Statistics and Computing, special issue MCEUPSA manuscript No.
(will be inserted by the editor)

Global sensitivity analysis of stochastic computer models with joint metamodels

Amandine MARREL · Bertrand IOOSS · Sébastien DA VEIGA · Mathieu RIBATET

the date of receipt and acceptance should be inserted later

Abstract The global sensitivity analysis method used to quantify the influence of uncertain input variables on the variability in numerical model responses has already been applied to deterministic computer codes; deterministic means here that the same set of input variables gives always the same output value. This paper proposes a global sensitivity analysis methodology for stochastic computer codes, for which the result of each code run is itself random. The framework of the joint modeling of the mean and dispersion of heteroscedastic data is used. To deal with the complexity of computer experiment outputs, nonparametric joint models are discussed and a new Gaussian process-based joint model is proposed. The relevance of these models is analyzed based upon two case studies. Results show that the joint modeling approach yields accurate sensitivity index estimatiors even when heteroscedasticity is strong.

Keywords Computer experiment, Generalized additive model, Gaussian process, Joint modeling, Sobol indices, Uncertainty.

A. Marrel
IFP Energies nouvelles, 1 & 4 avenue de Bois Préau, 92852 Rueil-Malmaison, France
E-mail: amandine.marrel@ifpen.fr Phone: +33-1-47525320 ; Fax: +33-1-47526030

B. Iooss
EDF, R&D, 6 Quai Watier, F-78401, Chatou, France

S. Da Veiga
IFP Energies nouvelles, 1 & 4 avenue de Bois Préau, 92852 Rueil-Malmaison, France

M. Ribatet
Université Montpellier II, Place Bataillon, 34095 Montpellier, France

1 Introduction

Many phenomena are modeled by mathematical equations which are implemented and solved using complex computer codes. These computer models often take as inputs a high number of numerical and physical variables. They can also generate several outputs (scalars or functions). For the development and the analyses of such computer models, the global Sensitivity Analysis (SA) method is an invaluable tool (Saltelli et al. (2000), Kleijnen (2008), Helton (2009)). It accounts for the whole input range of variation, and tries to explain output uncertainties on the basis of input uncertainties. These techniques, which often refer to the probabilistic framework and Monte Carlo methods, require a lot of simulations. The uncertain input variables are modeled by random variables and characterized by their probabilistic density functions. The SA methods are used for model calibration (Kennedy & O'Hagan (2001)), model validation (Bayarri et al. (2007a), (2007b)), decision making process (De Rocquigny et al. (2008)), i.e. all processes where it is useful to know which variables contribute most to output variability .

Current SA methods can handle deterministic computer codes, that is codes providing the same output values for the same input variables. Randomness is limited to model inputs, whereas the model itself is deterministic. For example, global sensitivity analysis tools have been applied to nuclear waste storage safety studies (Helton et al. (2006)) and pollutant transport modeling in aquifers (Volkova et al. (2008)). In such industrial studies, numerical models are often too time consuming, preventing the global

SA methods from being applied at once. To overcome this problem, the time consuming computer code is substituted by an approximate mathematical model, called metamodel (Sacks et al. (1989), Fang et al. (2006)). This function must be as representative as possible of the computer code, with good prediction capabilities. In addition, it must require a negligible calculation time. Several metamodels are classically used: Polynomials, splines, neural networks, Gaussian processes (Chen et al. (2006), Fang et al. (2006)).

This paper does not deal with deterministic computer codes, but focuses on stochastic numerical models - i.e. codes yielding different output values even with identical input variables. In other words, a stochastic model refers to the random simulation case, as introduced by Kleijnen (1997) for a queuing model. Such a computer model is inherently stochastic because the simulator uses random numbers. Contrary to noisy simulations (see for example Yeşilyurt et al. (1996) and Forrester et al. (2006)), a random simulation is not tunable and involves a random seed. The effect of this random seed on the output may be chaotic: a slight variation in the random seed can lead to a very different event realization. In the past, these input variables have been called "discontinuous parameters" (Zabalza et al. (2001)), "stochastic parameters" (Zabalza et al. (2004)), "scenario parameters" (Ruffo et al. (2006)) or "uncontrollable parameters" (Iooss & Ribatet (2009)). To avoid any confusion, we refer now to the generic term "seed variables", since it refers to the original nature of these input variables.

Typical stochastic computer codes are agent-based models (Siebers et al. (2010)), for instance simulating disease propagation (Boukouvalas & Cornford (2009)) or atmospheric pollution (Reich et al. (2009)). There are also models involving partial differential equations applied to heterogeneous random media, for instance fluid flows in oil reservoirs (Zabalza et al. (1998)) or acoustical wave propagation in turbulent fluids (Iooss et al. (2002)). Other examples are the unitary simulations of Monte Carlo neutronic models (computing elementary particle trajectories in a nuclear reactor, Picheny et al. (2011)) and the Lagrangian stochastic models (computing particle trajectories inside atmospheric or hydraulic turbulent media, Pope (1994)).

To approximate stochastic computer codes by metamodels, the simplest way is to model the mean and dispersion (i.e. the variance) of computer code outputs by two polynomial linear-regression models. This is used in the well known context of experimental data modeling under the name of Taguchian model in Response Surface Methodology (Myers et al. (2009)). Polynomial metamodels for robust optimization in deterministic simulation are discussed in Dellino et al. (2010). In stochastic simulation, Zabalza et al. (1998) proposed to model the mean and dispersion (i.e. the variance) of computer code outputs by two interlinked Generalized Linear Models (GLMs). This approach, called joint modeling, was previously studied in the context of experimental data modeling (Smyth (1989), McCullagh & Nelder (1989)). However, the parametric form of GLMs is restrictive for modeling complex computer code outputs. To bypass these limitations, Iooss & Ribatet (2009) suggested to use nonparametric models such as Generalized Additive Models (GAM, see Hastie & Tibshirani (1990), Wood & Augustin (2002)). In this paper, we develop a new joint metamodel, based upon the Gaussian process (Gp) model, which is one of the most relevant choices when dealing with computer experiments (Sacks et al. (1989), Chen et al. (2006)).

Iooss & Ribatet (2009) also developed a method rooted in joint modeling to perform a global sensitivity analysis of computer codes containing a functional input (governed by a seed variable). Their results stressed that the total sensitivity index of the seed variable can be derived just by taking the expectation of the dispersion component of the joint model.

In this paper, we first recap how to build a joint model when referring both to GLM and GAM. An original methodology based upon Gp is then proposed. The third section recalls the variance-based method of global sensitivity analysis for deterministic models, and shows how it can be extended to stochastic models using joint models. Particular attention is paid to the calculation of the so-called Sobol indices. The performance of the different joint metamodels are compared in the next section for a simple analytic function. Last, an industrial application is presented; namely, on a reservoir engineering case.

2 Joint modeling of mean and dispersion

Modeling the mean and variance of a response variable against some explanatory controllable variables is of primary concern in product development and

quality engineering methods. For example in Phadke (1989), experimentation is used to determine factor levels so that the product is insensitive to potential variations in environmental conditions. In the framework of robust design, it is equivalent to the optimization of a mean response function while minimizing a variance function. A first approach consists in building polynomial models approximating the mean and variance separately (Vining & Myers (1990), Bursztyn & Steinberg (2006)), based on repeated calculations with the same set of controllable input variables. This dual modeling approach has been successfully applied in many situations, especially for robust conception problems. However, our purpose here is to fit accurately both mean and dispersion components. Within this context, it has been shown that the dual model is less competitive than the joint model which simultaneously models the mean and variance (Zabalza et al. (1998), Lee & Nelder (2003)). The same authors have also shown that repeating calculations with the same set of controllable variables is inefficient in the joint modeling approach. It is actually recommended to keep all possible experiments to optimally cover the input variable space (which can be highly dimensional in real problems).

In this section, we describe three different joint models based on the metamodels classically used in the context of computer experiments. The computer code output is denoted Y and the random input variables are denoted $\mathbf{X} = (X_1, \ldots, X_p)$. Input random vector $\mathbf{X}$ has a known distribution in a bounded domain $\mathcal{X}$ of $\mathbb{R}^p$.

2.1 Joint Generalized Linear Models

The class of GLM allows us to extend the class of traditional linear models by the use of: (a) a distribution which belongs to the exponential family and (b) a link function which connects the explanatory variables to the explained variable (Nelder & Wedderburn (1972)). The first component of the model concerns the mean while the second one concerns the dispersion. The mean is described as follows:

$$\begin{cases} \mathbb{E}(Y_i) & = \mu_i, \qquad \eta_i = g(\mu_i) = \sum_j x_{ij}\beta_j \ , \\ \mathrm{Var}(Y_i) & = \phi_i v(\mu_i) \ , \end{cases} \tag{1}$$

$(Y_i)_{i=1\ldots n}$ are independent random variables with mean μ_i; x_{ij} are the observations of variable X_j; β_j are the regression parameters which have to be estimated; η_i is the mean linear predictor; $g(\cdot)$ is a differentiable monotonous function called 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 model framework, the dispersion parameter ϕ_i is no longer supposed to be constant as in a traditional GLM. Indeed, it is assumed to vary accordingly to the following model:

$$\begin{cases} \mathbb{E}(d_i) & = \phi_i, \qquad \zeta_i = h(\phi_i) = \sum_j u_{ij}\gamma_j \ , \\ \mathrm{Var}(d_i) & = \tau v_d(\phi_i) \ , \end{cases} \tag{2}$$

d_i is a statistic representative of dispersion, γ_j are regression parameters which 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. Variables (U_j) are generally taken among the explanatory variables of mean (X_j). However, they can also be different. To ensure positivity, $h(\phi) = \log\phi$ is often taken as the dispersion link function. In addition, statistic d representing dispersion is generally considered as the deviance contribution - which is approximately χ^2 distributed. Therefore, as the χ^2 distribution is a particular case of the Gamma distribution, we have $v_d(\phi) = \phi^2$ and $\tau \sim 2$.

