

Global sensitivity analysis of computer models with functional inputs

Bertrand IOOSS* and Mathieu RIBATET†

Submitted to: *Reliability Engineering and System Safety*
for the special SAMO 2007 issue

* CEA, DEN, DER/SESI/LCFR, F-13108 Saint-Paul-lez-Durance, France.

† École Polytechnique Fédérale de Lausanne, Chair of Statistics, STAT-IMA-FSB-EPFL,
Station 8, CH-1015 Lausanne, Switzerland.

Corresponding author: B. Iooss ; Email: bertrand.iooss@cea.fr
Phone: +33 (0)4 42 25 72 73 ; Fax: +33 (0)4 42 25 24 08

Abstract

Global sensitivity analysis is used to quantify the influence of uncertain input parameters on the response variability of a numerical model. The common quantitative methods are appropriate with computer codes having scalar input variables. This paper aims at illustrating different variance-based sensitivity analysis techniques, based on the so-called Sobol's indices, when some input variables are functional, such as stochastic processes or random spatial fields. In this work, we focus on large cpu time computer codes which need a preliminary metamodeling step before performing the sensitivity analysis. We propose the use of the joint modeling approach, i.e., modeling simultaneously the mean and the dispersion of the code outputs using two interlinked Generalized Linear Models (GLM) or Generalized Additive Models (GAM). The "mean model" allows to estimate the sensitivity indices of each scalar input variables, while the "dispersion model" allows to derive the total sensitivity index of the functional input variables. The proposed approach is compared to some classical sensitivity analysis methodologies on an analytical function. Lastly, the new methodology is applied to an industrial computer code that simulates the nuclear fuel irradiation.

Keywords: Sobol's indices, joint modeling, generalized additive model, metamodel, stochastic process, uncertainty

1 INTRODUCTION

Modern computer codes that simulate physical phenomena often take as inputs a high number of numerical parameters and physical variables, and return several outputs - scalars or functions. For the development and the use of such computer models, Sensitivity Analysis (SA) is an invaluable tool. The original technique, based on the derivative computations of the model outputs with respect to the model inputs, suffers from strong limitations for computer models simulating non-linear phenomena. More recent global SA techniques take into account the entire range of variation of the inputs and aim to apportion the whole output uncertainty to the input factor uncertainties (Saltelli et al. [21]). The global SA methods can also be used for model calibration, model validation, decision making process, i.e., any process where it is useful to know which are the variables that mostly contribute to the output variability.

The common quantitative methods are applicable to computer codes with scalar input variables. For example, in the nuclear engineering domain, global SA tools have been applied to numerous models where all the uncertain input parameters are modeled by random variables, possibly correlated - such as thermal-hydraulic system codes (Marquès et al. [13]), waste storage safety studies (Helton et al. [7]), environmental model of dose calculations (Iooss et al. [10]), reactor dosimetry processes (Jacques et al. [11]). Recent research papers have tried to consider more complex input variables in the global SA process, especially in petroleum and environmental studies:

- Tarantola et al. [27] work on an environmental assessment on soil models that use spatially distributed maps affected by random errors. This kind of uncertainty is modeled by a spatial random field (following a specified probability distribution), simulated at each code run. For the SA, the authors propose to replace the spatial input parameter by a “trigger” random parameter ξ that governs the random field simulation. For some values of ξ , the random field is simulated and for the other values, the random field values are put to zero. Therefore, the sensitivity index of ξ is used to quantify the influence of the spatial input parameter.
- Ruffo et al. [18] evaluate an oil reservoir production using a model that depends on different heterogeneous geological media scenarios. These scenarios, which are of limited number, are then substituted for a discrete factor (a scenario number) before performing the SA.

- Iooss et al. [9] study a groundwater radionuclide migration model which depend on several random scalar parameters and on a spatial random field (a geostatistical simulation of the hydrogeological layer heterogeneity). The authors propose to consider the spatial input parameter as an “uncontrollable” parameter. Therefore, they fit on a few simulation results of the computer model a double model, called a joint model: the first component models the effects of the scalar parameters while the second models the effects of the “uncontrollable” parameter.

In this paper, we tackle the problem of the global SA for numerical models and when some input parameters ε are functional. $\varepsilon(\mathbf{u})$ is a one or multi-dimensional stochastic function where \mathbf{u} can be spatial coordinates, time scale or any other physical parameters. Our work focuses on models that depend on scalar parameter vector \mathbf{X} and involve some stochastic process simulations or random fields $\varepsilon(\mathbf{u})$ as input parameters. The computer code output Y depends on the realizations of these random functions. These models are typically non linear with strong interactions between input parameters. Therefore, we concentrate our methodology on the variance based sensitivity indices estimation; that is, the so-called Sobol’s indices (Sobol [25], Saltelli et al. [21]).

To deal with this situation, a first natural approach consists in using either all the discretized values of the input functional parameter $\varepsilon(\mathbf{u})$ or its decomposition into an appropriate basis of orthogonal functions. Then, for all the new scalar parameters related to $\varepsilon(\mathbf{u})$, sensitivity indices are computed. However, in the case of complex functional parameters, this approach seems to be rapidly intractable as these parameters cannot be represented by a small number of scalar parameters (Tarantola et al. [27]). Moreover, when dealing with non physical parameters (for example coefficients of orthogonal functions used in the decomposition), sensitivity indices interpretation may be laborious. Indeed, most often, physicists would prefer to obtain one global sensitivity index related to $\varepsilon(\mathbf{u})$. Finally, a major drawback for the decomposition approach is related to the uncertainty modeling stage. More precisely, this approach needs to specify the probability density functions for the coefficients of the decomposition.

The following section presents three different strategies to compute the Sobol’s indices with functional inputs: (a) the macroparameter method, (b) the “trigger” parameter method and (c) the proposed joint modeling approach. Section 3 compares the relevance of these three strategies on an analytical example: the WN-Ishigami function. Lastly, the proposed approach is illustrated on an industrial computer code simulating fuel irradiation

in a nuclear reactor.

2 COMPUTATIONAL METHODS OF SOBOL'S INDICES

First, let us recall some basic notions about Sobol's indices. Let define the model

$$\begin{aligned} f : \mathbb{R}^p &\rightarrow \mathbb{R} \\ \mathbf{X} &\mapsto Y = f(\mathbf{X}) \end{aligned} \quad (1)$$

where Y is the code output, $\mathbf{X} = (X_1, \dots, X_p)$ are p independent inputs, and f is the model function. f is considered as a "black box", i.e. a function whose analytical formulation is unknown. The main idea of the variance-based SA methods is to evaluate how the variance of an input or a group of input parameters contributes to the output variance of f . These contributions are described using the following sensitivity indices:

$$S_i = \frac{\text{Var}[\mathbb{E}(Y|X_i)]}{\text{Var}(Y)}, \quad S_{ij} = \frac{\text{Var}[\mathbb{E}(Y|X_i X_j)]}{\text{Var}(Y)} - S_i - S_j, \quad S_{ijk} = \dots \quad (2)$$

These coefficients, namely the Sobol's indices, can be used for any complex model functions f . The second order index S_{ij} expresses the model sensitivity to the interaction between the variables X_i and X_j (without the first order effects of X_i and X_j), and so on for higher orders effects. The interpretation of these indices is natural as all indices lie in $[0, 1]$ and their sum is equal to one. The larger an index value is, the greater is the importance of the variable or the group of variables related to this index.

For a model with p inputs, the number of Sobol's indices is $2^p - 1$; leading to an intractable number of indices as p increases. Thus, to express the overall output sensitivity to an input X_i , Homma & Saltelli [8] introduce the total sensitivity index:

$$S_{T_i} = S_i + \sum_{j \neq i} S_{ij} + \sum_{j \neq i, k \neq i, j < k} S_{ijk} + \dots = \sum_{l \in \#i} S_l \quad (3)$$

where $\#i$ represents all the "non-ordered" subsets of indices containing index i . Thus, $\sum_{l \in \#i} S_l$ is the sum of all the sensitivity indices having i in their index. The estimation of these indices (Eqs. (2) and (3)) can be performed by simple Monte-Carlo simulations

based on independent samples (Sobol [24], Saltelli [20]), or by refined sampling designs introduced to reduce the number of required model evaluations significantly, for instance FAST (Saltelli et al. [23]) and quasi-random designs (Saltelli et al. [22]).

