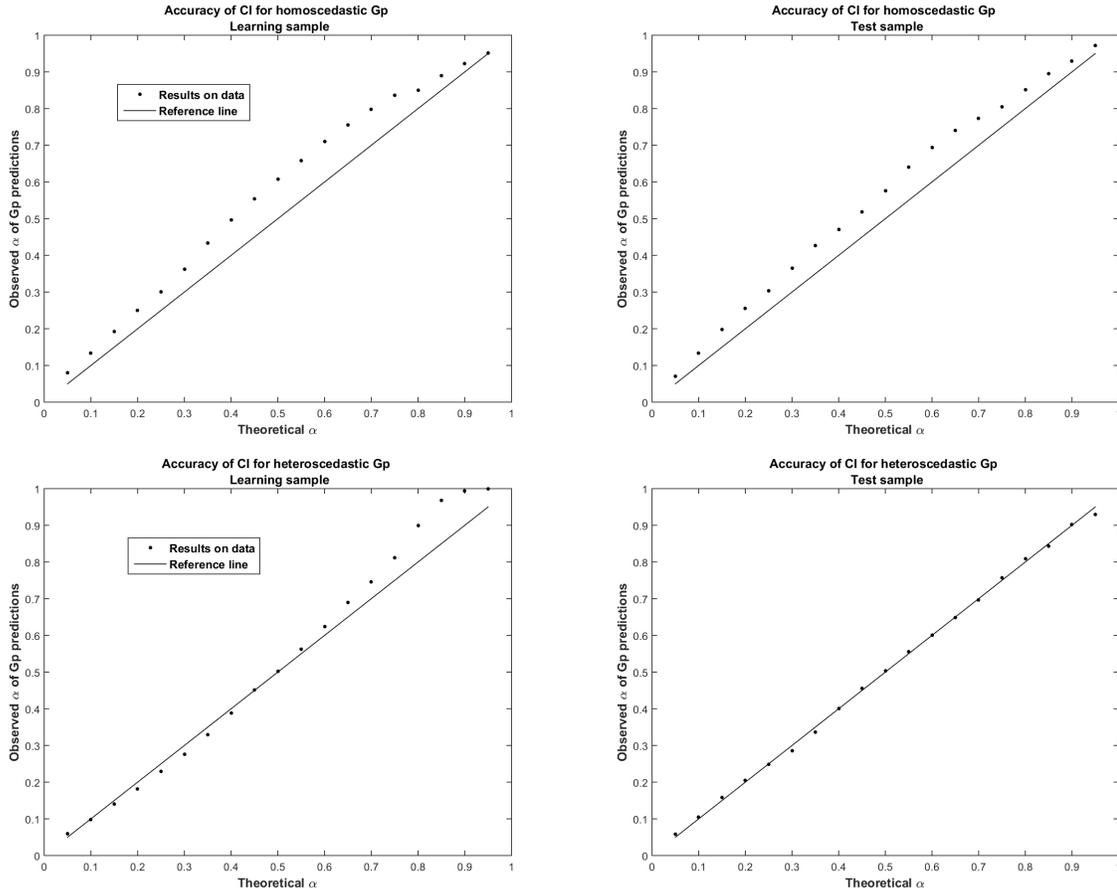

Fig. 5. Proportion of observations that lie within the α-confidence interval predicted by the Gp, according to the theoretical α. Top: results for homoscedastic Gp ($Gp_{m,1}$) on the learning sample (left) and on the test sample (right). Bottom: results for heteroscedastic Gp ($Gp_{m,2}$) on the learning sample (left) and on the test sample (right).

17

VI.   STEP 4A: VARIANCE-BASED SENSITIVITY ANALYSIS

Sensitivity Analysis methods allow to answer the question "How do the input parameters variations contribute, qualitatively or quantitatively, to the variation of the output?" [42]. These tools can detect non-significant input parameters in a screening context, determine the most significant ones, measure their respective contributions to the output or identify an interaction between several inputs which impacts strongly the model output. From this, engineers can guide the characterization of the model by reducing the output uncertainty: for instance, they can calibrate the most influential inputs and fix the non-influential ones to nominal values. Many surveys on sensitivity analysis exist in the literature, such as [43], [44] or [45]. Sensitivity analysis can be divided into two sub-domains: the local sensitivity analysis and the global sensitivity analysis. The first one studies the effects of small input perturbations around nominal values on the model output [46] while the second one considers the impact of the input uncertainty on the output over the whole variation domain of uncertain inputs [42]. We focus here on one of the most widely used global sensitivity indices, namely Sobol' indices which are based on output variance decomposition.