Finally, the joint model is fitted by maximizing the Extended Quasi Loglikelihood (EQL, Nelder & Pregibon (1987)). The EQL behaves as a loglikelihood for both mean and dispersion parameters. This justifies an iterative procedure to fit the joint model. First, a GLM is fitted on the mean; then from the estimate of d, another GLM is fitted on the dispersion. Weights for the next estimate of the GLM on the mean are obtained from the estimate of ϕ. This process can be repeated as often as required. Thus, it allows for entirely fitting the joint model (McCullagh & Nelder (1989)).

2.2 Extension to Generalized Additive Models

GAMs were introduced by Hastie & Tibshirani (1990). They extended the linear terms in the predictor expression $\eta = \sum_j \beta_j X_j$ of equation (1) to smooth

functions $\eta = \sum_j s_j(X_j)$. The $s_j(.)$ are unspecified functions obtained from the iterative fit of data by a smoothing function. GAMs provide a flexible method for identifying nonlinear covariate effects in exponential family models and other likelihood-based regression models. Fitting GAMs introduces an extra level of iteration in which each $s_j(.)$ function is alternately fitted assuming the others known. GAM terms can be mixed quite generally with GLM terms in deriving a model.

One common choice for s_j is 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 & Augustin (2002) described in details how GAMs can be constructed using penalized regression splines. Since numerical models often exhibit strong interactions between input variables, the incorporation of multidimensional smooth functions, like bi-dimensional spline terms $s_{ij}(X_i, X_j)$, is particularly important here.

Clearly, GAMs are a natural extension of GLMs. Therefore, in order to overcome limitations of joint GLM on practical cases, Iooss & Ribatet (2009) extended the joint GLM model to a joint GAM one. In equations (1) and (2), the linear predictors are replaced by sums of spline functions.

GAMs are generally fitted using penalized likelihood maximization. For this purpose, the likelihood is modified by adding a penalty term to each smooth function to penalize its wigglyness. More precisely, the penalized loglikelihood is defined as:

$$p\ell = \ell + \sum_{j=1}^{p} \lambda_j \int \left(\frac{\partial^2 s_j}{\partial x_j^2} \right)^2 dx_j \tag{3}$$

where ℓ is the loglikelihood function, p is the total number of smoothing terms and λ_j are penalized parameters which make it possible to balance goodness of fit and smoothness. The estimation of these penalized parameters is generally performed using score minimization and selection by Generalized Cross Validation (GCV) (Hastie & Tibshirani (1990)). Extension to EQL models is straightforward by substituting the likelihood function and deviance d by their EQ analogous. In practice, all smoothing parameters are jointly updated at each iteration of the fitting procedure. Therefore, a GLM/GAM is fitted for each trial set of smoothing parameters at each iteration, while GCV scores are evaluated only at convergence. One drawback of this approach is that the convergence of the algorithm is not ensured.

2.3 Joint Gaussian process modeling

In the computer experiment community, one popular choice of metamodel is the Gaussian process one. This model can be viewed as an extension of the kriging method, a spatial data interpolation method, to computer code data (Sacks et al. (1989)). Gp modeling considers the deterministic response

$$y = f(\mathbf{X}) \tag{4}$$

of the computer code as a realization of a random function $Y_{\mathrm{Gp}}(\mathbf{X})$ defined as follows:

$$Y_{\mathrm{Gp}}(\mathbf{X}) = f_0(\mathbf{X}) + Z(\mathbf{X}) \, . \tag{5}$$

$f_0(\mathbf{X})$ is a deterministic function (for example a polynomial) that provides the mean approximation of the computer code, and $Z(\mathbf{X})$ is a Gaussian centered stationary stochastic process fully characterized by its variance σ^2 and correlation function $R(\cdot)$. Given a learning sample of n simulation points $(X_s, Y_s) = \left((\mathbf{x}^{(1)}, y^{(1)}), \ldots, (\mathbf{x}^{(n)}, y^{(n)})\right)$, the conditional distribution of the response for a new input vector $\mathbf{x}^*$ is a Gaussian distribution with the two following moments :

$$\mathbb{E}[Y_{\mathrm{Gp}}(\mathbf{x}^*)|X_s, Y_s] = f_0(\mathbf{x}^*) + \mathbf{k}(\mathbf{x}^*)^t \mathbf{\Sigma}_s^{-1}(Y_s - F_s), \tag{6}$$

$$\mathrm{Var}[Y_{\mathrm{Gp}}(\mathbf{x}^*)|X_s, Y_s] = \sigma^2 - \mathbf{k}(\mathbf{x}^*)^t \mathbf{\Sigma}_s^{-1} \mathbf{k}(\mathbf{x}^*), \tag{7}$$

with $F_s = f(X_s)$; $\mathbf{k(x^*)}$ is the covariance vector between $\mathbf{x}^*$ and the learning sample and $\mathbf{\Sigma}_s$ the covariance matrix of the learning sample. The conditional mean (Eq. (6)) is used as a predictor and it can be shown, using its analytical expression, that it is an exact interpolator for the points of the learning sample. The variance formula (Eq. (7)) corresponds to the mean squared error (MSE) of this predictor and is also known as the kriging variance. Under the hypothesis of Gp model, this analytical formula for MSE gives a local indicator of the prediction accuracy.

For stochastic computer models, using an exact interpolator as the Gp is not pertinent. This property can be relaxed using a nugget effect. In this case, a constant term ξ ($\xi > 0$) is added to the covariance function of the Gp:

$$\mathrm{Cov}(Y(\mathbf{x}), Y(\mathbf{u})) = \sigma^2 \left(R(\mathbf{x} - \mathbf{u}) + \xi \delta(\mathbf{x} - \mathbf{u}) \right) \tag{8}$$

where $\delta(\mathbf{v}) = \begin{cases} 1 \text{ if } \mathbf{v} = 0, \\ 0 \text{ otherwise.} \end{cases}$

However, doing so, we suppose that the dispersion of the output is the same in the whole input variable domain. This homoscedasticity hypothesis is somewhat limitative and an heteroscedastic nugget effect can be considered. Recently, some authors (e.g. Kleijnen & van Beers (2005), Ginsbourger et al. (2008), Ankerman et al. (2010)) showed the usefulness of Gp for stochastic computer models in heteroscedastic cases. This approach consists in modeling the mean of the computer code with a Gp metamodel for which the nugget effect is assumed to vary with inputs ($\xi(x)$). Referring to the fitted Gp, one can derive the dispersion statistic d introduced in Equation (2) from the estimation of the MSE (given by the Gp model). This model does not include any fitting of the dispersion component but it involves a nugget effect that is different for each point of the learning sample. The dependence between dispersion and inputs is not really explained. Another approach, the treed Gaussian process of Gramacy & Lee (2008) is a fully non stationary model. It is then well-adapted to heteroscedastic computer codes. However, once again, this approach does not allow to obtain a metamodel for both mean and dispersion components.

Therefore, we focus on another method which is more relevant with the previous joint models: the joint Gp model. Robinson et al. (2010) recently proposed a semi-parametric dual modeling approach when there is no replication. Their methodology is based upon a Gp modeling for the mean component and a GLM for the squared residuals, which yields a parametric model for the dispersion. Kersting et al. (2007) and more recently Boukouvalas & Cornford (2009) introduced a joint model with a Gp for both mean and dispersion components. First, a Gp is fitted to the mean component. Its predictive distribution is then used to simulate a sample and compute an estimation of the noise level at each point. The MSE is used to compute several residuals for each point and estimate dispersion. A second Gp model is then fitted on the estimated dispersion. Finally, a combined Gp is deduced from these two Gp models. The process can be repeated until convergence. However, this methodology strongly depends on the MSE formulation and, consequently, on the Gp hypothesis, which is difficult to assess in practice.

In this paper, we prefer to deal only with the residuals observed at each point, after approximating the mean by a Gp model. We propose the following methodology:

– Step1 : Gp modeling of the mean component with homoscedastic nugget effect (8), denoted $Gp_{m,1}$. A nugget effect is required to relax the interpolation property of the Gp metamodel, which would yield zero residuals for the whole learning sample. We choose a first-degree polynomial trend with $f_0(\mathbf{x})$ written as:

$$f_0(\mathbf{x}) = \beta_0 + \sum_{j=1}^{p} \beta_j x_j \ ,$$

where $\boldsymbol{\beta} = [\beta_0, \ldots, \beta_p]^t$ is the regression parameter vector. Such a function was shown to be sufficient to capture the global trend of the computer code (Marrel et al. (2008), Martin & Simpson (2005)). The stochastic part $Z(\mathbf{x})$ is considered as a stationary process. For its correlation function, we propose a multidimensional differentiable exponential (MDE) function. This function, which is an anisotropic extension of the DE correlation function introduced by Chilès and Delfiner (1999) is defined as :

$$R(\mathbf{u}, \mathbf{v}) = \prod_{l=1}^{p} (1 + \theta_l |u_l - v_l|) \exp(-\theta_l |u_l - v_l|)$$

where $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_p]^t$ are the correlation parameters (also called hyperparameters) with $\theta_l \geq 0 \ \forall \ l = 1, \ldots, p$. The MDE correlation function offers a good compromise between the classical exponential and Gaussian correlation functions (it corresponds to the well-known Matérn correlation function with a power hyperparameter equals to 3/2). The MDE correlation is differentiable like the Gaussian one and, consequently, combined with a nugget effect, results in a smooth modeling which is well-suited to extract the mean component from noisy data. Moreover, the MDE correlation function, like the exponential one, tends to reduce the problems of ill-conditioned covariance matrix often observed with Gaussian correlation. Finally, on the analytical test in section 4, the MDE correlation function yields the best results in comparison with exponential and Gaussian ones, which confirms the good properties of MDE correlation function. Note that all Gp hyperparameters and the nugget effect are estimated by maximum likelihood method.