Let us now consider a supplementary input parameter which is a functional input variable $\varepsilon(\mathbf{u}) \in \mathbb{R}$ where $\mathbf{u} \in \mathbb{R}^d$ is a d -dimensional location vector. $\varepsilon(\mathbf{u})$ is defined by all its marginal and joint probability distributions. In this work, it is supposed that random function realizations can be simulated. For example, these realizations can be produced using geostatistical simulations (Lantuéjoul [12]) or stochastic processes simulations (Gentle [5]). Our model writes now

$$Y = f(\mathbf{X}, \varepsilon) \tag{4}$$

and in addition to the Sobol's indices related to \mathbf{X} , our goal is to derive methods to compute the sensitivity indices relative to ε , i.e., S_ε (first order index), S_{T_ε} (total sensitivity index), $S_{i\varepsilon}$ (second order indices), $S_{ij\varepsilon}, \dots$

2.1 The macroparameter method

With the macroparameter method, the functional input parameter is not seen as a functional by the computer code. It is discretized in a potentially large number of values (for example several thousands), each of them being an input scalar parameter of the computer code. As all these values come from the functional input parameter (which possesses a specific correlation structure), they can be considered as an ensemble of correlated input parameters. Taking into account correlation between input variables in sensitivity analysis has been a challenging problem, recently solved by a few authors (see Da Veiga et al. [2] for a recent review).

One solution, proposed by Jacques et al. [11], to deal with correlated input parameters, is to consider multi-dimensional sensitivity indices (Sobol [25]): each group of correlated parameters is considered as a multi-dimensional parameter or macroparameter. One therefore performs a sensitivity analysis by groups of correlated parameters. To estimate Sobol indices (first order, second order, \dots , total), a large number of input parameters (correlated and non correlated) have to be generated. As we know how to generate independent samples of a correlated variables group, the simple Monte-Carlo sampling technique can be used (Sobol [24], Saltelli [20]). However, more efficient techniques than simple Monte-

Carlo (in terms of the required size sample), as FAST or quasi Monte-Carlo which use deterministic samples, are prohibited with correlated input variables.

In our context, this approach, using the simple Monte-Carlo algorithm, seems to be relevant as the input functional parameter $\varepsilon(\mathbf{u})$ can be considered as a single multi-dimensional parameter (i.e. a macroparameter). For instance, the first order Sobol's index related to $\varepsilon(\mathbf{u})$ is defined as previously by

$$S_\varepsilon = \frac{\text{Var}[\mathbb{E}(Y|\varepsilon)]}{\text{Var}(Y)} \quad (5)$$

A simple way to estimate $S_\varepsilon = D_\varepsilon/D$ is based on the Sobol [24] algorithm:

$$\hat{f}_0 = \frac{1}{N} \sum_{k=1}^N f(\mathbf{X}_k^{(1)}, \varepsilon_k) \quad (6a)$$

$$\hat{D} = \frac{1}{N-1} \sum_{k=1}^N f^2(\mathbf{X}_k^{(1)}, \varepsilon_k) - \hat{f}_0^2 \quad (6b)$$

$$\hat{D}_\varepsilon = \frac{1}{N-1} \sum_{k=1}^N f(\mathbf{X}_k^{(1)}, \varepsilon_k) f(\mathbf{X}_k^{(2)}, \varepsilon_k) - \hat{f}_0^2 \quad (6c)$$

where $(\mathbf{X}_k^{(1)})_{k=1\dots N}$ and $(\mathbf{X}_k^{(2)})_{k=1\dots N}$ are two independent sets of N simulations of the input vector \mathbf{X} and $(\varepsilon_k)_{k=1\dots N}$ is a sample of N realizations of the random function $\varepsilon(\mathbf{u})$. To compute the sensitivity indices S_i , the same algorithm is used with two independent samples of $(\varepsilon_k)_{k=1\dots N}$. In the same way, the total sensitivity index S_{T_ε} is derived from the algorithm of Saltelli [20].

The major drawback of this method is that it may be cpu time consuming, mainly because of the sampling method. If d is the number of indices to be estimated, the cost of the Sobol's algorithm is $n = N(d+1)$ while the cost of Saltelli's algorithm to estimate d first order and d total sensitivity indices is $n = N(d+2)$. It is well known that, for complex computer models, an accurate estimation of Sobol's indices by the simple Monte-Carlo method (independent random samples) requires $N > 1000$, i.e. more than thousand model evaluations for one input parameter (Saltelli et al. [22]). In complex industrial applications, this approach is intractable due to the cpu time cost of one model evaluation and the possible large number of input parameters.

2.2 The “trigger” parameter method

Dealing with spatially distributed input variables, Tarantola et al. [27] propose an alternative that uses an additional scalar input parameter ξ - called the “trigger” parameter. $\xi \sim U[0, 1]$ governs the random function simulation. More precisely, for each simulation, if $\xi < 0.5$, the functional parameter $\varepsilon(\mathbf{u})$ is fixed to a nominal value $\varepsilon_0(\mathbf{u})$ (for example the mean $\mathbb{E}[\varepsilon(\mathbf{u})]$), while if $\xi > 0.5$, the functional parameter $\varepsilon(\mathbf{u})$ is simulated. Using this methodology, it is possible to estimate how sensitive the model output is to the presence of the random function. Tarantola et al. [27] use the Extended FAST method to compute the first order and total sensitivity indices of 6 scalar input factors and 2 additional “trigger” parameters. For their study, the sensitivity indices according to the “trigger” parameters are small and the authors conclude that it is unnecessary to model these spatial errors more accurately.

Contrary to the previous method, there is no restriction about the sensitivity indices estimation procedure - i.e. Monte-Carlo, FAST, quasi Monte-Carlo. However, there are two major drawbacks for this approach:

- As the macroparameter method, it also requires the use of the computer model to perform the SA and it may be problematic for large cpu time computer models. This problem can be compensated by the use of an efficient quasi Monte-Carlo algorithm for which the sampling design size can be decreased to $N = 100$.
- As underlined by Tarantola et al. [27], ξ reflects only the presence or the absence of the stochastic errors on $\varepsilon_0(\mathbf{u})$. Therefore, the term $\text{Var}[\mathbb{E}(Y|\xi)]$ does not quantify the contribution of the random function variability to the output variability $\text{Var}(Y)$. We will discuss about the significance of $\text{Var}[\mathbb{E}(Y|\xi)]$ later, during our analytical function application.

2.3 The joint modeling approach

To perform a variance-based SA for time consuming computer models, some authors propose to approximate the computer code, starting from an initial small-size sampling design, by a mathematical function often called response surface or metamodel (Marseguerra et al. [14], Volkova et al. [28], Fang et al. [3]). This metamodel, requiring negligible cpu time, is then used to estimate Sobol’s indices by any method, for example the simple Monte-Carlo algorithm. For metamodels with sufficient prediction capabilities, the bias between the

exact Sobol's indices (from the computer code) and the Sobol's indices estimated via the metamodel is negligible. Indeed, it has been shown that the unexplained variance part of the computer code by the metamodel (which can be measured) corresponds to this bias (Sobol [26]). Several choices of metamodel can be found in the literature: polynomials, splines, Gaussian processes, neural networks, . . . The fitting process is often based on least squares regression techniques. Thus, for the functional input problem, one strategy may be to fit a metamodel with a multi-dimensional scalar parameters representing $\varepsilon(\mathbf{u})$ as an input parameter - i.e. its discretization or its decomposition into an appropriate basis. This process would correspond to a metamodeling approach for the macroparameter method. However, this approach seems to be impracticable due to the potential large number of scalar parameters: applying regression techniques supposes to have more observation points (simulation sets) than input parameters and important numerical problem (like matrix conditioning) might occur while dealing with correlated input parameters.

A second option is to substitute each random function realization for a discrete number, which can correspond to the scenario parameter of Ruffo et al. [18] (where the number of geostatistical realizations is finite and fixed, and where each different value of the discrete parameter corresponds to a different realization). Then, a metamodel is fitted using this discrete parameter as a qualitative input variable. However, using a metamodel is interesting when only a few runs of the code is available, which corresponds to a more limited number of realizations of the functional input. This restriction of the possible realizations of the input random function to a few ones is not appropriate in a general context.

Another strategy considers $\varepsilon(\mathbf{u})$ as an uncontrollable parameter. A metamodel is fitted in function of the other scalar parameters \mathbf{X} :

$$Y_m(\mathbf{X}) = \mathbb{E}(Y|\mathbf{X}) \tag{7}$$

Therefore, using the relation

$$\text{Var}(Y) = \text{Var}[\mathbb{E}(Y|\mathbf{X})] + \mathbb{E}[\text{Var}(Y|\mathbf{X})] \tag{8}$$

it can be easily shown that the sensitivity indices of Y given the scalar parameters $\mathbf{X} =$

$(X_i)_{i=1\dots p}$ write (Iooss et al. [9])

$$S_i = \frac{\text{Var}[\mathbb{E}(Y_m|X_i)]}{\text{Var}(Y)}, \quad S_{ij} = \frac{\text{Var}[\mathbb{E}(Y_m|X_i X_j)]}{\text{Var}(Y)} - S_i - S_j, \quad \dots \quad (9)$$

and can be computed by classical Monte-Carlo techniques applied on the metamodel Y_m . Therefore, using equation (8), the total sensitivity index of Y according to $\varepsilon(\mathbf{u})$ corresponds to the expectation of the unexplained part of $\text{Var}(Y)$ by the metamodel Y_m :

$$S_{T_\varepsilon} = \frac{\mathbb{E}[\text{Var}(Y|\mathbf{X})]}{\text{Var}(Y)} \quad (10)$$

Using this approach, our objective is altered because it is impossible to decompose the ε effects into an elementary effect (S_ε) as well as the interaction effects between ε and the scalar parameters $(X_i)_{i=1\dots p}$. However, we see below that our technique allows a qualitative appraisal of the interaction indices.