VI.A.   Sobol' indices

A classical approach in global sensitivity analysis consists of computing the first-order and total Sobol' indices which are based on the output variance decomposition [47, 48]. If the variables $X_1, \ldots, X_d$ are independent and if $\mathbb{E}[g^2(X)] < +\infty$, we can apply the Hoeffding decomposition to the random variable $g(X)$ [49]:

$$g(X) \quad = \sum_{u \subset \{1, \ldots, d\}} g_u(X_u) \tag{8}$$

where $g_\emptyset = \mathbb{E}[g(X)]$, $g_i(X_i) = \mathbb{E}[g(X)|X_i] - g_\emptyset$ and $g_u(X_u) = \mathbb{E}[g(X)|X_u] - \sum_{v \subsetneq u} g_v(X_v)$, with $X_u = (X_i)_{i \in u}$, for all $u \subset \{1, \ldots, d\}$. All the 2^d terms in (8) have zero mean and are mutually uncorrelated with each other. This decomposition is unique and leads to the Sobol' indices. These are the elements of the $g(X)$ variance decomposition according to the different groups of input parameter interactions in (8). More precisely, for each $u \subset \{1, \ldots, d\}$, the first-order and total

Sobol' indices of X_u are defined by:

$$S_u = \frac{\text{Var}\left[g_u(X_u)\right]}{\text{Var}\left[g(X)\right]} \text{ and } S_u^T = \sum_{v \supset u} S_v.$$

S_u represents the part of the output variance explained by X_u, independently from the other inputs, and S_u^T is the part of the output variance explained by X_u considered separately and in interaction with the other input parameters.

In practice, we are usually interested in the first-order sensitivity indices $S_1, \ldots, S_d$, the total ones $S_1^T, \ldots, S_d^T$ and sometimes in the second-order ones S_{ij}, $1 \leq i < j \leq d$. The model g is devoid of interactions if $\sum_{i=1}^d S_i \approx 1$.

Sobol' indices are widely used in global sensitivity analysis because they are easy to interpret and directly usable in a dimension reduction approach. However, their estimation (based on Monte-Carlo methods for example) requires a large number of model evaluations, which is intractable for time expensive computer codes. An classical alternative solution consists in using a metamodel to compute these indices. Note that a connection can be made between the estimation error of Sobol' indices when using a metamodel and the predictivity coefficient Q^2 of the metamodel. Indeed, when the Q^2 is estimated on a probabilized sample of the inputs (in other words when it is computed under the probability distribution of the inputs), it provides an estimation of the part of variance unexplained by the metamodel. This can be kept in mind when interpreting the Sobol' indices estimated with the metamodel.

VI.B.   Sobol' indices with a joint Gp metamodel

In the case where a joint Gp metamodel is used to take into account an uncontrollable input X_ε, we have shown in [27] how to deduce Sobol' indices from this joint metamodel. Indeed, by adopting all the same notations as in Section V.B where $\mathbf{X_{exp}}$ is the vector of all the controllable (or explanatory) inputs and X_ε denotes the uncontrollable input (group of non explanatory inputs), we can show that the variance of the output variable $Y(\mathbf{X_{exp}}, X_\varepsilon)$ can be rewritten and deduced from the two components Y_m and Y_d (equations (5) and (6)):

$$\text{Var}[Y(\mathbf{X_{exp}}, X_\varepsilon)] = \text{Var}_{\mathbf{X_{exp}}}\left[Y_m(\mathbf{X_{exp}})\right] + \mathbb{E}_{\mathbf{X_{exp}}}\left[Y_d(\mathbf{X_{exp}})\right] \tag{9}$$

where $\mathbb{E}_X$ (resp. Var_X) denotes the mean (resp. variance) operator with respect to the probability density function of X. Furthermore, the variance of Y is the sum of the contributions of both all the explanatory inputs in $\mathbf{X_{exp}}$ and X_ε:

$$\mathrm{Var}(Y) = V_\varepsilon(Y) + \sum_{u \subset \mathbf{X_{exp}}} [V_u(Y) + V_{u\varepsilon}(Y)] \tag{10}$$

where $V_\varepsilon(Y) = \mathrm{Var}_{X_\varepsilon}[\mathbb{E}_{\mathbf{X_{exp}}}(Y|X_\varepsilon)]$, $V_u(Y) = \mathrm{Var}_{\mathbf{X_u}}[\mathbb{E}_{\mathbf{X_{-u}}}(Y|X_u)] - \sum_{v \subsetneq u} V_v(Y)$, $V_{u\varepsilon}(Y) = \mathrm{Var}_{X_u X_\varepsilon}[\mathbb{E}_{\mathbf{X_{exp}}_{-u}}(Y|X_u X_\varepsilon)] - V_u(Y) - V_\varepsilon(Y)$.

Variance of the mean component $Y_m(\mathbf{X_{exp}})$ denoted hereafter Y_m can be also decomposed:

$$\mathrm{Var}(Y_m) = \sum_{u \subset \mathbf{X_{exp}}} V_u(Y_m) . \tag{11}$$

As $V_u(Y_m) = \mathrm{Var}_{\mathbf{X_u}} \mathbb{E}_{\mathbf{X_{exp}}_{-u}}[\mathbb{E}_{X_\varepsilon}(Y|\mathbf{X_{exp}})|X_u] = V_u(Y)$, Sobol' indices according to any subset of input variables of $\mathbf{X_{exp}}$ can be derived and estimated from Y_m:

$$S_u = \frac{V_u(Y_m)}{\mathrm{Var}(Y)} \text{ for any } u \subset \mathbf{X_{exp}}. \tag{12}$$

Similarly, the total sensitivity index of X_ε is given by:

$$S_\varepsilon^T = \frac{V_\varepsilon(Y) + \sum_{u \subset \mathbf{X_{exp}}} V_{u\varepsilon}(Y)}{\mathrm{Var}(Y)} = \frac{\mathbb{E}_{\mathbf{X_{exp}}}[Y_d(\mathbf{X_{exp}})]}{\mathrm{Var}(Y)} . \tag{13}$$

Note that, as $Y_d(\mathbf{X_{exp}})$ is a positive random variable, the positivity of S_ε^T is guaranteed.

Therefore, from a joint Gp, it is only possible to estimate Sobol' indices of any subset of $\mathbf{X_{exp}}$ (equation (12)) and the total Sobol' index of X_ε (equation 13). The latter is interpreted as the total sensitivity index of the uncontrollable process. The individual index of X_ε or any interaction index involving X_ε are not directly reachable from joint Gp; their contributions in S_ε^T can not be distinguished. This constitutes a limitation of this approach. However, the potential interactions between X_ε and inputs of $\mathbf{X_{exp}}$ could be pointed out, considering all the primary and total effects of all the other parameters. The sensitivity analysis of Y_d can also be a relevant indicator: if a subset X_u of $\mathbf{X_{exp}}$ is not influential on Y_d, we can deduce that $S_{u\varepsilon}$ is equal to zero. Note that in practice, $\mathrm{Var}(Y)$ which appears in both equations (12) and (13) can be estimated directly from

the learning sample (empirical estimator) or from the fitted joint Gp, using equation (9).

VI.C. Results on IBLOCA test case

From the joint Gp built in section V.D, Sobol' indices of PII are estimated from Gp_m by using equation (12), $\text{Var}(Y)$ being estimated with Gp_m and Gp_d using equation (9). For this, intensive Monte Carlo methods are used (see e.g. the pick-and-freeze estimator of [50]): tens of thousands simulations of Gp are done. The obtained first Sobol' indices of PII are given by Table IV and represent 85% of the total variance of the output. Qualitative results of HSIC indices are confirmed and refined: X_{10} (STOIBC1 in Table I) remains the major influential input with 59% of explained variance, followed to a lesser extend by X_{12} (STOIBC3) and X_{22} (CLFBR) with for each of them 8% of variance. The *partial* total Sobol' indices involving only PII and derived from Gp_m show that additional 4% of variance is due to interaction between X_{10}, X_{12} and X_{22}. The related second order Sobol' indices are estimated and the significant ones are given in Table IV. The other PII have negligible influence. In short, the set of PII explain a total 89% of the output variance, of which 79% is only due to X_{10}, X_{12} and X_{22}.