- Step2 : Gp modeling of the dispersion component with homoscedastic nugget effect, denoted $Gp_{v,1}$. We compute the residuals from the predictor of $Gp_{m,1}$. Since there is no replication, we have only one residual for each point of the learning sample. No empirical estimation of the dispersion component can be made. However, the squared residuals can be considered as a realization of a process with the dispersion component as mean function. Consequently, a Gp metamodel with a nugget effect is fitted to the squared residuals. Its predictor is considered as an estimator of the dispersion component. Because of the positivity constraint on any variance estimator, a constant trend is chosen ($f_0(\mathbf{x}) = \beta_0$). A MDE correlation function is used for the same reasons as for $Gp_{m,1}$. Note that the exponential of a first-degree polynomial could also be used for the trend.
- Step3 : Gp modeling of the mean component with heteroscedastic nugget effect, denoted $Gp_{m,2}$. The predictor of $Gp_{v,1}$ provides an estimation of dispersion at each point. It is thus considered as the value of the heteroscedastic nugget effect: the homoscedastic hypothesis is removed. A new Gp, $Gp_{m,2}$, is fitted on data, with the estimated heteroscedastic nugget. The trend and correlation function of $Gp_{m,2}$ are similar to the ones of $Gp_{m,1}$ and the hyperparameters are still estimated by maximum likelihood.
- Step4 : Gp modeling of the dispersion component with homoscedastic nugget effect, denoted $Gp_{v,2}$. The Gp on the dispersion component is updated from $Gp_{m,2}$ following the same methodology as the one described in step 2.

Predictors of $Gp_{m,2}$ and $Gp_{v,2}$ provide respectively an estimator for the mean and dispersion components. This algorithm allows to start from an homoscedastic hypothesis in order to arrive to an heteroscedastic hypothesis, while minimizing the effort of optimization. It is possible to write a full Bayesian model involving two Gps. However, in practice, this theoretical formulation is not tractable, particularly in moderate to high dimension cases. Consequently, we propose an alternative based on a sequential algorithm. Note that the usual solution to deal with heteroscedastic cases is to use a sequential algorithm, like in Zabalza et al. (2001) and Boukouvalas & Cornford (2009). An appealing idea would be to repeat steps 3 and 4 to improve our estimation of the heteroscedastic effect. However, it is not guaranteed that such an iterative procedure converges. From our experience, a single update of $Gp_{m,1}$ and $Gp_{v,1}$, as proposed in the methodology above, is enough to remove the homoscedastic hypothesis.

3 Global sensitivity analysis

This section first considers deterministic, then stochastic simulation. Global SA methods have already been applied to deterministic computer codes. It amounts to considering the following model:

$$\begin{aligned} f : \mathbb{R}^p &\to \mathbb{R} \\ \mathbf{X} &\mapsto Y = f(\mathbf{X}) \end{aligned} \tag{9}$$

where $f(\cdot)$ is the model function (possibly analytically unknown), $\mathbf{X} = (X_1, \ldots, X_p)$ are p independent input random variables with known distribution and Y is the output random variable.

Among quantitative methods, variance-based methods are the most often used (Saltelli et al. (2000)). The main idea of these methods is to evaluate how the variance of an input or a group of inputs contributes to the output variance. To define the sensitivity indices, we use the unique functional ANOVA decomposition of any integrable function on $[0,1]^p$ into a sum of elementary functions (see for example Sobol (1993)):

$$\begin{aligned} f(X_1, \cdots, X_p) = f_0 + \sum_{i=1}^{p} f_i(X_i) + \sum_{i<j}^{p} f_{ij}(X_i, X_j) \\ + \ldots + f_{12..p}(X_1, \cdots, X_p) \, , \end{aligned} \tag{10}$$

where f_0 is a constant and each function of the decomposition respects the following condition:

$$\mathbb{E}[f_J(X_J)] = 0 \, . \tag{11}$$

In the equations above, we have used the usual index set notation. For instance with $J = \{1, 2\}$, X_J means (X_1, X_2), and f_J means f_{12}. Functions f_J are actually related to conditional expectations. We have:

$$f_J(X_J) = \sum_{J' \subset J} (-1)^{|J|-|J'|} \mathbb{E}\left[Y | X_{J'}\right] . \tag{12}$$

The independence of all the X_i ($i = 1, \ldots, p$) ensures that decomposition (10) is unique and we can write the model output variance as (Sobol (1993)):

$$\mathrm{Var}\left[Y\right] = \sum_{i=1}^{p} \sum_{|J|=i} V_J(Y) \, , \tag{13}$$

where $V_i(Y) = \mathrm{Var}[\mathbb{E}(Y|X_i)]$, $V_{ij}(Y) = \mathrm{Var}[\mathbb{E}(Y|X_iX_j)] - V_i(Y) - V_j(Y), \ldots$. Variance-based sensitivity indices, also called Sobol indices, are then defined by:

$$S_J = \frac{V_J(Y)}{\mathrm{Var}(Y)} \ . \tag{14}$$

The second order index S_{ij} expresses the sensitivity of the model to the interaction between variables X_i and X_j and so on for higher orders effects. Interpretation of these indices is straightforward as their sum is equal to one (from equation (13)): the larger an index value, the greater the importance of the variable or the group of variables linked to this index.

For a model with p inputs, the number of Sobol indices is $2^p - 1$. Clearly, the number of indices gets intractable as p increases. Thus, to express the overall sensitivity of the output to an input X_i, Homma & Saltelli (1996) introduce the total sensitivity index:

$$S_{T_i} = \sum_{J \supseteq i} S_J \ . \tag{15}$$

For example, for a model with three input variables, $S_{T_1} = S_1 + S_{12} + S_{13} + S_{123}$.

Estimation of these indices can be done using Monte Carlo simulations or alternative methods (FAST, quasi-Monte Carlo, etc. see Saltelli et al. (2000)). Algorithms were also recently introduced to reduce significantly the number of model evaluations (Saltelli et al. (2010)). As explained in the introduction, a powerful method consists in replacing complex computer models by metamodels with negligible calculation time (e.g. Volkova et al. (2008), Storlie et al. (2009)). Estimation of Sobol indices by Monte Carlo techniques (requiring thousands of simulations) can then be done using these metamodels.

In this work, we do not consider deterministic codes (Eq. (9)), but stochastic ones. Then, we introduce a new input variable X_ε, in addition to inputs $\mathbf{X} = (X_1, \ldots, X_p)$. This additional input, independent of $\mathbf{X}$, is the seed variable discussed in the introduction, and which makes the code stochastic. Thus, our definition of a stochastic model is the following:

$$\begin{aligned} f : \mathbb{R}^p \times \mathbb{N} &\to \mathbb{R} \\ (\mathbf{X}, X_\varepsilon) &\mapsto Y = f(\mathbf{X}, X_\varepsilon) \ . \end{aligned} \tag{16}$$

In practice, except for particular cases, an initial seed variable is selected by the user. The rest of the random number stream is "uncontrollable" because it is managed by the computer code itself (Kelton et al. (2007)). Therefore, classical sensitivity analysis techniques, like Monte Carlo algorithms or metamodels, cannot be used.