The sensitivity index estimations from equations (9) and (10) raise two difficulties:

1. It is well known that classical parametric metamodels (based on least squares fitting) are not adapted to estimate $\mathbb{E}(Y|\mathbf{X})$ accurately due to the presence of heteroscedasticity (induced by the effect of ε). Such cases are analyzed by Iooss et al. [9]. The authors show that heteroscedasticity may lead to sensitivity indices misspecifications.
2. Classical non parametric methods, such as Generalized Additive Models (Hastie and Tibshirani [6]) and Gaussian processes (Sacks et al. [19]) that can provide efficient estimation of $\mathbb{E}(Y|\mathbf{X})$ (examples are given in Iooss et al. [9]), even in high dimensional input cases ($p > 5$). However, these approaches are based on a homoscedasticity hypothesis and do not enable the estimation of $\text{Var}(Y|\mathbf{X})$.

To solve the second problem, Zabalza-Mezghani et al. [30] propose the use of a theory developed for experimental data (McCullagh and Nelder [15]): the simultaneous fitting of the mean and the dispersion by two interlinked Generalized Linear Models (GLM), which is called the joint modeling (see Appendix A.1). Besides, to resolve the first problem, this approach has been extended by Iooss et al. [9] to non parametric models. This generalization allows more complexity and flexibility. The authors propose the use of Generalized Additive Models (GAMs) based on penalized smoothing splines (Wood [29]). A succinct description of GAM and joint GAM is given in Appendix A.2. GAMs allow

model and variable selections using quasi-likelihood function, statistical tests on coefficients and graphical display. However, compared to other complex metamodels, GAMS impose an additive effects hypothesis. Therefore, two metamodels are obtained: one for the mean component $Y_m(\mathbf{X}) = \mathbb{E}(Y|\mathbf{X})$; and the other one for the dispersion component $Y_d(\mathbf{X}) = \text{Var}(Y|\mathbf{X})$. The sensitivity indices of \mathbf{X} are computed using Y_m with the standard procedure (Eq. (9)), while the total sensitivity index of $\varepsilon(\mathbf{u})$ is computed from $\mathbb{E}(Y_d)$ (Eq. (10)). Using the model for Y_d as well as the associated regression diagnostics, it is possible to deduce qualitative sensitivity indices for the interactions between $\varepsilon(\mathbf{u})$ and the scalar parameters of \mathbf{X} .

One major assumption of the joint modeling approach is that the “mean response” of the computer code is well handled using Y_m . Consequently, all the unexplained part of the computer model by this metamodel is due to the uncontrollable parameter. In other words, the better the mean component metamodel is, the smaller is the influence of the uncontrollable parameter. This is a strong assumption which has to be validated in order to avoid erroneous results. In fact, some simple statistical and graphical tools can be used while fitting the mean component (Iooss et al. [9]): the explained deviance value, the observed responses versus predicted values plot (and its quantile-quantile plot) and the deviance residuals plot. This last plot allows to detect some fitting problems by revealing possible biases or large residual values. Some examples are given in section 3.2. These tools can also be applied for the dispersion component fit. For a detailed overview of these diagnostic tools, one can refer to McCullagh & Nelder [15].

3 APPLICATION TO AN ANALYTICAL EXAMPLE

The three previously proposed methods are first illustrated on a simple analytical model with two scalar input variables and one functional input:

$$Y = f(X_1, X_2, \varepsilon(t)) = \sin(X_1) + 7 \sin(X_2)^2 + 0.1[\max_t(\varepsilon(t))]^4 \sin(X_1) \quad (11)$$

where $X_i \sim \mathcal{U}[-\pi; \pi]$ for $i = 1, 2$ and $\varepsilon(t)$ is a white noise, i.e. an i.i.d. stochastic process $\varepsilon(t) \sim \mathcal{N}(0, 1)$. In our model simulations, $\varepsilon(t)$ is discretized in one hundred values: $t = 1 \dots 100$. The function (11) is similar to the well-known Ishigami function (Homma and Saltelli [8]) but substitute the third parameter for the maximum of a stochastic pro-

cess. Consequently, we call our function the white-noise Ishigami function (WN-Ishigami). Although the WN-Ishigami function is an analytical model, the introduction of the maximum of a stochastic process inside a model is quite realistic. For example, some computer models simulating physical phenomena can use the maximum of time-dependent variable - river height, rainfall quantity, temperature. Such input variable can be modeled by a temporal stochastic process.

As for the Ishigami function, we can immediately deduce from the formula (11):

$$S_\varepsilon = S_{12} = S_{2\varepsilon} = S_{12\varepsilon} = 0 \quad (12)$$

Then, we have

$$S_{T_1} = S_1 + S_{1\varepsilon}, \quad S_{T_2} = S_2, \quad S_{T_\varepsilon} = S_{1\varepsilon} \quad (13)$$

In the following, we focus our attention on the estimation of S_1 , S_2 and S_{T_ε} .

Because of a particularly complex probability distribution of the maximum of a white noise, there is no analytical solution for the theoretical Sobol's indices S_1 , S_2 and S_{T_ε} for the WN-Ishigami function. Even with the asymptotic hypothesis (number of time steps tending to infinity), where the maximum of the white noise follows Generalized Extreme Value distribution, theoretical indices are unreachable. Therefore, our benchmark Sobol's indices values are derived from the Monte-Carlo method.

3.1 The macroparameter and “trigger” parameter methods

Table 1 contains the Sobol's index estimates using the macroparameter and “trigger” parameter methods. As explained before, we can only use some algorithms based on independent Monte-Carlo samples. We apply the algorithm of Sobol [24] that computes S_1 , S_2 , $S_{1\varepsilon}$ at a cost $n = 4N$ and the algorithm of Saltelli [20] which computes the first order indices S_1 , S_2 and the total sensitivity indices S_{T_1} , S_{T_2} , S_{T_ε} at a cost $n = 5N$ (where N is the size of the Monte-Carlo samples, cf. section 2.1). For the estimation, the size of the Monte-Carlo samples is limited to $N = 10000$ because of memory computer limit. Indeed, the functional input $\varepsilon(\mathbf{u})$ contains for each simulation set 100 values. Then, the input sample matrix has the dimension $N \times 102$ which becomes extremely large when N increases. To evaluate the effect of this limited Monte-Carlo sample size N , each Sobol's index estimate is associated to a standard-deviation estimated by bootstrap (Saltelli et al.

[22]) - with 100 replicates of the input-output sample. The obtained standard-deviations (sd) are relatively small, of the order of 0.01, which is rather sufficient for our exercise.

Remark: We have also tried to estimate Sobol's indices with smaller Monte-Carlo sample sizes N . The order of the obtained standard-deviations (estimated by bootstrap) of the Sobol's estimates are the following: $sd \sim 0.02$ for $N = 5000$, $sd \sim 0.04$ for $N = 1000$ and $sd \sim 0.06$ for $N = 500$. We conclude that the Monte-Carlo estimates are sufficiently accurate for $N > 5000$.

[Table 1 about here.]

Macroparameter

For the macroparameter method, the theoretical relations between indices given in (13) are satisfied. We are therefore confident with the estimates obtained with this method and we choose the Sobol's indices obtained with Saltelli's algorithm as the reference indices:

$$S_1 = 55.1\%, \quad S_2 = 20.7\%, \quad S_{T_\varepsilon} = 24.8\%$$

The S_ε , S_{12} , $S_{2\varepsilon}$ and $S_{12\varepsilon}$ indices (Eq. (12)) are not reported in table 1 as estimates are negligible.

Trigger parameter

Using the "trigger" parameter method, the estimates reported in table 1 are quite far from the reference values. The inadequacies are larger than 30% for all the indices, and can be larger than 60% for a few ones (S_2 and S_{T_2}). Moreover, the relations given in (13) are not satisfied at all. Actually, replacing the input parameter $\varepsilon(\mathbf{u})$ by ξ which governs the presence or the absence of the functional input parameter changes the model. When ε is not simulated, it is replaced by its mean (zero) and the WN-Ishigami function becomes $Y = \sin(X_1) + 7 \sin(X_2)^2$. Therefore, the mix of the WN-Ishigami model and this new model perturbs the estimation of the sensitivity indices, even those unrelated to ε (like X_2). In conclusion, the obtained results are in concordance with the expected results.

This result confirms our expectation: sensitivity indices derived from the "trigger" parameter method have not the same sense that the classical ones, i.e., the measure of the contribution of the input parameter variability to the output variable variability. The sensitivity indices obtained with these two methods are unconnected because the "trigger"

parameter method changes the structure of the model.

