13th International Conference on Applications of Statistics and Probability in Civil Engineering, ICASP13
Seoul, South Korea, May 26-30, 2019

Surrogating the response PDF of stochastic simulators using generalized lambda distributions

Xujia Zhu
Ph.D. candidate, Dept. of Civil Engineering, ETH Zürich, Zürich, Switzerland

Bruno Sudret
Professor, Dept. of Civil Engineering, ETH Zürich, Zürich, Switzerland

ABSTRACT: Computer simulation is used in all fields of applied science and engineering to represent complex systems and be able to make predictions about their behaviour. Kriging and polynomial chaos (PC) expansions are nowadays well-established techniques to surrogate complex models when a large number of runs is required, which is the case in the context of optimization or uncertainty quantification. Classically, computer models are deterministic, in the sense that they produce the exact same output quantities of interest when run twice with the same parameters. In contrast, in this paper, we are interested in *stochastic simulators*, for which there are extra internal sources of randomness in the computer code, so that two runs produce different results. Of interest is the construction of a surrogate that predicts the response probability density function (PDF) for any input parameter set. We propose a two-step approach based on a local inference of the response PDF in each point of an experimental design using generalized lambda distributions, the distribution parameters of which being represented in a second step by PC expansions. Two versions of the algorithm are proposed and compared on two analytical examples, which allow to assess their respective accuracy.

1. INTRODUCTION

1.1. Stochastic simulators

Computational codes (a.k.a. simulators) are nowadays widely used in the context of design optimization, uncertainty quantification and reliability analysis. A simulator is called *deterministic* if repeated runs with the same input parameters produce the same corresponding output quantity of interest (QoI), *e.g.* stresses computed from a finite element model. In contrast, a *stochastic simulator* provides different results when run twice with the same input values. In other words, for a given vector of input parameters, the QoI of a stochastic simulator is a random variable. The reason for the intrinsic randomness is that some source of randomness inside the model, which can be represented by latent variables, is not taken explicitly into account as input parameters. Therefore, if all the relevant variables that uniquely determine the output cannot be fully specified (*e.g.* stochastic loads on wind turbines when only knowing some characteristic value of the wind climate), the model output will be uncertain.

Due to the random nature of the stochastic simulator, repeated runs with the same input parameters (so-called *replications*) are necessary to characterize the distribution of the QoI. However, numerical models can be time-consuming: a single model evaluation may require hours to even days of simulation (*e.g.* complex fluid dynamic codes). Therefore, it is advantageous to construct surrogate models (a.k.a. emulators) based on a set of model evaluations, called *experimental design* (ED). Conventional surrogate modelling methods such as Gaussian processes (a.k.a Kriging) (Rasmussen and Williams, 2006), polynomial chaos expansions (Ghanem and Spanos, 2003) that have been developed for deterministic simulators cannot be directly applied to stochastic codes because of their random nature.

Two types of approaches can be applied to es-

timate the conditional probability density distribution (PDF) of the QoI given the input vector. The first one is referred to as *statistical approach* in the sequel. If the conditional PDF belongs to the exponential family, generalized linear models (GLM) can be efficiently applied (McCullagh and Nelder, 1989; Hastie and Tibshirani, 1990). In the case where the probability distribution is arbitrary and no prior knowledge of its shape is available, nonparametric estimators may be considered, notably kernel density estimators (Fan and Gijbels, 1996; Hall et al., 2004) and projection estimators (Efromovich, 2010). However, nonparametric estimators suffer from the so-called *curse of dimensionality* (Tsybakov, 2009), meaning that the estimation accuracy decreases fast with increasing input dimension.

The second approach is the *replication-based method*. The main idea is to use replications to characterize the output distribution through a few hyper-parameters and then treat those as outputs of a deterministic simulator. So far, nonparametric estimators have been used to estimate the distribution based on replications (Moutoussamy et al., 2015), and thus many repeated runs are necessary, *e.g.* 10^4 replications per ED point are used in Browne et al. (2016). As we can observe, the existing methods either assume a specific form of the distribution or requires a large number of model evaluations. In order to have a reasonable number of model runs without losing flexibility, we propose to approximate the response PDF of the stochastic simulator by the *generalized lambda distribution* (GLD).