However, for a stochastic model as defined by equation (16), joint metamodels (section 2) yield two GLMs, two GAMs or two Gps, one for the mean and another for the dispersion component:

$$Y_m(\mathbf{X}) = \mathbb{E}(Y|\mathbf{X}) \tag{17}$$

$$Y_d(\mathbf{X}) = \mathrm{Var}(Y|\mathbf{X}) = \mathbb{E}\left[(Y - Y_m(\mathbf{X}))^2|\mathbf{X}\right] \ . \tag{18}$$

Referring to the total variance formula, the variance of the output variable Y can be rewritten as:

$$\begin{aligned} \mathrm{Var}(Y) &= \mathrm{Var}\left[\mathbb{E}\left(Y|\mathbf{X}\right)\right] + \mathbb{E}\left[\mathrm{Var}\left(Y|\mathbf{X}\right)\right] \\ &= \mathrm{Var}\left[Y_m(\mathbf{X})\right] + \mathbb{E}\left[Y_d(\mathbf{X})\right] \ . \end{aligned} \tag{19}$$

Furthermore, the variance of Y is the sum of the contributions of all the input variables $\mathbf{X} = (X_1, \ldots, X_p)$ and X_ε:

$$\mathrm{Var}(Y) = V_\varepsilon(Y) + \sum_{i=1}^{p} \sum_{|J|=i} \left[V_J(Y) + V_{J\varepsilon}(Y)\right] \tag{20}$$

where we use the same notations as in equation (13) and $V_\varepsilon(Y) = \mathrm{Var}[\mathbb{E}(Y|X_\varepsilon)]$, $V_{i\varepsilon}(Y) = \mathrm{Var}[\mathbb{E}(Y|X_iX_\varepsilon)] - V_i(Y) - V_\varepsilon(Y), \ldots$

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

$$\mathrm{Var}(Y_m) = \sum_{i=1}^{p} \sum_{|J|=i} V_J(Y_m) \ . \tag{21}$$

Note that

$$\begin{aligned} V_i(Y_m) &= \mathrm{Var}[\mathbb{E}(Y_m|X_i)] \\ &= \mathrm{Var}\{\mathbb{E}[\mathbb{E}(Y|\mathbf{X})|X_i]\} \\ &= \mathrm{Var}[\mathbb{E}(Y|X_i)] = V_i(Y) \ . \end{aligned} \tag{22}$$

Moreover, sensitivity indices for variable Y according to input variables $\mathbf{X} = (X_i)_{i=1\ldots p}$ can be derived from equation (14):

$$S_J = \frac{V_J(Y_m)}{\mathrm{Var}(Y)} \ . \tag{23}$$

These Sobol indices can be computed using the same classical Monte Carlo techniques as for the deterministic model. These algorithms are applied to the metamodel defined by the mean component Y_m of the joint model.

Thus, all terms contained in $\mathrm{Var}[Y_m(\mathbf{X})]$ of equation (19) have been considered. Then, $\mathbb{E}[Y_d(\mathbf{X})]$ can

be estimated by a simple numerical integration of $Y_d(\mathbf{X})$ following the distribution of $\mathbf{X}$. $Y_d(\mathbf{X})$ is evaluated with a metamodel, for example the dispersion component of the joint model. Therefore, the total sensitivity index of X_ε is given by:

$$S_{T_\varepsilon} = \frac{V_\varepsilon(Y) + \sum_{i=1}^{p} \sum_{|J|=i} V_{J\varepsilon}(Y)}{\mathrm{Var}(Y)} = \frac{\mathbb{E}[Y_d(\mathbf{X})]}{\mathrm{Var}(Y)} \ . \tag{24}$$

As $Y_d(\mathbf{X})$ is a positive random variable, positivity of S_{T_ε} is guaranteed. In practice, $\mathrm{Var}(Y)$ can be estimated from the data or from simulations of the fitted joint model, using equation (19). If $\mathrm{Var}(Y)$ is computed from the data, it may be better to estimate $\mathbb{E}[Y_d(\mathbf{X})]$ with $\mathrm{Var}(Y) - \mathrm{Var}[Y_m(\mathbf{X})]$ to satisfy equation (19). In our applications, the total variance will be estimated using the fitted joint model.

In the case of a stochastic computer code whose random nature is due to intrinsic noise, S_{T_ε} has no physical meaning, but can be used as a measure of the stochastic nature of the model. If the seed variable X_ε manages one (or several) stochastic process with a physical significance, S_{T_ε} is interpreted as the total sensitivity index of this stochastic process.

Finally, let us note that we cannot quantitatively distinguish the various contributions in S_{T_ε} (S_ε, $S_{i\varepsilon}$, $S_{ij\varepsilon}$, ...). Indeed, it is not possible to combine the functional ANOVA decomposition of $Y_m(\mathbf{X})$ with the functional ANOVA decomposition of $Y_d(\mathbf{X})$ in order to deduce the unknown sensitivity indices. Forming composite indices still remains an open problem which needs further research. However, Iooss & Ribatet (2009) show that the analysis of the terms in a regression model fitted to Y_d and their t-values gives useful qualitative information. For example, if an input variable X_i is not present in Y_d, we can deduce the following correct information: $S_{i\varepsilon} = 0$. Moreover, if the t-values analysis and the deviance analysis show that an input variable X_i has a smaller influence than another input variable X_j, we can suppose that the interaction between X_i and X_ε is less influential than the interaction between X_j and X_ε. For a joint model which does not yield an explicit regression model for Y_d (like Gp), the same deductions can be made based upon the sensitivity analysis of Y_d. If an input variable X_i is not influential on Y_d, we can deduce that $S_{i\varepsilon}$ is equal to zero.

4 Application and numerical studies on a toy example

The proposed method is first illustrated on an artificial analytical model with three input variables, called the Ishigami function (Homma & Saltelli (1996), Saltelli at al. (2000)):

$$Y(X_1, X_2, X_3) = \sin(X_1) + 7\sin(X_2)^2 + 0.1X_3^4 \sin(X_1) \tag{25}$$

where $X_i \sim \mathcal{U}[-\pi; \pi]$ for $i = 1, 2, 3$. For this function, all the Sobol sensitivity indices (S_1, S_2, S_3, S_{12}, S_{13}, S_{23}, S_{123}, S_{T_1}, S_{T_2}, S_{T_3}) are known. This function is used in most benchmarks of global sensitivity analysis algorithms. In our study, the classical problem is altered by considering X_1 and X_2 as the input random variables, and X_3 as the input generated by the seed variable. It means that the X_3 random values are not used in the modeling procedure; this variable is generated by the seed variable which is considered to be uncontrollable.

However, sensitivity indices have the same theoretical values as in the standard case. For this analytical function case, the analytical expressions of the mean component $Y_m(X_1, X_2)$ and dispersion component $Y_d(X_1, X_2)$ can be directly computed:

$$\begin{aligned} Y_m(X_1, X_2) &= \mathbb{E}(Y|X_1, X_2) \\ &= \left(1 + \frac{\pi^4}{50}\right) \sin(X_1) + 7[\sin(X_2)]^2 \ , \\ Y_d(X_1, X_2) &= \mathrm{Var}(Y|X_1, X_2) \\ &= \pi^8 \left(\frac{1}{900} - \frac{1}{2500}\right) [\sin(X_1)]^2 \ . \end{aligned} \tag{26}$$

Note that the dispersion only depends on the input variable X_1. In this analytical example, only one input variable (X_1) interacts with the uncontrollable one (X_3); see equation (25). As a result, the effect of X_1 on the output is affected by the seed. On the contrary, the effect of the other input (X_2) is not. Such an example, where only one part of the inputs interacts with the uncontrollable parameter, is of particular interest. Indeed, in practice, as illustrated by PUNQ application in section 5, one objective can be to discriminate the input variables between the ones which interact with the uncontrollable and the ones which do not interact.

4.1 Joint metamodeling

To build the learning sample, a Monte Carlo random sampling is used: 500 samples of (X_1, X_2, X_3) are simulated yielding 500 observations for Y. There is no replication in the (X_1, X_2) plane because it has been shown that repeating calculations with the same set of controllable variables is inefficient in the joint modeling approach (Zabalza et al. (1998), Lee & Nelder (2003)). Therefore, we argue that it is better to keep all the possible experiments to optimally cover the input variable space (which can be highly dimensional in real problems). To illustrate this phenomenon, we also consider in section 4.2 a joint Gp metamodel built on a design with replications, following the methodology recently proposed by Ankenman et al. (2010). In practice, to generate the set of controllable variables especially in the case of a high number of variables, Latin hypercubes or quasi-Monte Carlo sequences are preferred to pure Monte Carlo samples (Fang et al. (2006)).

In this section, joint GLM, GAM and Gp models are compared. To evaluate the accuracy of the metamodels for both Y_m and Y_d, we use the predictivity coefficient Q_2. It is the determination coefficient R^2 computed from a test sample (composed here by $n_{\text{test}} = 10000$ randomly chosen points):

$$Q_2(Y, \hat{Y}) = 1 - \frac{\sum_{i=1}^{n_{\text{test}}} \left(Y_i - \hat{Y}_i\right)^2}{\sum_{i=1}^{n_{\text{test}}} \left(\bar{Y} - Y_i\right)^2},$$

where Y denotes the n_{test} true observations (or exact values) of the test set, $\bar{Y}$ their empirical mean and $\hat{Y}$ the metamodel predicted values. For each joint model, two predictivity coefficients are computed using equation (26) to have the exact values: one for Y_m and one for Y_d. The results are given by Table 1.

The GLM for Y_m is a fourth order polynomial. Only the explanatory terms are selected in our regression model using analysis of deviance and the Fisher statistics (McCullagh & Nelder (1989)). For Y_d, using analysis of deviance techniques, only X_1^2 is found as explanatory variable. For the joint GAM estimation, we keep some parametric terms by applying a term selection procedure. The Q_2 results for the mean component show the relevance of GAM and Gp while the GLM is less efficient. The nonparametric models are more accurate and adapted to fit the Ishigami function which is strongly non linear. For the dispersion component, the Q_2 results illustrate the efficiency, even when there is no replication, of joint Gp and GAM models (resp. $Q_2 = 0.91$ and $Q_2 = 0.92$) and the inadequacy of GLM ($Q_2 < 0$). For the GAMs, the explanatory variable X_1 is identified to model Y_d; the interaction between X_1 and X_3, the input generated by the seed variable, is therefore retrieved. For the Gps where no explicit expression is available, we compute the Sobol sensitivity indices of the dispersion component in order to understand which inputs are involved in the dispersion component. We use a Monte Carlo algorithm to obtain $S_{T_1}(Y_d) = 0.999$ and $S_{T_2}(Y_d) = 0.008$. These results draw the same correct conclusion as joint GAM: X_2 is not an explanatory factor for the dispersion and only X_1 interacts with X_3 in the Ishigami function (25).