3.2 The joint modeling approach

We apply now the joint modeling approach which requires an initial input-output sample to fit the joint metamodel - the mean component Y_m and the dispersion component Y_d . For our application, a learning sample size of $n = 500$ was considered; i.e., n independent random samples of $(X_1, X_2, \varepsilon(\mathbf{u}))$ were simulated leading to n observations for Y . Joint GLM and joint GAM fitting procedures are fully described in Iooss et al. [9]. Some graphical residual analyses are particularly useful to check the relevance of the mean and dispersion components of the joint models. In the following, we give the results of the joint models fitting on a learning sample $(X_1, X_2, \varepsilon(\mathbf{u}), Y)$. Let us recall that we fit a model to predict Y in function of (X_1, X_2) .

Joint GLM fitting

For the joint GLM, a fourth order polynomial for the parametric form of the model is considered. Moreover, only the explanatory terms are retained in our regression model using analysis of deviance and the Fisher statistics. The equation of the mean component writes:

$$Y_m = 1.77 + 4.75X_1 + 1.99X_2^2 - 0.51X_1^3 - 0.26X_2^4 . \quad (14)$$

The value estimates, standard-deviation estimates and Student test results on the regression coefficients are given in table 2. Residuals graphical analysis makes it also possible to appreciate the model goodness-of-fit.

[Table 2 about here.]

The explained deviance of this model is $D_{expl} = 73\%$. It can be seen that it remains 27% of non explained deviance due to the model inadequacy and/or to the functional input parameter. The predictivity coefficient, i.e. coefficient of determination R^2 computed on a test sample, is $Q_2 = 70\%$. Q_2 is relatively coherent with the explained deviance.

For the dispersion component, using analysis of deviance techniques, none significant explanatory variable were found: the heteroscedastic pattern of the data has not been retrieved. Thus, the dispersion component is supposed to be constant (see Table 2);

and the joint GLM model is equivalent to a simple GLM - but with a different fitting procedure.

Joint GAM fitting

At present, we try to model the data using a joint GAM. For each component (mean and dispersion), Student test for the parametric part and Fisher statistics for the non parametric part allow us to keep only the explanatory terms (see Table 3). The resulting model is described by the following features:

$$\begin{aligned} Y_m &= 3.76 - 5.54X_1 + s_1(X_1) + s_2(X_2) , \\ \log(Y_d) &= 1.05 + s_{d1}(X_1) , \end{aligned} \tag{15}$$

where $s_1(\cdot)$, $s_2(\cdot)$ and $s_{d1}(\cdot)$ denote three penalized spline smoothing terms.

[Table 3 about here.]

The explained deviance of the mean component is $D_{expl} = 92\%$ and the predictivity coefficient is $Q_2 = 77\%$. Therefore, the joint GAM approach outperforms the joint GLM one. Indeed, the proportion of explained deviance is clearly greater for the GAM model. Even if this is obviously related to an increasing number of parameters; this is also explained as GAMs are more flexible than GLMs. This is confirmed by the increase of the predictivity coefficient - from 70% to 77%. Moreover, due to the GAMs flexibility, the explanatory variable X_1 is identified for the dispersion component. The interaction between X_1 and the functional input parameter $\varepsilon(\mathbf{u})$ which governs the heteroscedasticity of this model is therefore retrieved.

Figure 1 shows that the deviance residuals for the mean component of the joint GAM seem to be more homogeneously dispersed around the x -axis than the deviance residuals of the joint GLM. This leads to a better prediction from the joint GAM on the whole range of the observations.

[Figure 1 about here.]

Sobol's indices

From the joint GLM and the joint GAM, Sobol's sensitivity indices can be computed using equations (9) and (10) - see Table 4. The reference values are extracted from the results of the macroparameter method via Saltelli's algorithm (see Table 1) and from the

WN-Ishigami analytical form (11) (for example we know that $S_{12} = 0$ because there is no interaction between X_1 and X_2). The standard deviation estimates (sd) are obtained from 100 replicates of the Monte-Carlo estimation procedure - which uses $N = 10000$ for the size of the Monte-Carlo samples (see section 2.1). The joint GLM and joint GAM give approximately good estimates of S_1 and S_2 . Despite the joint GLM leads to an acceptable estimation for S_{T_ε} , we will see later that it is fortuitous. The estimation of S_{T_ε} with the joint GAM seems also satisfactory but not accurate. In fact, an efficient modeling of $\text{Var}(Y|\mathbf{X})$ is difficult, which is a common statistical difficulty in heteroscedastic regression problems (Antoniadis & Lavergne [1]).

Another way to estimate the total sensitivity index S_{T_ε} is to compute the unexplained variance of the mean component model given directly by $1 - Q_2$, where Q_2 is the predictivity coefficient of the mean component model. In practical applications, Q_2 can be estimated using leave-one-out or cross validation procedures. In our analytical case, the index is estimated with the former method and leads to a correct estimation - 0.23 instead of 0.25.

[Table 4 about here.]

For the other sensitivity indices, the conclusions draw from the GLM formula are completely erroneous. As the dispersion component is constant, $S_\varepsilon = S_{T_\varepsilon} = 0.268$ while $S_\varepsilon = 0$ in reality. In contrary, the deductions draw from GAM formulas are exact: (X_1, ε) interaction sensitivity is strictly positive ($S_{1\varepsilon} > 0$) because X_1 is active in the dispersion component Y_d , $S_{2\varepsilon} = S_{12\varepsilon} = 0$, $S_{T_2} = S_2$ and $S_{12} = S_{23} = S_{123} = 0$. The drawback of this method is that some indices ($S_{1\varepsilon}$, S_ε and S_{T_1}) remain unknown due to the non separability of the dispersion component effects. However, we can easily deduce some variation intervals which contain these indices: S_ε and $S_{1\varepsilon}$ are smaller than S_{T_ε} while $S_1 + \min(S_{1\varepsilon}) \leq S_{T_1} \leq S_1 + \max(S_{1\varepsilon})$. All these additional information provide qualitative importance measures for the unknown indices.

By estimating Sobol's indices with those obtained from other learning samples, we observe that the estimates are rather dispersed: it seems that the estimates are not robust according to different learning samples for the joint models. To examine this effect, we propose to study two different sample sizes ($n = 200$ and $n = 500$). For each sample size, the distribution of the Sobol's indices estimates is assessed using a bootstrap procedure. Figures 2 and 3 show the results of this investigation, which are particularly convincing.

Several conclusion can be drawn:

- For the joint GAM, the boxplot interquartile interval of each index contains its reference value. In contrary, the joint GLM fails to obtain correct estimates: except for S_1 , the sensitivity reference values are outside the interquartile intervals of the obtained boxplots.
- The superiority of the joint GAM with respect to the joint GLM is corroborated, especially for S_2 and S_{T_ε} .
- The increase of the learning sample size has no effect on the joint GLM results (due to the parametric form of this model). However, for the joint GAM, boxplots widths are strongly reduced from $n = 200$ to $n = 500$. In addition, the mean estimates seem to converge to the reference values.
- As explained before, the estimation of S_{T_ε} using the predictivity coefficient Q_2 is markedly better than through the dispersion component model. This is not the case for the joint GLM. Moreover, we confirm that the previous result of table 4, $S_{T_\varepsilon} = 0.268$, was a good case: with 100 replicates, S_{T_ε} ranges from 0.24 to 0.35 (Figure 3).

[Figure 2 about here.]

[Figure 3 about here.]

In conclusion, this example shows that the joint models, and specially the joint GAM, can adjust rather complex heteroscedastic situations. Of course, additional tests are needed to confirm this result. Moreover, the joint models offer a theoretical basis to compute efficiently global sensitivity indices for models with functional input parameter. Finally, the required number of computer model evaluations is much smaller with the joint modeling method (here $n = 200$ or $n = 500$ gives good results with the joint GAM) compared to the one of Monte-Carlo based techniques. For exemple, using the macroparameter method (cf. section 3.1) and taking $N = 5000$, we need to compute $n = 25000$ model evaluations to estimate first order and total sensitivity indices (via Saltelli's algorithm).

4 APPLICATION TO A NUCLEAR FUEL IRRADIATION SIMULATION

The METEOR computer code, developed within the Fuel Studies Department in CEA Cadarache, studies the thermo-mechanical behavior of the fuel rods under irradiation in a nuclear reactor core. In particular, it computes the fission gas swelling and the cladding creep (Garcia et al. [4]). These two output variables are considered in our analysis. These variables are of fundamental importance for the physical understanding of the fuel behavior and for the monitoring of the nuclear reactor core.

Input parameters of such mechanical models can be evaluated either by database analyses, arguments invoking simplifying hypotheses, expert judgment. All these considerations lead to assign to each input parameter a nominal value associated with an uncertainty. In this study, six uncertain input parameters are considered: the initial internal pressure X_1 , the pellet and cladding radius X_2 , X_3 , the microstructural fuel grain diameter X_4 , the fuel porosity X_5 and the time-dependent irradiation power $P(t)$. X_1, \dots, X_5 are all modeled by Gaussian independent random variables with the following coefficient of variations: $cv(X_1) = 0.019$, $cv(X_2) = 1.22 \times 10^{-3}$, $cv(X_3) = 1.05 \times 10^{-3}$, $cv(X_4) = 0.044$, $cv(X_5) = 0.25$. The last variable $P(t)$ is a temporal function (discretized in 3558 values) and its uncertainty $\varepsilon(t)$ is modelled like a stochastic process. For simplicity, an Additive White Noise (AWN), of uniform law ranging between -5% and $+5\%$, was introduced.