At present, from Gp_d and using equation (13), the total effect of the group of the sixteen non-PII inputs (i.e. X_ε) is estimated to 9.7%. This includes its effect alone and in interaction with the PII. To further investigate these interactions, Sobol indices of Y_d are estimated and HSIC-based statistical dependence tests are applied on Y_d. They reveal that only X_{10}, X_{14}, X_2, X_{22} and X_4 potentially interact with the group of non-PII inputs.

TABLE IV
First and second Sobol' indices of PII (in %), estimated with Gp_m of the joint Gp metamodel.

PII	X_{10}	X_2	X_{12}	X_{22}	X_{15}	X_{13}	X_9	X_5	X_{14}	X_{26}	X_{27}
1st Sobol' index	59	3	8	8	2	1	2	0	2	0	0

Interaction between PII	$X_{10}\times X_{22}$	$X_{10}\times X_{22}$
2nd Sobol' index	3	1

VII.   STEP 4B: QUANTILE ESTIMATION

As already said in the introduction, once a predictive Gp metamodel has been built, it can be used to perform uncertainty propagation and in particular, estimate probabilities or, as here, quantiles.

VII.A.   Gp-based quantile estimation

The most trivial and intuitive approach to estimate a quantile with a Gp metamodel is to apply the quantile definition to the predictor of the metamodel. This direct approach yields a so called *plug-in* estimator. More precisely, with a Gp metamodel, this approach consists in using only the predictor of the Gp metamodel (i.e. the expectation conditional to the learning sample) in order to estimate the quantile. As the expectation of the Gp mean is a deterministic function of the inputs, this provides a deterministic expression of the quantile but no confidence intervals are available. Moreover, for high (resp. low) quantiles, this methods tends to substantially underestimate (resp. overestimate) the *true* quantile because the metamodel predictor is usually constructed by smoothing the computer model output values (see an analysis of this phenomenon in [14]).

To overcome this problem, [51] has proposed to take advantage of the probabilistic-type Gp metamodel by using its entire structure: not only the mean of the conditional Gp metamodel but also its covariance structure are taken into account. In this *full-Gp* based approach also called modular Bayesian approach, the quantile definition is therefore applied to the global Gp metamodel and leads to a random variable. The expectation of this random variable can be then considered as a quantile estimator. Its variance and, more generally, all its distribution can then be used as an indicator of the accuracy of the quantile estimate. Confidence intervals can be deduced, which constitutes a great advantage of this *full-Gp* based approach. Moreover, the efficiency of this approach for high quantile (of the order of 95%) has been illustrated by [52].

In practice, the estimation of quantile with the *full-Gp* approach is based on stochastic simulations (conditionally to the learning sample) of the Gp metamodel, by using the standard method of conditional simulations [53]. We recall just that, to generate conditional Gaussian simulations on a given interval, a discretization of the interval is first considered, then the vector of the conditional expectation and the matrix of the conditional covariance on the discretized

interval are required. Note that [54] uses this approach with Gp conditional simulations for the estimation of Sobol' indices and their associated confidence intervals.

In this paper, from our joint Gp model, we will compare the *full-Gp* approach applied to either the homoscedastic Gp or the heteroscedastic Gp and will proceed as follows:

- For the homoscedastic Gp, the standard technique of conditional simulations is applied to $Gp_{m,1}$ (built to estimate Y_m, with a constant nugget effect).

- For the heteroscedastic Gp, we propose a new technique to simulate the conditional Gp trajectories:

  1. The heteroscedastic Gp built for Y_m (namely $Gp_{m,2}$) provides the conditional expectation vector and a preliminary conditional covariance matrix.

  2. The Gp built for Y_d (namely $Gp_{d,2}$) is predicted and provides the heteroscedastic nugget effect which is added to the diagonal of the conditional covariance matrix of the previous step.

  3. Conditional simulations are then done using the standard method [53].

VII.B. Results on IBLOCA test case

In this section, we focus on the estimation of the 95%-quantile of the PCT for the IBLOCA application. From the learning sample of size $n = 500$ and the joint Gp, we compare here the following approaches to estimate the PCT quantile:

- The classical empirical quantile estimator, denoted $\hat{q}_{95}^{\text{emp}}$. A bootstrap method (see for example [55]) makes it possible to obtain in addition a 90%-confidence interval for this empirical quantile.