In order to make a finer comparison between GLM, GAM and Gp models, we examine how well they predict the mean $Y_m(X_1, X_2)$ at inputs for which we have no data. We can also compare the different dispersion models $Y_d(X_1)$. The exact analytical expressions of Y_m and Y_d are given in Eq. (26). Let us remark that we visualize Y_d versus X_1 only because, for GLM and GAM dispersion models, there is no dependence in X_2 and, for the Gp dispersion model, there is an extremely small X_2-dependence (we then take $X_2 = 0$). Figure 1 plots the theoretical Y_m and Y_d surfaces (left panels) and their estimates derived from the fitted joint GLM, joint GAM and joint Gp models. As shown before, the joint GLM is inadequate for both Y_m and Y_d. The joint GAM and Gp fully reproduce Y_m. Spline terms of GAM are perfect smoothers and the MDE correlation function used for the Gp offers good smoothing properties combined with the flexibility of Gp model. It prevents Gp from being impacted by residual noise on the observations. Besides, as it could be expected from its good properties, the MDE correlation function yields the best results in comparison with exponential and Gaussian ones which are not displayed. For Y_d, joint GP results are superior to joint GAM ones. The joint Gp model finely reproduces the behaviour of the dispersion component.

Note that for the two dispersion models in GAMs and Gps, fitted observations have been taken from the mean component residuals on the learning sample. An appealing idea would be to use another solution by taking predicted residuals, for example by applying a cross validation procedure. We tested this approach for the joint Gp and, from our experience, it does not improve the accuracy of the joint models

Table 1 Results for the fitting of different metamodels for the Ishigami function. Both Q_2 for the mean and the dispersion components are given. In the formulas for GAM, $s_1(\cdot)$, $s_2(\cdot)$ and $s_{d1}(\cdot)$ are three spline terms.

	$Q_2(Y_m)$	$Q_2(Y_d)$	Formula
Joint GLM	0.80	-0.61	$Y_m = 2.17 + 2.56X_1 + 1.93X_2^2 - 0.28X_1^3 - 0.25X_2^4$ $\log(Y_d) = 1.78 - 0.04X_1^2$
Joint GAM	0.99	0.92	$Y_m = 3.52 - 2.43X_1 + s_1(X_1) + s_2(X_2)$ $\log(Y_d) = 0.59 + s_{d1}(X_1)$
Joint Gp	0.98	0.91	—

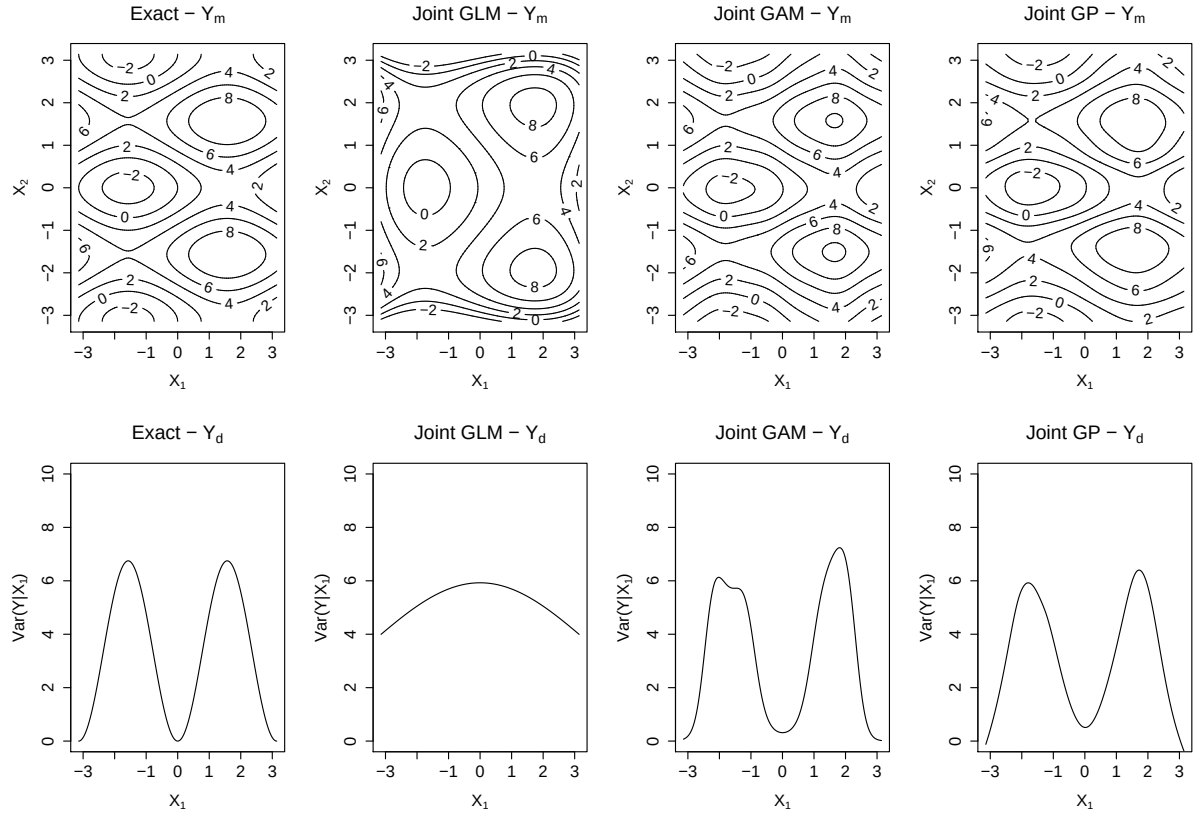

Fig. 1 Mean component (up) and dispersion component (down) for the exact analytical model, Joint GLM, Joint GAM and Joint Gp (Ishigami application).

(even for smaller size of learning sample). Worse, it can make the model estimation less robust.

4.2 Sobol indices

Table 2 depicts Sobol sensitivity indices for the joint GLM, joint GAM and joint Gp based upon equations (23) and (24) and using Monte Carlo estimation procedure. Tens of thousands of joint model computations are made for one index estimation in order to ensure convergence of Monte Carlo estimation. The joint GLM gives only a good estimation of S_1 and S_{12} , while S_2 and S_{T_3} are badly estimated (relative error greater than 50% for S_{T_3}). The joint GAM and GLM give very accurate estimations of all the Sobol indices: negligible error for S_1, S_1, S_{12} and less than 5% of relative error for S_{T_3}. The three joint models correctly show a negligible interaction between X_1 and X_2. These results stress the efficiency of non-parametric models and, for Gp, the interest of developing a robust methodology to use it as a joint model. In conclusion, joint GAM and Gp provide precise estimations of both sensitivity indices of the

Table 2 Sobol sensitivity indices for the Ishigami function: exact and estimated values from joint GLM, joint GAM and joint Gp.

Indices	Theoretical Value	Joint GLM	Joint GAM	Joint Gp
S_1	0.314	0.319	0.310	0.312
S_2	0.442	0.296	0.454	0.450
S_{12}	0	2.10^{-4}	2.10^{-4}	0.004
S_{T_3}	0.244	0.385	0.236	0.233

input variables and total sensitivity index of the seed variable.

As explained at the end of section 3, some conclusions on the various contributions in S_{T_3} can be drawn from the analysis of the dispersion component. For the joint GLM and joint GAM, only X_1 is involved in Y_d (see Table 1). The deduced zero interaction indices are: $S_{23} = S_{123} = 0$. Moreover, it ensures that $S_{13} > 0$. Variation intervals can be deduced from the elementary relations between sensitivity indices (e.g. $S_1 \leq S_{T_1}$, $S_{13} \leq S_{T_3}$, etc). For the joint Gp, a sensitivity analysis of Y_d shows an influence of X_1 higher than 99.9% and yields the qualitative conclusion that X_2 is not influential in Y_d. The same deduced interaction indices as for GLMs and GAMs are made. All the obtained interactions and variation intervals are compiled in Table 3. Even if the interaction indices remain unknown, the deductions drawn by these analyses are correct and informative. This is due to the non separability of the dispersion component effects.

Remark 1 *To illustrate that Y_m and Y_d do not bring enough information to quantitatively estimate all the Sobol indices, we can consider the two following trivial analytical models of two inputs:*

$$\begin{aligned} Y_1(X_1, X_2) &= X_2 X_1 \ , \\ Y_2(X_1, X_2) &= X_2 \, |X_1| \end{aligned} \tag{27}$$

where X_1 and X_2 are independent random variables with zero mean and unit variance. Under these hypothesis, Y_1 and Y_2 have different variance decompositions. Indeed, the Sobol indices for Y_1 are: $S_1 = S_2 = 0$ and $S_{1,2} = 1$, while for Y_2: $S_1 = 0$, $S_2 = \mathbb{E}(|X_1|)^2 \neq 0$ and $S_{12} \neq 1$. If X_1 is considered as the input random variable and X_2 as the seed input variable, it can be easily shown that Y_1 and Y_2 have the same mean and dispersion components: $Y_m = 0$ and $Y_d = {X_1}^2$. This example illustrates that Y_m and Y_d do not bring enough information to quantitatively estimate the different contributions in the total effect of the seed variable. However, the sensitivity analysis of Y_d can yield interval variations for sensitivity indices and also useful information concerning the potential influence of the interactions.