As in the previous application, additionally to its scalar random variables, the model includes an input functional variable $P(t)$. To compute Sobol's indices of this model, we have first tried to use the macroparameter method. We have succeeded to perform the calculations with $N = 1000$ (for the Monte-Carlo sample sizes of Eqs. (6a), (6b) and (6c)). The sensitivity indices estimates have been obtained after 10 computation days and were extremely imprecise, with strong variations between 0 and 1. Because of the required cpu time, an increase of the sample size N to obtain acceptable sensitivity estimates was unconceivable. Therefore, the goal of this section is to show how the use of the joint modeling approach allows to estimate the sensitivity indices of the METEOR model and, in particular, to quantify the functional input variable influence.

500 METEOR calculations were carried out using Monte-Carlo sampling of the input parameters (using Latin Hypercube Sampling). As expected, the AWN on $P(t)$ generates an increase in the standard deviation of the output variables (compared to simulations

without a white noise): 6% increase for the variable *fission gas swelling* and 60% for the variable *cladding creep*.

4.1 Gas swelling

We start by studying the gas swelling model output. With a joint GLM, the following result for Y_m and Y_d were obtained:

$$\begin{cases} Y_m & = -76 - 0.4X_1 + 20X_2 + 8X_4 + 134X_5 + 0.02X_4^2 - 2X_2X_4 - 6X_4X_5 \\ \log(Y_d) & = -2.4X_1 \end{cases} \quad (16)$$

The explained deviance of the mean component is $D_{expl} = 86\%$. As the residual analyses of mean and dispersion components do not show any biases, the resulting model seems satisfactory. The joint GAM was also fitted on these data and led to similar results. Thus, it seems that spline terms are useless and that a joint GLM model is appropriate.

Table 5 shows the results for the Sobol's indices estimation using Monte-Carlo methods applied on the metamodel (16). The standard deviation (*sd*) estimates are obtained from 100 replicates of the Monte-Carlo estimation procedure -which uses 10^5 model computations for one index estimation. It is useless to perform the Monte-Carlo estimation for some indices because they can be deduced from the joint model equations. For example, $S_3 = 0$ (resp. $S_{\varepsilon_2} = 0$) because X_3 (res. X_2) is not involved in the mean (resp. dispersion) component in equation (16). Moreover, we know that $S_{1\varepsilon} > 0$ because X_1 is an explanatory variable inside the dispersion component Y_d . However, this formulation does not allow to have any idea about S_ε which reflects the first order effect of ε . Therefore, some indices are not accessible, such as S_ε and $S_{1\varepsilon}$ non distinguishable inside the total sensitivity index S_{T_ε} . Finally, we can check that $\sum_{i=1}^5 S_i + \sum_{i,j=1, i < j}^5 S_{ij} + S_{T_\varepsilon} = 1$ holds - up to numerical approximations.

It can be seen that X_4 (grain diameter) and X_5 (fuel porosity) are the most influent factors (each one having 40% of influence), and do not interact with the irradiation power $P(t)$ (represented by its uncertainty ε). In addition, the effect of $P(t)$ is not negligible ($S_{T_\varepsilon} = 14\%$) and parameter X_1 (internal pressure) acts mainly with its interaction with $P(t)$. A sensitivity analysis by fixing X_1 could allow us to obtain some information about the first order effect of ε in the model.

4.2 Cladding creep

We study now the cladding creep model output. With a joint GLM, the model for Y_m and Y_d is:

$$\begin{cases} Y_m & = -2.75 + 1.05X_2 - 0.15X_3 - 0.58X_5 \\ \log(Y_d) & = 156052 - 76184X_2 + 9298X_2^2 \end{cases} \quad (17)$$

The explained deviance of the mean component is $D_{expl} = 26\%$. As the residual analyses of mean and dispersion components show some biases, the resulting model is not satisfactory.

For the joint GAM, the spline terms $\{s(X_2), s(X_3), s(X_5)\}$ and $s(X_2)$ are added within the mean component and the dispersion component respectively. The explained deviance of the mean component is $D_{expl} = 29\%$ which is not significantly greater than 26%. However, as the mean component residual biases of the joint GAM are smaller than those observed for the joint GLM, the joint GAM seems to be more relevant than the joint GLM.

Table 5 shows the Sobol's index estimates using Monte-Carlo methods and deductions from the joint model equations. For the joint GLM and joint GAM of the cladding creep, $\sum_{i=1}^5 S_i + \sum_{i,j=1,i < j}^5 S_{ij} + S_{T_\varepsilon} = 1$ holds – up to numerical imprecisions. Due to the proximity of the two joint models, results are similar. This analysis shows that the parameter X_2 (pellet radius) explains 28% of the uncertainty of the cladding creep phenomenon, while the other scalar parameters have negligible influence. The greater part of the cladding creep variance ($S_{T_\varepsilon} = 70\%$) is explained by the irradiation power uncertainty (the AWN). Physicists may be interested in quantifying the interaction influence between the pellet radius and the irradiation power. Unfortunately, this interaction is not available for the moment in our analysis.

[Table 5 about here.]

5 CONCLUSION

This paper has proposed a solution to perform global sensitivity analysis for time consuming computer models which depend on functional input parameters, such as a stochastic process or a random field. Our purpose concerned the computation of variance-based importance measures of the model output according to the uncertain input parameters. We

have discussed a first natural solution which consists in integrating the functional input parameter inside a macroparameter, and using standard Monte-Carlo algorithms to compute sensitivity indices. This solution is not applicable for time consuming computer code. We have discussed another solution, used in previous studies, based on the replacement of the functional input parameter by a “trigger” parameter that governs the integration or not of the functional input uncertainties. However, the estimated sensitivity indices are not the expected ones due to changes in the model structure carrying out by the method itself. Finally, we have proposed an innovative strategy, the joint modeling method, based on a preliminary step of double (and joint) metamodel fitting, which resolves the large cpu time problem of Monte-Carlo methods. It consists in rejecting the functional input parameters in noisy input variables. Then, two metamodels depending only on the scalar random input variables are simultaneously fitted: one for the mean function and one for the dispersion (variance) function.

Tests on an analytical function have shown the relevance of the joint modeling method, which provides all the sensitivity indices of the scalar input parameters and the total sensitivity index of the functional input parameter. In addition, it reveals in a qualitative way the influential interactions between the functional parameter and the scalar input parameters. It would be interesting in the future to be able to distinguish the contributions of several functional input parameters that are currently totally mixed in one sensitivity index. This is the main drawback of the proposed method in its present form.

In an industrial application, the usefulness and feasibility of our methodology has been established. Indeed, other methods are not applicable in this application because of large cpu time of the computer code. To a better understanding of the model behavior, the information brought by the global sensitivity analysis can be very useful to the physicist or the modeling engineer. The joint model can also be useful to propagate uncertainties in complex models, containing input random functions, to obtain some mean predictions with their confidence intervals.

6 ACKNOWLEDGMENTS

This work was supported by the “Simulation” program managed by the CEA/Nuclear Energy Division. We are grateful to P. Obry (CEA Cadarache/Département d’Etude des Combustibles) for the authorization to use the METEOR application. All the statistical

parts of this work have been performed within the R environment and the “sensitivity” and “JointModeling” packages. We are grateful to the referees whose comments significantly helped to improve the paper.

APPENDIX A: JOINT MODELING OF MEAN AND DISPERSION

A.1 Joint Generalized Linear Models

GLMs allow to extend traditional linear models by the use of a distribution which belongs to the exponential family and a link function that connects the explanatory variables to the explained variable (Nelder & Wedderburn [17]). The joint GLM consists in putting a GLM on the mean component of the model and a GLM on the dispersion component of the model.

The mean component is therefore described by:

$$\begin{cases} \mathbb{E}(Y_i) &= \mu_i, & \eta_i = g(\mu_i) = \sum_j x_{ij}\beta_j, \\ \text{Var}(Y_i) &= \phi_i v(\mu_i), \end{cases} \quad (18)$$

where $(Y_i)_{i=1\dots n}$ are independent random variables with mean μ_i ; x_{ij} are the observations of the parameter X_j ; β_j are the regression parameters that have to be estimated; η_i is the mean linear predictor; $g(\cdot)$ is a differentiable monotonous function (called the link function); ϕ_i is the dispersion parameter and $v(\cdot)$ is the variance function. To estimate the mean component, the functions $g(\cdot)$ and $v(\cdot)$ have to be specified. Some examples of link functions are the identity (traditional linear model), root square, logarithm, and inverse functions. Some examples of variance functions are the constant (traditional linear model), identity and square functions.