- The *plug-in* estimators from the homoscedastic or the heteroscedastic Gp, denoted $\hat{q}_{95}^{Gp_{m,1}\text{-pi}}$ and $\hat{q}_{95}^{Gp_{m,2}\text{-pi}}$. No confidence intervals are obtained using this estimation method.

- The *full-Gp* estimators from the homoscedastic or the heteroscedastic Gp, denoted $\hat{q}_{95}^{Gp_{m,1}\text{-full}}$ and $\hat{q}_{95}^{Gp_{m,2}\text{-full}}$ respectively. As explained in the previous section, confidence intervals can be deduced with this *full-Gp* approach.

Table V synthesizes all the results obtained for the PCT 95%-quantile with these different approaches given above. In addition, 90%-confidence intervals are given when they are available.

As explained in Section V.D, we also have 600 other CATHARE simulations, for a total of 1100 PCT simulations (learning sample plus test sample). A reference value of the 95%-quantile is obtained from this full sample with the classical empirical quantile estimator:

$$\hat{q}_{95}^{\mathrm{ref}} = 742.28 \ ^{\circ}\mathrm{C} \ .$$

The empirical estimator based on the learning sample is imprecise but its bootstrap-based confidence interval is consistent with the reference value. As expected, the *plug-in* approaches provide nasty estimations of the quantile, which is here strongly underestimated. The *full-Gp* approach based on the homoscedastic assumption overestimates the quantile; this is consistent with the analysis of Gp confidence intervals in Figure 5 (too large confidence intervals provided by homoscedastic Gp). Finally, the best result is obtained with the conditional simulation method based on the heteroscedastic Gp metamodel, built with the joint Gp method. This approach yields a more accurate prediction than the usual homoscedastic Gp and outperforms the empirical estimator in terms of confidence interval. Once again, the heteroscedasticity hypothesis is clearly relevant in this case.

TABLE V
Results for the 95%-quantile estimates of the PCT (in °C) with its 90%-confidence interval (CI) when available. The reference method uses the full sample of size $n = 1100$, while the other methods are applied from the learning sample of size $n = 500$. The *plug-in* and *full-Gp* estimators are based on either the homoscedastic or heteroscedastic Gp metamodel.

	Reference	Empirical	*Plug-in*		*Full-Gp* (conditional simulations)	
			homo-Gp	hetero-Gp	homo-Gp	hetero-Gp
	$\hat{q}_{95}^{\mathrm{ref}}$	$\hat{q}_{95}^{\mathrm{emp}}$	$\hat{q}_{95}^{Gp_{m,1}\text{-pi}}$	$\hat{q}_{95}^{Gp_{m,2}\text{-pi}}$	$\hat{q}_{95}^{Gp_{m,1}\text{-full}}$	$\hat{q}_{95}^{Gp_{m,2}\text{-full}}$
Mean	**742.28**	746.80	736.26	735.83	747.11	**741.46**
90%-CI	—	$[736.7; 747.41]$	—	—	$[742.93; 751.32]$	$[\mathbf{738.76}; \mathbf{744.17}]$

VIII.   CONCLUSION

This paper has been focused on an advanced statistical methodology for BEPU analysis, illustrated by a high dimensional thermal-hydraulic test case simulating accidental scenario in a Pressurized Water Reactor (IBLOCA test case). This methodology, which can serve to estimate sensitivity indices and high-level quantiles of the thermal-hydraulic outputs, is mainly based on a predictive joint Gp metamodel, built with an efficient sequential algorithm. First, an initial screening step based on advanced dependence measures and associated statistical tests enabled to identify a group of significant inputs, allowing a reduction of the dimension. The efforts of optimization when fitting the metamodel can then be concentrated on the main influential inputs. The robustness of metamodeling is thus increased. Moreover, thanks to the joint metamodel approach, the non-selected inputs are not completely removed: the residual uncertainty due to dimension reduction is integrated in the metamodel and the global influence of non-selected inputs is so controlled.