In this paper, the input parameters of the computational model are modelled by random variables gathered into a random vector $\boldsymbol{X} = (X_1, X_2, \ldots, X_M)^T$. The output random variable is denoted by Y. The lower case $\boldsymbol{x} = (x_1, x_2, \ldots, x_M)^T$ refers to one realization of the input variables and y is the associated QoI upon one model evaluation with $\boldsymbol{x}$. The joint distribution of $\boldsymbol{X}$ is denoted by $f_{\boldsymbol{X}}(\boldsymbol{x})$ and its support is denoted by $\mathscr{D}_{\boldsymbol{x}}$. We assume that input variables are mutually independent. The stochastic simulator is evaluated over the experimental design $\mathscr{X} = \left\{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(N)}\right\}$. The upper index (i) denotes the i^{th} point of ED. If the stochastic simulator is repeatedly run R times for each point of ED $\boldsymbol{x}^{(i)}$, the associated output is denoted by $\left\{y^{(i,1)}, y^{(i,2)}, \ldots, y^{(i,R)}\right\}$, where the upper index (i,r) refers to the result of the r^{th} replication for the i^{th} point of the experimental design. The goal of the paper is to develop a surrogate model to predict the response PDF of stochastic simulators based on generalized lambda distributions and polynomial chaos expansions.

The paper is organized as follows. The generalized lambda distribution and polynomial chaos expansions are introduced respectively in Section 2 and Section 3. In Section 4, we present the developed algorithms, and the performance of the proposed methods are illustrated through examples in Section 5.

2. GENERALIZED LAMBDA DISTRIBUTION

The generalized lambda distribution (GLD) is a highly flexible four-parameter continuous probability distribution. It is able to approximate most of the well-known parametric probability distributions (Karian and Dudewicz, 2000), *e.g.* normal, uniform, Student's t, exponential, lognormal, Weibull distribution *etc.* as illustrated in Figure 1.

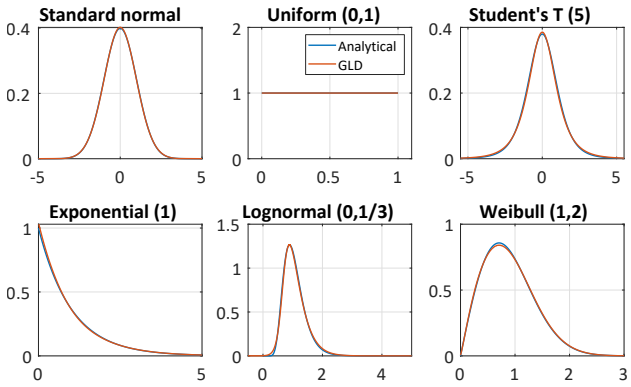


Figure 1: Some parametric PDFs approximated using the generalized lambda distribution.

Instead of a direct parametrization of the PDF, the GLD parametrizes the quantile function Q, which is the inverse of the cumulative distribution function $Q = F^{-1}$. Therefore, the range of definition of Q is $[0,1]$ and Q is non-decreasing. In this paper, we consider the FMKL family (Freimer

et al., 1988) defined by:

$$Q_{\mathrm{FMKL}}(u) = \lambda_1 + \frac{1}{\lambda_2}\left(\frac{u^{\lambda_3}-1}{\lambda_3} - \frac{(1-u)^{\lambda_4}-1}{\lambda_4}\right). \tag{1}$$

Note that parametrizing the quantile function is equivalent to modelling the inverse probability integral transform, meaning that the random variable Y with Q as quantile function and the random variable $Q(U)$ with $U \sim \mathscr{U}(0,1)$ follow the same distribution. Therefore, the PDF of GLD can be calculated through a change of variables. In general, the PDF cannot be expressed analytically as implied in the following equations:

$$f_Y(y) = \frac{1}{Q'(Q^{-1}(y))} = \frac{\lambda_2}{u^{\lambda_3-1}+(1-u)^{\lambda_4-1}} \tag{2}$$

$$u = Q^{-1}(y) \tag{3}$$

Many estimation methods have been proposed to fit the GLD to data (Chalabi et al., 2011). Karian and Dudewicz (2010); Corlu and Meterelliyoz (2016) compared different methods through exhaustive Monte Carlo simulation with various tests. Unfortunately, none of the existing estimators are shown to always outperform the others. The performance depends on the shape of the true distribution, the sample size and the goodness-of-fit criterion used for the comparison. In this paper, we will apply the maximum likelihood estimation (Su, 2007) and the method of moments which relies on matching the first four standardized moments (Karian and Dudewicz, 2000). It is worth remarking here that, because the PDF of GLD is not explicitly given (see Eq. (2)), the calculation of the likelihood requires solving the nonlinear equation (3), which can be time-consuming for large data sets.