Within the joint modeling framework, the dispersion parameter ϕ_i is not supposed to be constant as in a traditional GLM, but is supposed to vary according to the model:

$$\begin{cases} \mathbb{E}(d_i) &= \phi_i, & \zeta_i = h(\phi_i) = \sum_j u_{ij}\gamma_j, \\ \text{Var}(d_i) &= \tau v_d(\phi_i), \end{cases} \quad (19)$$

where d_i is a statistic representative of the dispersion, γ_j are the regression parameters

that have to be estimated, $h(\cdot)$ is the dispersion link function, ζ_i is the dispersion linear predictor, τ is a constant and $v_d(\cdot)$ is the dispersion variance function. u_{ij} are the observations of the explanatory variable U_j . The variables (U_j) are generally taken among the explanatory variables of the mean (X_j) , but can also be different. To ensure positivity, a log link function is often chosen for the dispersion component. For the statistic representing the dispersion d , the deviance contribution (which is close to the distribution of a χ^2) is considered. Therefore, as the χ^2 is a particular case of the Gamma distribution, $v_d(\phi) = \phi^2$ and $\tau \sim 2$. In particular, for the Gaussian case, these relations are exact: d is χ^2 distributed and $\tau = 2$.

The joint model is fitted using Extended Quasi-Loglikelihood (EQL) (Nelder & Pregibon [16]) maximization. The EQL behaves as a log-likelihood for both mean and dispersion parameters. This justifies an iterative procedure to fit the joint model. Statistical tools available in the GLM fitting are also available for each component of the joint model: deviance analysis, Student and Fisher tests, residuals graphical analysis. It allows to make some variable selection in order to simplify model expressions.

A.2 Joint Generalized Additive Models

Generalized Additive models (GAM) allow the linear term in the linear predictor $\eta = \sum_j \beta_j X_j$ of equation (18) to be replaced by a sum of smooth functions $\eta = \sum_j s_j(X_j)$ (Hastie & Tibshirani [6]). The $s_j(\cdot)$'s are unspecified functions that are obtained by fitting a smoother to the data, in an iterative procedure. GAMs provide a flexible method for identifying nonlinear covariate effects in exponential family models and other likelihood-based regression models. The fitting of GAM introduces an extra level of iteration in which each spline is fitted in turn assuming the others known. GAM terms can be mixed quite generally with GLM terms in deriving a model.

One common choice for s_j are the smoothing splines, i.e. splines with knots at each distinct value of the variables. In regression problems, smoothing splines have to be penalized in order to avoid data overfitting. Wood [29] has described in details how GAMs can be constructed using penalized regression splines. This approach is particularly appropriate as it allows the integrated model selection using Generalized Cross Validation (GCV) and related criteria, the incorporation of multi-dimensional smooths and relatively well founded inference using the resulting models. Because numerical models often exhibit strong interactions between input parameters, the incorporation of multi-dimensional

smooth (for example the bi-dimensional spline term $s_{ij}(X_i, X_j)$) is particularly important in our context.

GAMs are generally fitted using penalized likelihood maximization. For this purpose, the likelihood is modified by the addition of a penalty for each smooth function, penalizing its “wiggleness”. Namely, the penalized loglikelihood (PL) is defined as:

$$PL = L + \sum_{j=1}^p \lambda_j \int \left(\frac{\partial^2 s_j}{\partial x_j^2} \right)^2 dx_j \quad (20)$$

where L is the loglikelihood function, p is the total number of smooth terms and λ_j are “tuning” constants that compromise between goodness of fit and smoothness. Estimation of these “tuning” constants is generally achieved using the GCV score minimization (Wood [29]).

We have seen that GAMs extend in a natural way GLMs. Iooss et al. [9] have shown how to extend joint GLM to joint GAM. Extension of PL to penalized extended quasi-likelihood (PEQL) is straightforward by substituting the likelihood function PL and the deviance d for their extended quasi counterparts. The fitting procedure of the joint GAM is similar to the one of joint GLM.

References

- [1] A. Antoniadis and C. Lavergne. Variance function estimation in regression by wavelet methods. In A. Antoniadis and G. Oppenheim, editors, *Wavelets and statistics*. Springer, 1995.
- [2] S. Da Veiga, F. Wahl, and F. Gamboa. Local polynomial estimation for sensitivity analysis for models with correlated inputs. *Technometrics*, submitted, 2008. Available at URL: <http://fr.arxiv.org/abs/0803.3504>.
- [3] K-T. Fang, R. Li, and A. Sudjianto. *Design and modeling for computer experiments*. Chapman & Hall/CRC, 2006.
- [4] P. Garcia, C. Struzik, M. Agard, and V. Louche. Mono-dimensional mechanical modelling of fuel rods under normal and off-normal operating conditions. *Nuclear Science and Design*, 216:183–201, 2002.
- [5] J.E. Gentle. *Random number generation and Monte Carlo methods*. Springer, second edition, 2003.
- [6] T. Hastie and R. Tibshirani. *Generalized additive models*. Chapman and Hall, London, 1990.
- [7] J.C. Helton, J.D. Johnson, C.J. Salaberry, and C.B. Storlie. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering and System Safety*, 91:1175–1209, 2006.

- [8] T. Homma and A. Saltelli. Importance measures in global sensitivity analysis of non linear models. *Reliability Engineering and System Safety*, 52:1–17, 1996.
- [9] B. Iooss, M. Ribatet, and A. Marrel. Global sensitivity analysis of stochastic computer models with generalized additive models. *Technometrics*, submitted, 2008. Available at URL: <http://fr.arxiv.org/abs/0802.0443v1>.
- [10] B. Iooss, F. Van Dorpe, and N. Devictor. Response surfaces and sensitivity analyses for an environmental model of dose calculations. *Reliability Engineering and System Safety*, 91:1241–1251, 2006.
- [11] J. Jacques, C. Lavergne, and N. Devictor. Sensitivity analysis in presence of model uncertainty and correlated inputs. *Reliability Engineering and System Safety*, 91:1126–1134, 2006.
- [12] C. Lantuéjoul. *Geostatistical simulations - Models and algorithms*. Springer, 2002.
- [13] M. Marquès, J.F. Pignatell, P. Saignes, F. D’Auria, L. Burgazzi, C. Müller, R. Bolado-Lavin, C. Kirchsteiger, V. La Lumia, and I. Ivanov. Methodology for the reliability evaluation of a passive system and its integration into a probabilistic safety assesment. *Nuclear Engineering and Design*, 235:2612–2631, 2005.
- [14] M. Marseguerra, R. Masini, E. Zio, and G. Cojazzi. Variance decomposition-based sensitivity analysis via neural networks. *Reliability Engineering and System Safety*, 79:229–238, 2003.
- [15] P. McCullagh and J.A. Nelder. *Generalized linear models*. Chapman & Hall, 1989.
- [16] J.A. Nelder and D. Pregibon. An extended quasi-likelihood function. *Biometrika*, 74:221–232, 1987.
- [17] J.A. Nelder and R.W.M. Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society A*, 135:370–384, 1972.
- [18] P. Ruffo, L. Bazzana, A. Consonni, A. Corradi, A. Saltelli, and S. Tarantola. Hydrocarbon exploration risk evaluation through uncertainty and sensitivity analysis techniques. *Reliability Engineering and System Safety*, 91:1155–1162, 2006.
- [19] J. Sacks, W.J. Welch, T.J. Mitchell, and H.P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4:409–435, 1989.
- [20] A. Saltelli. Making best use of model evaluations to compute sensitivity indices. *Computer Physics Communication*, 145:280–297, 2002.
- [21] A. Saltelli, K. Chan, and E.M. Scott, editors. *Sensitivity analysis*. Wiley Series in Probability and Statistics. Wiley, 2000.
- [22] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Salsana, and S. Tarantola. *Global sensitivity analysis - The primer*. Wiley, 2008.
- [23] A. Saltelli, S. Tarantola, and K. Chan. A quantitative, model-independent method for global sensitivity analysis of model output. *Technometrics*, 41:39–56, 1999.
- [24] I.M. Sobol. Sensitivity estimates for non linear mathematical models. *Mathematical Modelling and Computational Experiments*, 1:407–414, 1993.

- [25] I.M. Sobol. Global sensitivity indices for non linear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, 55:271–280, 2001.
- [26] I.M. Sobol. Theorems and examples on high dimensional model representation. *Reliability Engineering and System Safety*, 79:187–193, 2003.
- [27] S. Tarantola, N. Giglioli, N. Jesinghaus, and A. Saltelli. Can global sensitivity analysis steer the implementation of models for environmental assesments and decision-making? *Stochastic Environmental Research and Risk Assesment*, 16:63–76, 2002.
- [28] E. Volkova, B. Iooss, and F. Van Dorpe. Global sensitivity analysis for a numerical model of radionuclide migration from the RRC "Kurchatov Institute" radwaste disposal site. *Stochastic Environmental Research and Risk Assesment*, 22:17–31, 2008.
- [29] S. Wood. *Generalized Additive Models: An Introduction with R*. CRC Chapman & Hall, 2006.
- [30] I. Zabalza-Mezghani, E. Manceau, M. Feraille, and A. Jourdan. Uncertainty management: From geological scenarios to production scheme optimization. *Journal of Petroleum Science and Engineering*, 44:11–25, 2004.