In order to have stronger evidence for the performance differences between joint GLM, joint GAM and joint Gp, we perform 100 repetitions of the joint models fitting process with different Monte Carlo samples (keeping the learning sample size $n = 500$). In an attempt to illustrate that repeating calculations with the same set of controllable variables is less efficient in the joint modeling approach, we also include a comparison with the joint Gp built on a design with replications. To do this, we keep the same analytical form for the Gps on mean and dispersion components as in our joint Gp methodology, but to estimate them we follow the methodology recently proposed by Ankenman et al. (2010). The same set of controllable variables is repeated n_{rep} times, each replication corresponding to a different value of the uncontrollable parameter. A first Gp is estimated on the empirical mean of the set of controllable variable. Then, a second Gp is adjusted on the empirical variance. An estimation of the nugget effect at each point is deduced from the predictor of this second Gp divided by the number of replications. The Gp on the mean is then updated using these nugget effect estimates. Here, we consider different number of replications $n_{rep} = 10$ and $n_{rep} = 20$ and different sizes of the learning sample in the (X_1, X_2) plane (respectively 50 and 25) in order to have the same total number of simulations ($n = 500$). Table 4 shows the results of these computations. As previously, joint GLM results show that this model is inadequate for the Ishigami function. Q_2 of the joint Gp dispersion component is 10% larger than Q_2 of the joint GAM dispersion component. The Sobol indices estimates for the joint GAM and joint Gp are both satisfactory. Concerning the approach with replications, a balance has to be found between the accuracy of the empirical moments (the higher number of replications, the better) and the exploration of the controllable input space (the lower number of replications, the better). We can observe that the mean component and the Sobol indices are correctly estimated with $n_{rep} = 10$ but this number of replications is not sufficient to estimate the variance component. If we increase the number of replications, the accu-

Table 3 Sobol sensitivity indices deduced from Y_d analysis for the Ishigami function: exact and estimated values or variation intervals from joint GLM, joint GAM and joint Gp.

Indices	Theoretical Value	Joint GLM	Joint GAM	Joint Gp
S_{13}	0.244	$]0, 0.385]$	$]0, 0.236]$	$]0, 0.233]$
S_{23}	0	0	0	0
S_{123}	0	0	0	0
S_{T_1}	0.557	$]0.319, 0.704]$	$]0.310, 0.546]$	$]0.312, 0.545]$
S_{T_2}	0.443	0.296	0.454	0.454
S_3	0	$[0, 0.385]$	$[0, 0.236]$	$[0, 0.233]$

racy of the variance component estimate is improved, but the accuracy of the mean component and Sobol indices estimates decreases. In all cases, better results (in terms of accuracy for mean and dispersion components) are obtained by building the joint Gp metamodel on a design without replications which maximizes the exploration of the controllable input space.

In conclusion, the Ishigami example shows that the joint nonparametric models, and specially our proposed joint Gp model, can fit complex heteroscedastic cases for which classical metamodels are inadequate. Moreover, joint models offer a theoretical basis to compute efficiently global sensitivity indices of stochastic models. An analytical model with strong and high order interactions will probably strengthen the superiority of the Gp joint model (because spline high order interaction terms are difficult to include inside a GAM). Besides, in the industrial application of section 5, we only use the joint Gp model.

4.3 Convergence studies

In order to provide some practical guidance for the sampling size issue, we perform a convergence study for the joint Gp modeling and the estimated sensitivity indices. We consider different learning sample size n varying from 50 to 500. The learning points are sampled by simple Monte Carlo and 100 replications are made for each n. The different sets of points are all sampled independently and there is no adaptive approach here. The objective is only to illustrate the convergence speed of the joint Gp predictivity and to give an idea of the number of simulations required in this analytical case with only 2 inputs.

Figure 2 shows some convergence results for the accuracy on mean and dispersion components and the estimation of total sensitivity index S_{T_3} of the input X_3 generated by the seed variable. The predictivity coefficients Q_2 on both Y_m and Y_d are obtained from a test sample composed of 1000 randomly chosen points.

We can notice the rapid convergence of the predictivity coefficient $Q_2(Y_m)$ and the estimation of S_{T_3}. The speed of convergence for S_1, S_2 and S_{12} computed from Y_m are not shown here but are similar to the one of $Q_2(Y_m)$. Accurate modeling of Y_m and estimations of Sobol indices are obtained as soon as $n = 100$. Convergence of the predictivity coefficient $Q_2(Y_d)$ is also observed but is slower than for Y_m. Several hundreds of simulations are required to correctly fit the dispersion component. Thus, in the case of strong and complex heteroscedasticity and when no replication exists, the fitting of the dispersion can be relatively difficult. In practice, convergence of estimated sensitivity indices and their confidence interval (by a bootstrap technique for example) can be plotted and examined visually. It can be a good indicator of the accuracy in fitting Y_m. It can also point out the need of additional simulations. Nevertheless, it does not totally ensure accuracy in fitting the dispersion component.

5 An application case: the PUNQ model

The joint Gp metamodeling methodology is now applied to PUNQ (Production forecasting with UNcertainty Quantification) test case which is an oil reservoir model derived from real field data (Manceau et al. (2001)). The considered reservoir is surrounded by an aquifer in the north and the west, and delimited by a fault in the south and the east. The geological model is composed of five independent layers, three of good quality (layers 1, 3 and 5) and two of poorer quality. A multiphasis fluid flow simulator is used to forecast the oil production during 12 years. 8 scalars variables characteristic of media, rocks, fluids or aquifer activity are considered as uncertain: the coefficient of aquifer strength (AQUI1), horizontal and vertical permeability multipliers in good layers (resp. MPV1 and MPH1), horizontal and vertical permeability multipliers in poor layers (resp. MPV2 and MPH2), coordinate of production well location (P1Y), residual oil saturation after waterflood and

Table 4 For the Ishigami function, from 100 repetitions of the joint GLM, joint GAM, joint Gp following our methodology and joint Gp with replications following Ankenman's method (sample size of $n = 500$ Monte Carlo simulations): exact and mean estimated values of Q_2 and Sobol indices (with standard deviations sd).

	Exact values	Joint GLM		Joint GAM		Joint Gp		Joint Gp with replications Ankenman's method with $n_{rep} = 10$		Joint Gp with replications Ankenman's method with $n_{rep} = 20$	
		mean	sd	mean	sd	mean	sd	mean	sd	mean	sd
$Q_2(Y_m)$	1	0.801	0.003	0.995	0.002	0.981	0.005	0.938	0.035	0.786	0.107
$Q_2(Y_d)$	1	-0.780	0.331	0.725	0.148	0.823	0.100	0.137	0.803	0.516	0.568
S_1	0.314	0.309	0.023	0.295	0.020	0.295	0.029	0.286	0.084	0.265	0.094
S_2	0.442	0.302	0.028	0.448	0.023	0.440	0.036	0.440	0.136	0.419	0.132
S_{T_3}	0.244	0.388	0.022	0.258	0.014	0.222	0.019	0.221	0.075	0.300	0.083

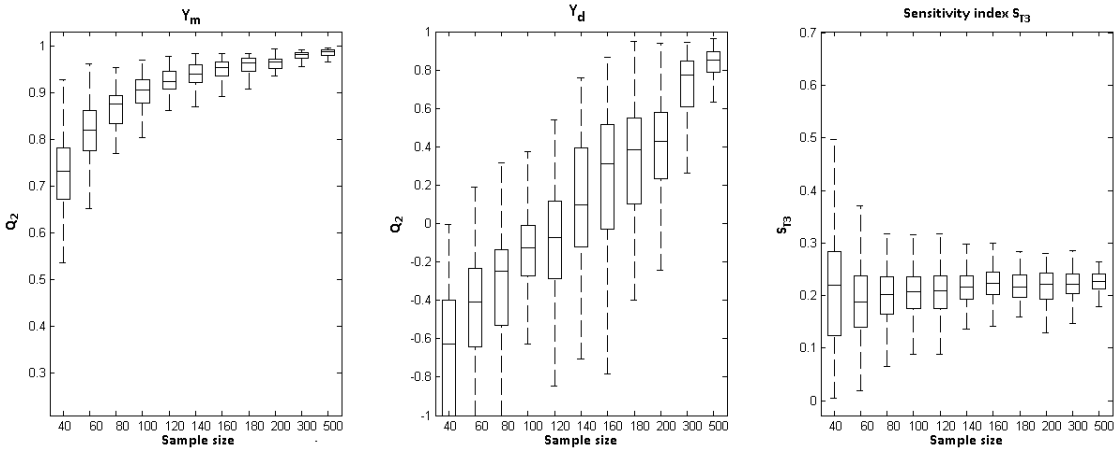

Fig. 2 For the Ishigami function and the joint Gp model, boxplot of $Q_2(Y_m)$ (left), $Q_2(Y_d)$ (center) and estimated S_{T_3} in function of the learning sample size n.

after gas flood (resp. SORW and SORG). Additionally to these 8 uncertain input variables, the porosity map of the first layer is considered as unknown. Geostatistical simulation can be performed to obtain realizations of the porosity map. The resulting spatial random field cannot be summarized by a few scalar values. Therefore, as explained in our introduction, this geostatistical porosity map has to be considered as generated by a seed variable of the computer model.