3. Polynomial chaos expansions

As a popular surrogate modelling method, polynomial chaos expansions (PCE) represent a deterministic function by a series of polynomials in the input variables that are orthogonal with respect to the probability distribution of the input. Consider a scalar-valued deterministic computational model $\mathscr{M}(\boldsymbol{X})$. If $\mathscr{M}(\boldsymbol{X})$ has finite variance, it can be decomposed as

$$\mathscr{M}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha}\in\mathbb{N}^M} a_{\boldsymbol{\alpha}}\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \tag{4}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_M) \in \mathbb{R}^M$ denotes the multi-index defining the function $\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ and $a_{\boldsymbol{\alpha}}$ is the associated coefficient. $\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ is constructed as the tensor product of univariate polynomials:

$$\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) = \prod_{j=1}^{M} \phi_{\alpha_j}^{(j)}(X_j) \tag{5}$$

where $\phi_{\alpha_j}^{(j)}$ is the polynomial of degree α_j associated with the j^{th} component of $\boldsymbol{X}$. By construction, polynomials $\left\{\phi^{(j)}, j \in \mathbb{N}\right\}$ are orthogonal with respect to the distribution of X_j:

$$\mathbb{E}_{X_j}\left[\phi_m^{(j)}(X_j)\cdot\phi_n^{(j)}(X_j)\right] = \delta_{m,n} \tag{6}$$

with δ being the Kronecker delta. Xiu and Karniadakis (2002) listed polynomials that fulfil Eq. (6) for some distributions, *e.g.* normal, uniform, exponential. For arbitrary distributions, orthogonal polynomials can be obtained through the so-called *Gram–Schmidt procedure*.

For non-polynomial models $\mathscr{M}$, the decomposition Eq. (4) results in an infinite number of terms. Therefore, truncation schemes must be applied, which consists in approximating $\mathscr{M}(\boldsymbol{x})$ by a finite series defined by a finite subset $\mathscr{A} \subset \mathbb{N}^M$. It has been observed that PCE suffer from the curse of dimensionality (Sudret, 2015), due to the exponential increase of the basis size with increasing input dimension or polynomial degree. To overcome this problem, sparse PC models have been proposed, which only contain a small number of basis functions. One method for constructing a sparse model is *least angle regression* (LAR) (Efron et al., 2004). This method selects only the most important basis functions among a candidate set and has shown its excellent performance (Blatman and Sudret, 2011).

4. Two-step approach and joint modelling

As mentioned in Section 2, we assume that the response PDF of the stochastic simulator for a given

input vector $\boldsymbol{x}$ can be approximated by the GLD with distribution parameters $\boldsymbol{\lambda}$ that are functions of $\boldsymbol{x}$:

$$Y(\boldsymbol{x}) \sim GLD\left(\lambda_1(\boldsymbol{x}), \lambda_2(\boldsymbol{x}), \lambda_3(\boldsymbol{x}), \lambda_4(\boldsymbol{x})\right) \tag{7}$$

Under appropriate assumptions discussed in Section 3, $\boldsymbol{\lambda}(\boldsymbol{X})$ admit PC representations. In the Freimer-Mudholkar-Kollia-Lin (FMKL) GLD model, $\lambda_2(\boldsymbol{x})$ should always be positive. As a result, PCE are built on $\log\left(\lambda_2(\boldsymbol{x})\right)$, where log denotes the natural logarithmic function. This leads to:

$$\lambda_k(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha}\in\mathbb{N}^M} a_{k,\boldsymbol{\alpha}}\,\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \;\; k = 1,3,4 \tag{8}$$

$$\lambda_2(\boldsymbol{X}) = \exp\left(\sum_{\boldsymbol{\alpha}\in\mathbb{N}^M} a_{2,\boldsymbol{\alpha}}\,\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right) \tag{9}$$

Similar to Moutoussamy et al. (2015); Browne et al. (2016), one straightforward way to build a surrogate model is the two-step approach presented in Algorithm 1.

Algorithm 1 Two-step algorithm

1: Estimate the distribution parameters $\hat{\boldsymbol{\lambda}}^{(i)}$ based on replications $\left\{y^{(i,1)}, y^{(i,2)}, \ldots, y^{(i,R)}\right\}$ for each design point $\boldsymbol{x}^{(i)}$
2: Build a sparse polynomial chaos model for each component of the distribution parameters $\tilde{\boldsymbol{\lambda}}(\boldsymbol{x})$ by applying least angle regression to the data $\left(\boldsymbol{x}^{(i)}, \hat{\boldsymbol{\lambda}}^{(i)}\right)$

The first step of Algorithm 1 provides an estimator $\hat{\boldsymbol{\lambda}}^{(i)}$ of the distribution parameters $\boldsymbol{\lambda}\left(\boldsymbol{x}^{(i)}\right)$. In this paper, we tested both the method of moments and the maximum likelihood estimation (the comparison is presented in Section 5). The second step considers the obtained data to build 4 surrogate models of the distribution parameters according to Eqs. (8)-(9).