List of Figures

1	Deviance residuals for the joint GLM and the Joint GAM versus the fitted values (WN-Ishigami application). Dashed lines correspond to local polynomial smoothers.	27
2	WN-Ishigami application. Comparison of Sobol's indices estimates for the learning sample size: $n = 200$. For each index, the horizontal line is the reference value.	28
3	WN-Ishigami application. Comparison of Sobol's indices estimates for the learning sample size: $n = 500$. For each index, the horizontal line is the reference value.	29

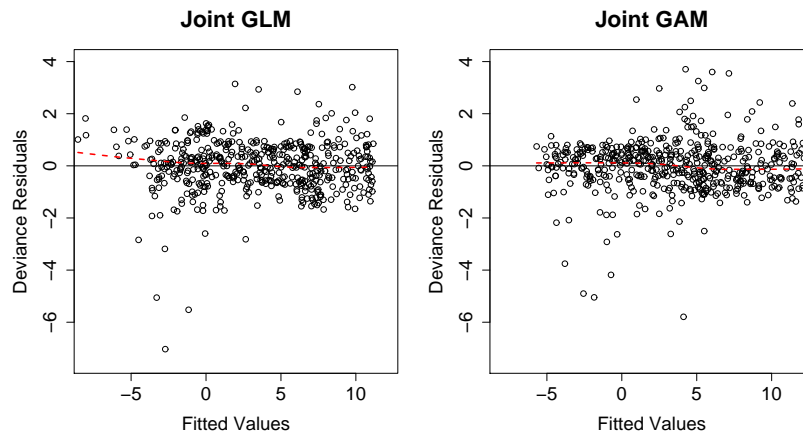


Figure 1: Deviance residuals for the joint GLM and the Joint GAM versus the fitted values (WN-Ishigami application). Dashed lines correspond to local polynomial smoothers.

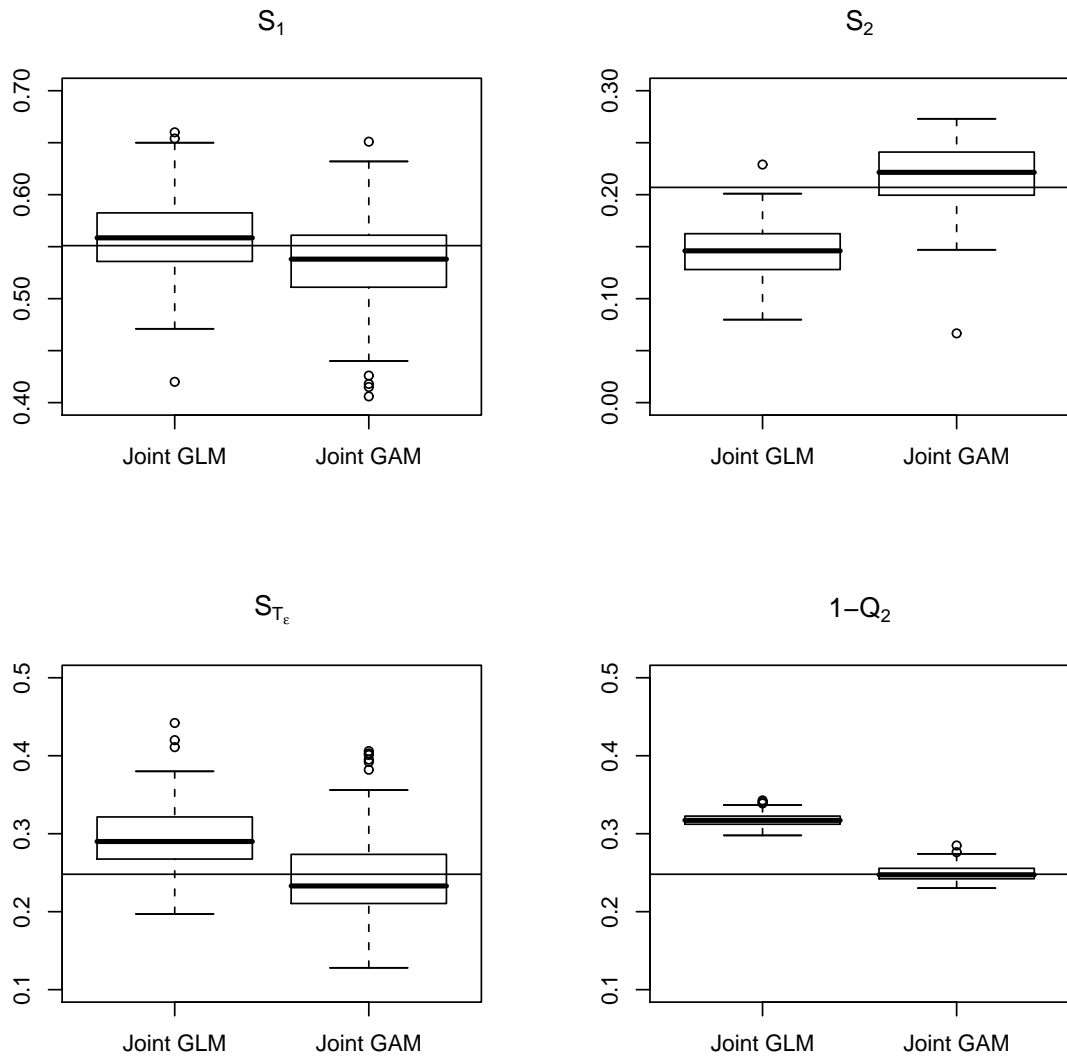


Figure 2: WN-Ishigami application. Comparison of Sobol's indices estimates for the learning sample size: $n = 200$. For each index, the horizontal line is the reference value.

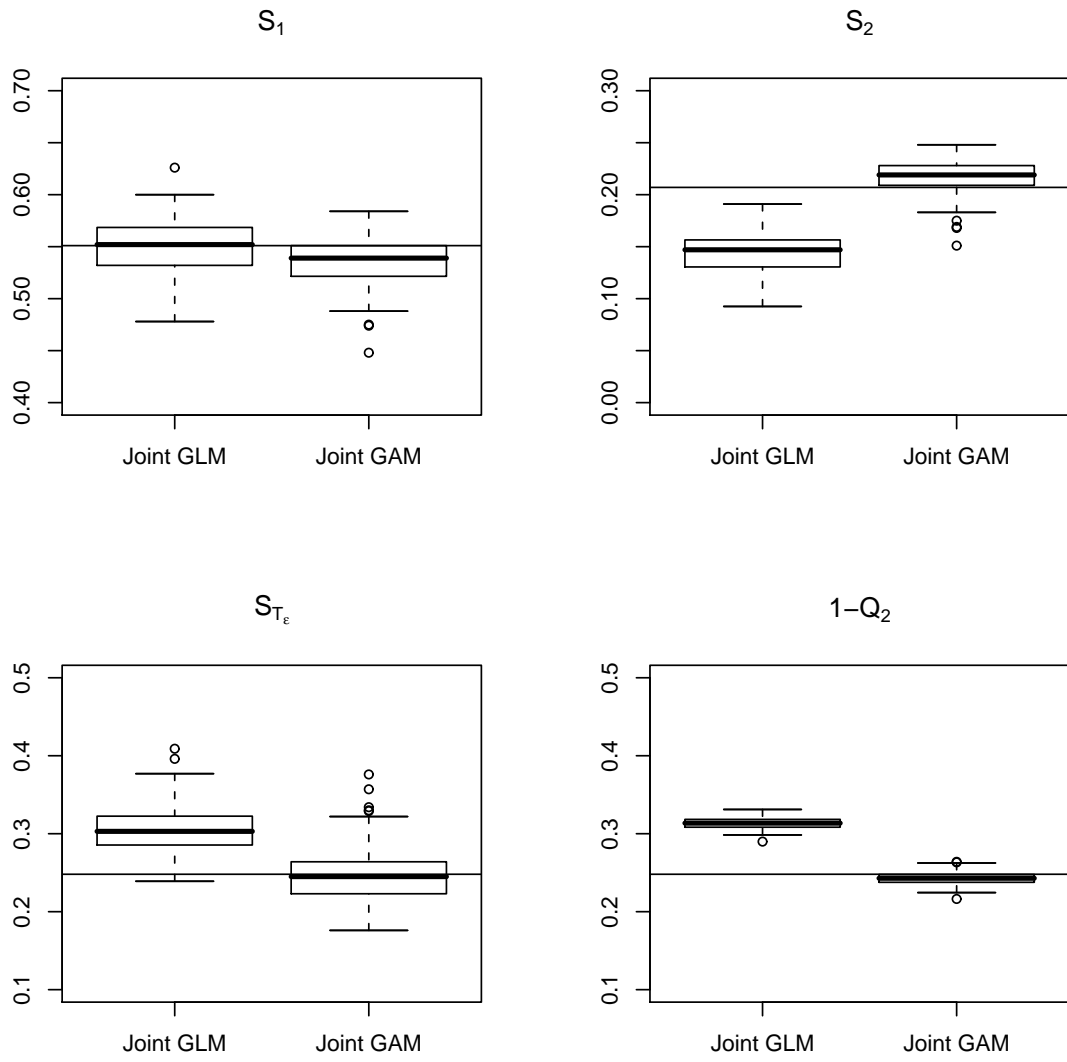


Figure 3: WN-Ishigami application. Comparison of Sobol's indices estimates for the learning sample size: $n = 500$. For each index, the horizontal line is the reference value.