Among the simulator outputs, we focus here on the produced oil rate after 12 years of exploitation. The objective is to study the impact of both controllable input variables and seed variable on the forecast of produced oil rate. A sensitivity analysis is carried out to identify the most influential inputs among the controllable variables, to quantify the total part of uncertainty related to the porosity map and to point out the potential interaction between the map and the controllable variables. This sensitivity analysis would constitute, for example, a preliminary step before an optimization (robust or not) of tunable parameters like well locations. In this case, the negligible variables identified in the sensitivity analysis would be fixed and tunable parameters would be jointly optimized if there are strong interactions, or separately otherwise. Moreover, this optimization could be done independently from the uncertain porosity map or, if a strong influence of the map with potential interactions is identified, a more refined modeling of interactions between tunable parameters and uncontrollable ones should be used for the optimization. Thus, the results of sensitivity analysis can yield a guidance for a later optimization.

To build the joint Gp model and to make the sensitivity analysis, a learning sample is simulated. The Latin hypercube sampling method is used to obtain a sample of $N = 1000$ random vectors (each one of dimension 8) for the controllable inputs. In addition, for each simulation, an independent realization of the porosity map (denoted X_ϵ) is randomly chosen among a basis of available porosity map realizations. The $N = 1000$ simulations are computed with the fluid flow simulator. Then, a joint Gp model is fitted on simulations, following the proposed methodol-

Table 5 Sobol sensitivity indices for the PUNQ Case estimated by joint Gp modeling.

Input variable	1^{st} order index	"quasi" total effect
AQUI1	0.138	0.154
MPH1	0.101	0.114
MPH2	0.024	0.033
MPV1	0	0
MPV2	0	0
P1Y	0.058	0.069
SORG	0	0
SORW	0.179	0.200

ogy in section 2.3. Sensitivity indices of controllable variables are estimated from the model of the mean component. Table 5 gives their first order indices and "quasi" total effects. "Quasi" refers here to the total effect including only the interactions with the other controllable variables and not with X_ϵ. The difference between first order and "quasi" total indices is a good indicator of possible interactions.

Independently from X_ϵ, the controllable variables have mainly first order effects: there are few interactions between the controllable variables. Their first order effects represent 50% of the output variability. The most influential controllable input is SORW followed by AQUI, MPH1 and P1Y. Only these 4 controllable variables are influential on the mean component, all the others are negligible.

Then, the total effect of X_ϵ can be computed using the estimated dispersion model: $S_{T_\epsilon} = 0.412$. The porosity map has a high total effect of 41% and, consequently, the controllable variables explain alone 59% of the output variability. Sensitivity analysis of the dispersion model shows that the four variables (MPH2, MPV1, MPV2 and SORG) do not interact with X_ϵ. Thus, these four variables, previously identified as non influential on the mean component, are both non influential independently or not from the porosity map. Concerning the others variables, the one which potentially has the higher interaction with X_ϵ is SORW, followed by MPH1, P1Y and AQUI. These results are coherent with the physics. Indeed, the potential quantity of oil in a layer is linked to the porosity (the higher the porosity, the higher the potential quantity of oil). Moreover, referring to the pressure of the aquifer, the lower SORW, the higher the percentage of oil that can be extracted from the layer. As the produced oil rate is linked to the potential quantity in the layer and the potential extracted percentage, it is coherent to detect potential interaction between the porosity map and SORW. As a conclusion, variables MPH2, MPV1, MPV2 and SORG explain 59% of the output variance and potentially interact with the porosity map to explain a part of the 41% remaining.

To illustrate the usefulness of a joint model in this application, we propose to use a graphical tool. It consists in evaluating the proportions Δ of observations that lie within the α-theoretical confidence intervals which are built from the mean and dispersion models and with an additional Gaussian hypothesis. Under this hypothesis, the α-theoretical confidence interval CI_α is given by :

$$CI_\alpha = \left[Y_m(X) - t_\alpha \sqrt{Y_d(X)}; Y_m(X) + t_\alpha \sqrt{Y_d(X)}\right] \tag{28}$$

where t_α is the $\left(1 - \frac{\alpha}{2}\right)$ quantile of the standard normal distribution. Here, we estimate CI_α by replacing $Y_m(X)$ and $Y_d(X)$ with the predictor of $Gp_{m,2}$ and $Gp_{v,2}$, respectively (cf. section 2.3).

We can visualize the proportions Δ (i.e. the observed confidence intervals) against the α-theoretical confidence interval. By definition, if a model is suited for both mean and dispersion modeling, the points should be located around the $y = x$ line. As a consequence, this plot is useful to compare the goodness of fit for the different models. Figure 3 gives the results obtained with the joint and simple Gp modeling. For the simple modeling, only the mean component is fitted and a constant nugget effect is used. It can be seen that the joint Gp is clearly the most accurate model. Indeed, all its points are close to the theoretical $y = x$ line, while the simple Gp tends to give too large confidence intervals. Thus, the heteroscedasticity hypothesis is justified and, in this case, a joint Gp model is clearly more competitive than the simple Gp.

6 Conclusion

In this paper, we have used, in the context of stochastic computer codes, the sensitivity analysis approach based on joint metamodels, first proposed by Iooss

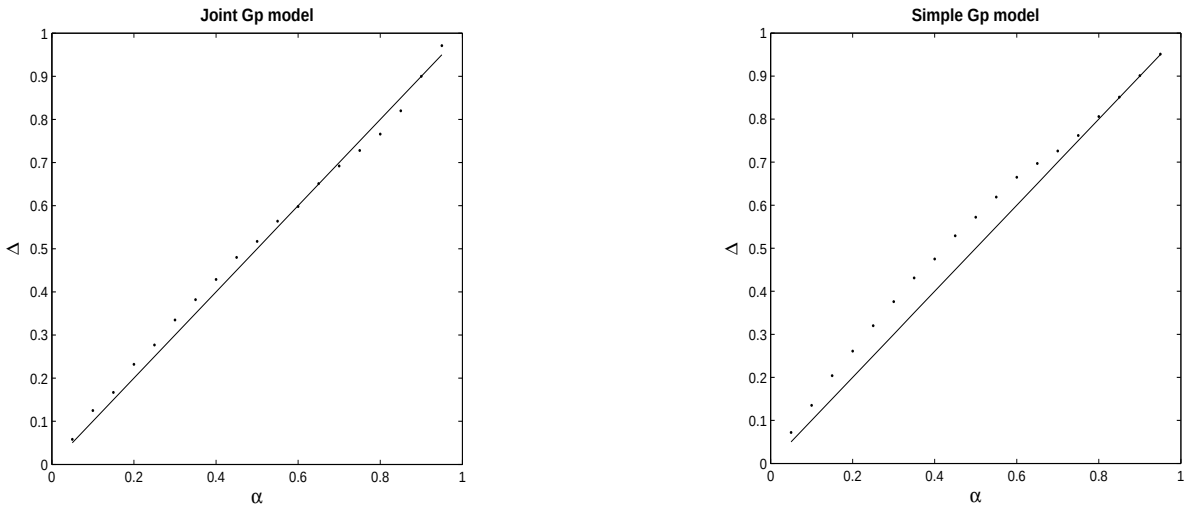

Fig. 3 Proportion Δ of observations that lie within the α theoretical confidence interval in function of the confidence level α for PUNQ data. Joint Gp model, simple Gp model.

& Ribatet (2009). This method can be useful if the following conditions hold:

- if the computer model contains some seed variables which are uncontrollable (the model is no more deterministic but stochastic);
- if a metamodel is needed due to CPU time expensive computer model;
- if some of the seed variables interact with some controllable inputs;
- if some information about the influence of the interactions between the seed variables and the other input variables is of interest.

The solution consists in modeling the mean and the dispersion of the code outputs by two explanatory models. The classical way is to separately build these models. In this paper, the use of the joint modeling is preferred. Zabalza et al. (1998) applied the joint GLM approach to model stochastic computer codes. However, the behavior of some numerical models can be complex and Iooss & Ribatet (2009) introduced the joint GAM which has proven its flexibility in harsh situations. In this paper, we have introduced a new joint Gp model, based on MDE correlation function. This latter model is shown to be more efficient than the former to model dispersion component on a test function. More work is needed in order to study this promising model on stochastic computer codes involving many input variables and strong interactions between model inputs. Moreover, this paper has shown that joint models offer a theoretical basis to compute Sobol sensitivity indices in an efficient way. The analytical formula (for joint GAM) and the sensitivity indices (for joint Gp) of the dispersion component are useful to complete the sensitivity analysis results of the computer code.

The performance of our joint Gp model approach was assessed on an industrial application. Compared to other methods, the modeling of the dispersion component allows to obtain a robust estimation of the total sensitivity index of the seed variable. This yields correct estimations of the first order indices of the input variables. In addition, it reveals the influential interactions between the seed variable and the other input variables. Obtaining quantitative values for these interaction effects is still a challenging problem.

In future work, it would be convenient to test the new approach recently proposed by Gijbels et al. (2010). These authors propose to handle nonparametrically the joint estimation of mean and dispersion functions in extended GLM. The starting point for modeling are GLM in which we no longer admit a linear form for the mean regression function, but allow it to be any smooth function of the covariate(s). The mean regression function and the dispersion function are then estimated using P-splines with difference type of penalty to prevent from overfitting.