From this joint Gp metamodel, several statistical analyses, not feasible with the numerical model due to its computational cost, become accessible. Thus, on the IBLOCA application, a quantitative sensitivity analysis based on variance decomposition is performed using the joint Gp: Sobol' indices are computed and reveal that the output is mainly explained by four uncertain inputs. One input is strongly influential with around 60% of output variance explained, the three others being of minor influence. The quite less influence of all the other inputs is also confirmed.

The last step is to use the joint Gp metamodel to perform an uncertainty analysis of the system operation. To do so, the uncertainties of model inputs have to be propagated inside the computer model to estimate failure probabilities and/or quantiles. In the IBLOCA test case, we are particularly interested by the estimation of the 95%-quantile of the model output temperature. In nuclear safety, as in other engineering domains, methods of conservative computation of quantiles [6, 56] have been largely studied. We have shown in the present work how to use and simulate the joint Gp metamodel to reach the same objective: the uncertainty of the influential inputs are directly and accurately propagated through the mean component of the joint metamodel while a confidence bound is derived from the dispersion component in order to take into account the residual uncertainty of the other inputs. On this last point, the interest of heteroscedastic approach in joint Gp is emphasized: results on the IBLOCA test case are definitely better than the results

with the homoscedastic approach.

As future works, one should mention the use of Gp metamodels to identify penalizing transients in safety analysis, with respect to particular scenario inputs. As in the present work, strong difficulties are raised by the cpu time cost of the computer code and the large number of uncertain (and uncontrollable) inputs.

ACKNOWLEDGMENTS

REFERENCES

[1] M. Pourgol-Mohamad, M. Modarres, and A. Mosleh, "Integrated methodology for thermal-hydraulic code uncertainty analysis with application," *Nuclear Technology*, **165**, 333 (2009).

[2] A. Bucalossi, A. Petruzzi, M. Kristof, and F. D'Auria, "Comparison between best-estimate-plus-uncertainty methods and conservative tools for nuclear power plant licensing," *Nuclear Technology*, **172**, 29 (2010).

[3] A. Prosek and B. Mavko, "The state-of-the-art theory and applications of best-estimate plus uncertainty methods," *Nuclear Technology*, **158**, 69 (2007).

[4] G. Wilson, "Historical insights in the development of Best estimate Plus Uncertainty safety analysis," *Annals of Nuclear Energy*, **52**, 2 (2013).

[5] F. Sanchez-Saez, A. Sànchez, J. Villanueva, S. Carlos, and S. Martorell, "Uncertainty analysis of large break loss of coolant accident in a pressurized water reactor using non-parametric methods," *Reliability Engineering and System Safety*, **174**, 19 (2018).

[6] W. Nutt and G. Wallis, "Evaluation of nuclear safety from the outputs of computer codes in the presence of uncertainties," *Reliability Engineering and System Safety*, **83**, 57 (2004).

[7] G. Wallis, "Uncertainties and probabilities in nuclear reactor regulation," *Nuclear Engineering and Design*, **237**, 1586 (2004).

[8] A. de Crécy, P. Bazin, H. Glaeser, T. Skorek, J. Joucla, P. Probst, K. Fujioka, B. Chung, D. Oh, M. Kyncl, R. Pernica, J. Macek, R. Meca, R. Macian, F. D'Auria, A. Petruzzi, L. Batet, M. Perez, and F. Reventos, "Uncertainty and sensitivity analysis of the LOFT L2-5 test: Results of the BEMUSE programme," *Nuclear Engineering and Design*, **12**, 3561 (2008).

[9] A. Petruzzi and F. D'Auria, "Approaches, relevant topics, and internal method for uncertainty evaluation in predictions of thermal-hydraulic system codes," *Science and Technology of Nuclear Installations*, **2008**, *Article ID 325071, 17 pages, DOI:10.1155/2008/325071* (2008).

[10] R. Martin and W. Nutt, "Perspectives on the application of order-statistics in best-estimate plus uncertainty nuclear safety analysis," *Nuclear Engineering and Design*, **241**, 274 (2011).

[11] E. de Rocquigny, N. Devictor, and S. Tarantola (Editors), *Uncertainty in industrial practice*, Wiley (2008).

[12] K.-T. Fang, R. Li, and A. Sudjianto, *Design and modeling for computer experiments*, Chapman & Hall/CRC (2006).

[13] A. Forrester, A. Sobester, and A. Keane (Editors), *Engineering design via surrogate modelling: a practical guide*, Wiley (2008).