List of Tables

1	Sobol’s sensitivity indices (with standard deviations sd) obtained from two Monte-Carlo algorithms (Sobol [24] and Saltelli [20]) and two integration methods of the functional input ε (macroparameter and “trigger” parameter) on the WN-Ishigami function. “—” indicates that the value is not available.	31
2	For the WN-Ishigami function, summary results of the joint GLM fitting, for the mean component Y_m and the dispersion component Y_d . Estimate standard errors as well as statistics and p-values for the Student’s test are reported.	32
3	For the WN-Ishigami function, summary results of the joint GAM fitting, for the mean component Y_m and the dispersion component Y_d . Estimate standard errors as well as statistics and p-values for the Student’s test are reported. For the smoothing splines, the estimated degree of freedom (edf), the rank of the smoother and the statistics and p-values for the null hypotheses that each smooth term is zero are reported.	33
4	Sobol’s sensitivity indices (with standard deviations) for the WN-Ishigami function: exact and estimated values from joint GLM and joint GAM (fitted with a 500-size sample). “Method” indicates the estimation method: MC for the Monte-Carlo procedure, Eq for a deduction from the model equations and Q_2 for the deduction of the predictivity coefficient Q_2 . “—” indicates that the value is not available.	34
5	Sobol’s sensitivity indices (with standard deviations sd) from joint models fitted on the outputs of the METEOR code. “Method” indicates the estimation method: MC for the Monte-Carlo procedure and Eq for a deduction from the joint model equation. “—” indicates that the value is not available.	35

Table 1: Sobol’s sensitivity indices (with standard deviations sd) obtained from two Monte-Carlo algorithms (Sobol [24] and Saltelli [20]) and two integration methods of the functional input ε (macroparameter and “trigger” parameter) on the WN-Ishigami function. “—” indicates that the value is not available.

Indices	Macroparameter				“Trigger” parameter			
	Sobol’s algo		Saltelli’ algo		Sobol’s algo		Saltelli’ algo	
	Values	sd	Values	sd	Values	sd	Values	sd
S_1	0.540	1.3e-2	0.551	1.6e-2	0.304	1.3e-2	0.330	1.8e-2
S_{T_1}	—	—	0.808	2.0e-2	—	—	0.656	1.4e-2
S_2	0.197	1.1e-2	0.207	0.8e-2	0.329	1.4e-2	0.348	1.5e-2
S_{T_2}	—	—	0.212	0.7e-3	—	—	0.532	1.3e-2
$S_{1\varepsilon}$	0.268	2.4e-2	—	—	0.177	2.2e-2	—	—
S_{T_ε}	—	—	0.248	1.3e-2	—	—	0.336	1.4e-2

Table 2: For the WN-Ishigami function, summary results of the joint GLM fitting, for the mean component Y_m and the dispersion component Y_d . Estimate standard errors as well as statistics and p-values for the Student's test are reported.

Y_m				
	Estimate	Std. Error	t-value	Pr(> t)
(Intercept)	1.77495	0.22436	7.911	1.68e-14
X_1	4.75219	0.16283	29.186	< 2e-16
X_2^2	1.99965	0.14331	13.953	< 2e-16
X_1^3	-0.51254	0.02479	-20.679	< 2e-16
X_2^4	-0.25952	0.01657	-15.659	< 2e-16
$\log(Y_d)$				
	Estimate	Std. Error	t-value	Pr(> t)
(Intercept)	1.9652	0.1373	14.32	<2e-16

Table 3: For the WN-Ishigami function, summary results of the joint GAM fitting, for the mean component Y_m and the dispersion component Y_d . Estimate standard errors as well as statistics and p-values for the Student's test are reported. For the smoothing splines, the estimated degree of freedom (edf), the rank of the smoother and the statistics and p-values for the null hypotheses that each smooth term is zero are reported.

Y_m				
	Estimate	Std. Error	t-value	Pr(> t)
(Intercept)	3.76439	0.09288	40.53	<2e-16
X_1	-5.53920	0.33607	-16.48	<2e-16
	edf	Est.rank	F	p-value
$s_1(X_1)$	5.656	8	151.1	<2e-16
$s_2(X_2)$	8.597	9	411.4	<2e-16
$\log(Y_d)$				
	Estimate	Std. Error	t-value	Pr(> t)
(Intercept)	1.05088	0.07885	13.33	<2e-16
	edf	Est.rank	F	p-value
$s_{d1}(X_1)$	8.781	9	36.09	<2e-16

Table 4: Sobol’s sensitivity indices (with standard deviations) for the WN-Ishigami function: exact and estimated values from joint GLM and joint GAM (fitted with a 500-size sample). “Method” indicates the estimation method: MC for the Monte-Carlo procedure, Eq for a deduction from the model equations and Q_2 for the deduction of the predictivity coefficient Q_2 . “—” indicates that the value is not available.

Indices	Reference		Joint GLM			Joint GAM		
	Values	<i>sd</i>	Values	<i>sd</i>	Method	Values	<i>sd</i>	Method
S_1	0.551	1.6e-2	0.580	3e-3	MC	0.554	4e-3	MC
S_2	0.207	0.8e-2	0.181	7e-3	MC	0.228	6e-3	MC
S_{T_ε}	0.248	1.3e-2	0.268	1e-3	MC	0.218	1e-3	MC
			0.30	—	Q_2	0.23	—	Q_2
S_{12}	0		0	—	Eq	0	—	Eq
$S_{1\varepsilon}$	0.248	1.3e-2	0	—	Eq]0, 0.23]	—	Eq
$S_{2\varepsilon}$	0		0	—	Eq	0	—	Eq
$S_{12\varepsilon}$	0		0	—	Eq	0	—	Eq
S_{T_1}	0.808	2.0e-2	0.580	3e-3	Eq]0.554, 0.784]	—	Eq
S_{T_2}	0.212	0.7e-3	0.181	7e-3	Eq	0.228	6e-3	Eq
S_ε	0		0.268	1e-3	Eq]0, 0.23]	—	Eq

Table 5: Sobol’s sensitivity indices (with standard deviations sd) from joint models fitted on the outputs of the METEOR code. “Method” indicates the estimation method: MC for the Monte-Carlo procedure and Eq for a deduction from the joint model equation. “—” indicates that the value is not available.

Indices	Gas swelling			Cladding creep					
	Joint GLM			Joint GLM			Joint GAM		
	Values	sd	Method	Values	sd	Method	Values	sd	Method
S_1	0.029	6e-3	MC	0.000	1e-3	MC	0.000	1e-3	MC
S_2	0.024	5e-3	MC	0.294	1e-4	MC	0.282	2e-4	MC
S_3	0	—	Eq	0.006	1e-3	MC	0.007	1e-3	MC
S_4	0.394	5e-3	MC	0.000	1e-3	MC	0.000	1e-3	MC
S_5	0.409	6e-3	MC	0.006	1e-3	MC	0.006	1e-3	MC
S_{24}	0.002	5e-3	MC	0	—	Eq	0	—	Eq
S_{45}	0.000	9e-3	MC	0	—	Eq	0	—	Eq
other S_{ij}	0	—	Eq	0	—	Eq	0	—	Eq
S_{T_ε}	0.143	1e-4	MC	0.694	1e-4	MC	0.704	3e-4	MC
S_ε	[0, 0.143]	—	—	[0, 0.694]	—	—	[0, 0.704]	—	—
$S_{1\varepsilon}$]0, 0.143]	—	—	0	—	Eq	0	—	Eq
$S_{2\varepsilon}$	0	—	Eq]0, 0.694]	—	—]0, 0.704]	—	—
other $S_{i\varepsilon}$	0	—	Eq	0	—	Eq	0	—	Eq
S_{T_1}]0.029, 0.172]	—	—	0.000	1e-3	Eq	0.000	4e-3	Eq
S_{T_2}	0.026	7e-3	Eq]0.294, 0.988]	—	—]0.282, 0.986]	—	—
S_{T_3}	0	—	Eq	0.006	1e-3	Eq	0.007	4e-3	Eq
S_{T_4}	0.396	7e-3	Eq	0.000	1e-3	Eq	0.000	4e-3	Eq
S_{T_5}	0.409	0.011	Eq	0.006	1e-3	Eq	0.006	4e-3	Eq