Acknowledgements This work was backed by the "Monitoring and Uncertainty" project of IFP Energies Nouvelles and French National Research Agency (ANR) through

COSINUS program (project COSTA BRAVA n^{o}ANR-09-COSI-015). The authors are grateful to Mickaele Le Ravalec for her help with the English.

References

Ankenman, B., Nelson, B., and Staum, J. (2010) Stochastic Kriging for Simulation Metamodeling. *Operations Research*, 58:371–382.

Bayarii, M.J., Berger, J., Cafeo, J., Garcia-Donato, G., Liu, F., Palomo, J., R.J.Parthasarathy, Paulo, R., Sacks, J., and Walsh, D. (2007a). Computer model validation with functional output. *The Annals of Statistics*, 35:1874–1906.

Bayarii, M.J., Berger, J., Paulo, R., Sacks, J., Cafeo, J.A., Cavendish, J., Lin, C., and Tu, J. (2007b). A framework for validation of computer models. *Technometrics*, 49:138–154.

Boukouvalas, A. and Cornford, D. (2009). Learning heteroscedastic Gaussian processes for complex datasets. Technical report, Neural Computing Research Group, Aston University, Birmingham, UK.

Bursztyn, D. and Steinberg, D. (2006). Screening experiments for dispersion effects. In Dean, A. and Lewis, S., editors, *Screening - Methods for experimentation in industry, drug discovery and genetics*. Springer.

Chen, V., Tsui, K.-L., Barton, R., and Meckesheimer, M. (2006). A review on design, modeling and applications of computer experiments. *IIE Transactions*, 38:273–291.

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

Dellino, G., J.P.C. Kleijnen, and C. Meloni (2010). Robust optimization in simulation: Taguchi and Response Surface Methodology. *International Journal of Production Economics*, 125:52–59.

De Rocquigny, E., Devictor, N., and Tarantola, S., editors (2008). *Uncertainty in industrial practice*. Wiley.

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

Forrester, A.I.J., Keane, A.J., and Bressloff, N.W. (2006). Design and analysis of "Noisy" computer experiments. *AIAA Journal*, 44:2331–2339.

Gijbels I., Prosdocimi I., Claeskens G., (2010). Nonparametric estimation of mean and dispersion functions in extended generalized linear models. *Test*, 19:580–608.

Ginsbourger, D., Roustant, O., and Richet, Y. (2008). Kriging with heterogeneous nugget effect for the approximation of noisy simulators with tunable fidelity. In *Proceedings of Joint Meeting of the Statistical Society of Canada and the Société Franaise de Statistique*, Ottawa, Canada.

Gramacy, R.B. and Lee, H.K.H. (2008). Bayesian treed Gaussian process models with an application to computer modeling. *Journal of the American Statistical Association*, 103:1119–1130.

Hastie, T. and Tibshirani, R. (1990). *Generalized additive models*. Chapman and Hall, London.

Helton, J.C., Johnson, J., Salaberry, C., and Storlie, C. (2006). Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering and System Safety*, 91:1175–1209.

Helton, J.C. (2009). Conceptual and computational basis for the quantification of margins and uncertainty. *Sandia National Laboratories*, Report SAND2009-3055.

Homma, T. and Saltelli, A. (1996). Importance measures in global sensitivity analysis of non linear models. *Reliability Engineering and System Safety*, 52:1–17.

Iooss, B., Lhuillier, C., and Jeanneau, H. (2002). Numerical simulation of transit-time ultrasonic flowmeters due to flow profile and fluid turbulence. *Ultrasonics*, 40:1009–1015.

Iooss, B. and Ribatet, M. (2009). Global sensitivity analysis of computer models with functional inputs. *Reliability Engineering and System Safety*, 94:1194–1204.

Kelton, W.D., Sadowski, R.P., Sturrock, D.T. (2007). Simulation with Arena; fourth edition. *McGraw-Hill, Boston*.

Kennedy, M. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society*, 63(3):425–464.

Kersting, K., Plagemann, C., Pfaff, P. and Burgard, W. (2007). Most likely heteroscedastic Gaussian process regression. *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, Oregon, USA.

Kleijnen, J.P.C (1997). Sensitivity analysis and related analyses: a review of some statistical techniques. *Journal of Statistical Computation and Simulation*, 57:111–142.

Kleijnen, J.P.C and van Beers, W. (2005). Robustness of kriging when interpolating in random simulation with heterogeneous variances: some experiments. *European Journal of Operational Research*,

165:826–834.

Kleijnen, J.P.C (2008). Design and analysis of simulation experiments. *Springer*.

Lee, Y. and Nelder, J. (2003). Robust design via generalized linear models. *Journal of Quality Technology*, 35(1):2–12.

Manceau, E., Mezghani, Zabalza-Mezghani, I., and Roggero, F. (2001). Combination of experimental design and joint modeling methods for quantifying the risk associated with deterministic and stochastic uncertainties - An integrated test study. *2001 SPE Annual Technical Conference and Exhibition, New Orleans, 30 September-3 October*, paper SPE 71620.

Marrel, A., Iooss, B., Van Dorpe, F., and Volkova, E. (2008). An efficient methodology for modeling complex computer codes with Gaussian processes. *Computational Statistics and Data Analysis*, 52:4731–4744.

Martin, J. and Simpson, T. (2005). Use of kriging models to approximate deterministic computer models. *AIAA Journal*, 43:853–863.

McCullagh, P. and Nelder, J. (1989). *Generalized linear models*. Chapman & Hall.

Myers, R.H., Montgomery, D.C., and Anderson-Cook, C.M. (2009). Response surface methodology: process and product optimization using designed experiments; third edition. *Wiley, New-York*

Nelder, J. and Pregibon, D. (1987). An extended quasi-likelihood function. *Biometrika*, 74:221–232.

Nelder, J. and Wedderburn, R. (1972). Generalized linear models. *Journal of the Royal Statistical Society A*, 135:370–384.

Phadke, M. (1989). *Quality engineering using robust design*. Prentice-Hall, New-York, NY.

Picheny, V., Ginsbourger, D., Richet, Y., and Caplin, G. (2011). Optimization of noisy computer experiments with tunable precision. *Technometrics*, accepted, in revision.

Pope, B. (1994). Lagrangian pdf methods for turbulent reactive flows. *Annual Review of Fluid Mechanics*, 26:23–63.

Reich, B.J., Kalendra, E., Storlie, C.B., Bondell, H.D., and Fuentes, M. (2009). Variable selection for Bayesian density estimation: Application to human exposure simulation. *Environmental and Ecological Statistics*, submitted.

Robinson, T. J., Birch, J. and Alden Starnes, B. (2010). A semi-parametric approach to dual modeling when no replication exists. *Journal of Statistical Planning and Inference*, 140:2860–2869.

Ruffo, P., Bazzana, L., Consonni, A., Corradi, A., Saltelli, A., and Tarantola, S. (2006). Hydrocarbon exploration risk evaluation through uncertainty and sensitivity analysis techniques. *Reliability Engineering and System Safety*, 91:1155–1162.

Sacks, J., Welch, W., Mitchell, T., and Wynn, H. (1989). Design and analysis of computer experiments. *Statistical Science*, 4:409–435.

Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., Tarantola, S. (2010). Variance based sensitivity analysis of model output. Design and estimator for total sensitivity index. *Computer Physics Communication*, 181:259–270.

Saltelli, A., Chan, K., and Scott, E., editors (2000). *Sensitivity analysis*. Wiley Series in Probability and Statistics. Wiley.

Siebers, P.O., Macal, C.M , Garnett, J., Buxton, D., and Pidd, M. (2010). Discrete-event simulation is dead, long live agent-based simulation! *Journal of Simulation*, 4:204–210.

Smyth, G. (1989). Generalized linear models with varying dispersion. *Journal of the Royal Statistical Society B*, 51:47–60.

Sobol, I. (1993). Sensitivity estimates for non linear mathematical models. *Mathematical Modelling and Computational Experiments*, 1:407–414.

Storlie, C.B., Swiler, L.P., Helton, J.C., and Salaberry, C.J. (2009). Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models (2009). *Reliability Engineering and System Safety*, 94:1735–1763.

Vining, G. and Myers, R. (1990). Combining Taguchi and response-surface philosophies - a dual response approach. *Journal of Quality Technology*, 22:38–45.

Volkova, E., Iooss, B., and Van Dorpe, F. (2008). 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.

Wood, S. and Augustin, N. (2002). GAMs with integrated model selection using penalized regression splines and applications to environmental modelling. *Ecological modelling*, 157:157–177.

Yeşilyurt, S., Ghaddar, C.K., Cruz M.E., and Patera, A.T. (1996). Bayesian-validated surrogates for noisy computer simulations; application to random media. *SIAM J. Sci. Comput.*, 17:973–992.

Zabalza, I., Dejean, J., and Collombier, D. (1998). Prediction and density estimation of a horizontal well productivity index using generalized linear models. In *Proceedings of ECMOR VI, Peebles, Scotland, September 8-11, 1998*.

Zabalza, I., Manceau, E., and Roggero, F. (2001). A new approach for quantifying the impact of geostatistical uncertainty on production forecasts: The joint modeling method. In *Proceedings of IAMG Conference, Cancun, Mexico, September 6-12, 2001*.

Zabalza-Mezghani, I., Manceau, E., Feraille, M., and Jourdan, A. (2004). Uncertainty management: From geological scenarios to production scheme optimization. *Journal of Petroleum Science and Engineering*, 44:11–25.