[14] C. Cannamela, J. Garnier, and B. Iooss, "Controlled stratification for quantile estimation," *Annals of Apllied Statistics*, **2**, 1554 (2008).

[15] E. Zio, F. Di Maio, S. Martorell, and Y. Nebot, "Neural network and Order Statistics for Quantifying nuclear power plant Safety Margins," S. Martorell, C. G. Soares, and J. Barnett (Editors), *Safety, reliability and risk analysis - Proceedings of the ESREL 2008 Conference*, 2873–2881, CRC Press (2009).

[16] G. Lorenzo, P. Zanocco, M. Giménez, M.Marquès, B. Iooss, R. Bolado-Lavin, F. Pierro, G. Galassi, F. DAuria, and L. Burgazzi, "Assessment of an isolation condenser of an integral reactor in view of uncertainties in engineering parameters," *Science and Technology of Nuclear Installations*, **2011**, *Article ID 827354, 9 pages, DOI: 10.1155/2011/827354* (2011).

[17] R. Ghanem, D. Higdon, and H. Owhadi (Editors), *Springer Handbook on Uncertainty Quantification*, Springer (2017).

[18] T. Santner, B. Williams, and W. Notz, *The design and analysis of computer experiments*, Springer (2003).

[19] T. Muehlenstaedt, O. Roustant, L. Carraro, and S. Kuhnt, "Data-driven Kriging models based on FANOVA-decomposition," *Statistics & Computing*, **22**, 723 (2012).

[20] N. Durrande, D. G. O., Roustant, and L. Carraro, "ANOVA kernels and RKHS of zero mean functions for model-based sensitivity analysis," *Journal of Multivariate Analysis*, **155**, 57 (2013).

[21] W. Welch, R. Buck, J. Sacks, H. Wynn, T. Mitchell, and M. Morris, "Screening, predicting, and computer experiments," *Technometrics*, **34**, *1*, 15 (1992).

[22] A. Marrel, B. Iooss, F. Van Dorpe, and E. Volkova, "An efficient methodology for modeling complex computer codes with Gaussian processes," *Computational Statistics and Data Analysis*, **52**, 4731 (2008).

[23] D. Woods and S. Lewis, "Design of experiments for screening," R. Ghanem, D. Higdon, and H. Owhadi (Editors), *Springer Handbook on Uncertainty Quantification*, 1143–1185, Springer (2017).

[24] S. Da Veiga, "Global sensitivity analysis with dependence measures," *Journal of Statistical Computation and Simulation*, **85**, 1283 (2015).

[25] M. De Lozzo and A. Marrel, "New improvements in the use of dependence measures for sensitivity analysis and screening," *Journal of Statistical Computation and Simulation*, **86**, 3038 (2016).

[26] B. Iooss and A. Marrel, "An efficient methodology for the analysis and metamodeling of computer experiments with large number of inputs," *Proceedings of UNCECOMP 2017 Conference*, Rhodes Island, Greece (2017).

[27] A. Marrel, B. Iooss, S. Da Veiga, and M. Ribatet, "Global sensitivity analysis of stochastic computer models with joint metamodels," *Statistics and Computing*, **22**, 833 (2012).

[28] I. Zabalza, J. Dejean, and D. Collombier, "Prediction and density estimation of a horizontal well productivity index using generalized linear models," *ECMOR VI, Peebles* (1998).

[29] P. Mazgaj, J.-L. Vacher, and S. Carnevali, "Comparison of CATHARE results with the experimental results of cold leg intermediate break LOCA obtained during ROSA-2/LSTF test 7," *EPJ Nuclear Sciences & Technology*, **2**, *1* (2016).

[30] M. McKay, R. Beckman, and W. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, **21**, 239 (1979).

[31] G. Damblin, M. Couplet, and B. Iooss, "Numerical studies of space filling designs: Optimization of Latin hypercube samples and subprojection properties," *Journal of Simulation*, **7**, 276 (2013).

[32] V. Joseph, E. Gul, and S. Ba, "Maximum projection designs for computer experiments," *Biometrika*, **102**, 371 (2015).

[33] R. Jin, W. Chen, and A. Sudjianto, "An efficient algorithm for constructing optimal design of computer experiments," *Journal of Statistical Planning and Inference*, **134**, 268 (2005).

[34] J. Loeppky, J. Sacks, and W. Welch, "Choosing the sample size of a computer experiment: A practical guide," *Technometrics*, **51**, 366 (2009).

[35] E. Parzen, "On Estimation of a Probability Density Function and Mode," *The Annals of Mathematical Statistics*, **33**, *3*, 1065 (1962).

[36] S. Kucherenko and B. Iooss, "Derivative-Based Global Sensitivity Measures," R. Ghanem, D. Higdon, and H. Owhadi (Editors), *Springer Handbook on Uncertainty Quantification*, 1241–1263, Springer (2017).

[37] O. Roustant, F. Barthe, and B. Iooss, "Poincaré inequalities on intervals - application to sensitivity analysis," *Electronic Journal of Statistics*, **2**, 3081 (2017).

[38] B. Iooss and P. Lemaître, "A review on global sensitivity analysis methods," C. Meloni and G. Dellino (Editors), *Uncertainty management in Simulation-Optimization of Complex Systems: Algorithms and Applications*, 101–122, Springer (2015).

[39] G. Gretton, O. Bousquet, A. Smola, and B. Schölkopf, "Measuring statistical dependence with Hilbert-Schmidt norms," *Proceedings Algorithmic Learning Theory*, 63–77, Springer-Verlag (2005).

[40] J. Sacks, W. Welch, T. Mitchell, and H. Wynn, "Design and analysis of computer experiments," *Statistical Science*, **4**, 409 (1989).

[41] C. Rasmussen and C. Williams, *Gaussian processes for machine learning*, MIT Press (2006).

[42] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Salsana, and S. Tarantola, *Global sensitivity analysis - The primer*, Wiley (2008).

[43] J. Kleijnen, "Sensitivity analysis and related analyses: a review of some statistical techniques," *Journal of Statistical Computation and Simulation*, **57**, 111 (1997).

[44] H. Frey and S. Patil, "Identification and review of sensitivity analysis methods," *Risk Analysis*, **22**, 553 (2002).

[45] J. Helton, J. Johnson, C. Salaberry, and C. Storlie, "Survey of sampling-based methods for uncertainty and sensitivity analysis," *Reliability Engineering and System Safety*, **91**, 1175 (2006).

[46] D. Cacuci, "Sensitivity theory for nonlinear systems. I. Nonlinear functional analysis approach," *Journal of Mathematical Physics*, **22**, 2794 (1981).

[47] I. Sobol, "Sensitivity estimates for non linear mathematical models," *Mathematical Modelling and Computational Experiments*, **1**, 407 (1993).

[48] T. Homma and A. Saltelli, "Importance measures in global sensitivity analysis of non linear models," *Reliability Engineering and System Safety*, **52**, 1 (1996).

[49] W. Hoeffding, "A class of statistics with asymptotically normal distributions," *Annals of Mathematical Statistics*, **19**, 293 (1948).

[50] F. Gamboa, A. Janon, T. Klein, A. Lagnoux, and C. Prieur, "Statistical inference for Sobol pick freeze Monte Carlo methods," *Statistics*, **50**, 881 (2016).

[51] J. Oakley, "Estimating percentiles of uncertain computer code outputs," *Applied Statistics*, **53**, 83 (2004).

[52] B. Rutherford, "A response-modeling alternative to surrogate models for support in computational analyses," *Reliability Engineering and System Safety*, **91**, 1322 (2006).

[53] J.-P. Chilès and P. Delfiner, *Geostatistics: Modeling spatial uncertainty*, Wiley, New-York (1999).

[54] L. Le Gratiet, C. Cannamela, and B. Iooss, "A Bayesian approach for global sensitivity analysis of (multifidelity) computer codes," *SIAM/ASA Journal on Uncertainty Quantification*, **2**, 336 (2014).

[55] E. Zio and F. Di Maio, "Bootstrap and Order Statistics for Quantifying Thermal-Hydraulic Code Uncertainties in the Estimation of Safety Margins," *Science and Technology of Nuclear Installations*, **2008**, *Article ID 340164, 9 pages, DOI:10.1155/2008/340164* (2008).

[56] J. Hessling and J. Uhlmann, "Robustness of Wilks' conservative estimate of confidence intervals," *International Journal for Uncertainty Quantification*, **5**, 569 (2015).