In general, the estimator $\hat{\boldsymbol{\lambda}}^{(i)}$ is based on samples of finite size due to the computational limit. The generalized lambda distribution is so flexible that few samples will not guarantee accurate estimations (Corlu and Meterelliyoz, 2016) and none of the existing estimators have been proved to be unbiased. Consequently, the two-step algorithm qualitatively requires a large number of replications R to achieve a good estimation (quantitative results will be shown in Section 5).

This drawback of the approach is due to the use of two separate steps, *i.e.* two objective functions are optimized sequentially while the data are only used in the first step. Moreover, the distribution parameters are surrogated independently in the second step of Algorithm 1, and thus the interactions among components of $\boldsymbol{\lambda}$ are ignored. From a statistical perspective, the true parameters of this GLD model are the PCE coefficients $\boldsymbol{a}$. The goal is to define an objective function for optimiting the PCE coefficients $\boldsymbol{a}$ so that all the available data may be used at once. Similar to the generalized linear model (McCullagh and Nelder, 1989) where the maximum likelihood estimator is used to estimate the model parameters, we propose now a joint modelling framework in Algorithm 2.

Algorithm 2 Joint modelling

1: Apply Algorithm 1 to get $\tilde{\boldsymbol{\lambda}}(\boldsymbol{x})$ with $\tilde{\boldsymbol{a}}$ as coefficients
2: $\hat{\boldsymbol{a}} \leftarrow \arg\min_{\boldsymbol{a}} l(\boldsymbol{a})$ where

$$l(\boldsymbol{a}) := \sum_{i,r} -\log f_{Y|X}\left(y^{(i,r)} \middle| \boldsymbol{\lambda}\left(\boldsymbol{x}^{(i)}\right)\right) \tag{10}$$

$$\lambda_k(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha}\in\mathscr{A}_k} a_{k,\boldsymbol{\alpha}}\,\psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) \;\; k = 1,3,4 \tag{11}$$

$$\lambda_2(\boldsymbol{X}) = \exp\left(\sum_{\boldsymbol{\alpha}\in\mathscr{A}_2} a_{2,\boldsymbol{\alpha}}\,\psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right) \tag{12}$$

with $\mathscr{A}_k$ denoting the sparse space selected by the least angle regression for $\lambda_k(\boldsymbol{x})$ in Algorithm 1

The first step of Algorithm 2 is to apply Algorithm 1, which provides a first estimation $\tilde{\boldsymbol{a}}$ of the PCE coefficients. Because least angle regression is used in Algorithm 1, the first step of Algorithm 2 also selects the basis functions for each component of $\boldsymbol{\lambda}(\boldsymbol{x})$. The second step in Algorithm 2 estimates the coefficients associated with the selected basis functions through the maximum likelihood estima-

tion using all the available data.

The details of calculating the likelihood function with given PCE coefficients $\boldsymbol{a}$ for the data point $\left(\boldsymbol{x}^{(i)}, y^{(i,r)}\right)$ are illustrated in Figure 2 and described here. The preliminary step (referred as step 0 in Figure 2) evaluates the basis functions $\psi_{\boldsymbol{\alpha}}$ on $\boldsymbol{x}^{(i)}$. Step 1 calculates the associated distribution parameters $\boldsymbol{\lambda}^{(i)} = \boldsymbol{\lambda}\left(\boldsymbol{x}^{(i)}\right)$ according to Eqs. (11)-(12) in which the model parameters $\boldsymbol{a}$ are used. The different basis functions selected for each component of $\boldsymbol{\lambda}(\boldsymbol{x})$ make the two layers (layer 1 and layer 2 in Figure 2) involved in this step not fully connected. Step 2 solves the nonlinear equation $u_{i,r} = Q^{-1}\left(y^{(i,r)}\right)$. According to Eq. (1), the nonlinear equation is explicitly written as

$$y^{(i,r)} = \lambda_1^{(i)} + \frac{1}{\lambda_2^{(i)}}\left(\frac{u_{i,r}^{\lambda_3^{(i)}} - 1}{\lambda_3^{(i)}} - \frac{(1-u_{i,r})^{\lambda_4^{(i)}} - 1}{\lambda_4^{(i)}}\right). \tag{13}$$

Step 3 finally computes the negative log-likelihood via Eq. (2). More precisely, we have

$$f_{Y|X}\left(y^{(i,r)}\middle|\boldsymbol{\lambda}^{(i)}\right) = \frac{\lambda_2^{(i)}}{u_{i,r}^{\lambda_3^{(i)}-1} + (1-u_{i,r})^{\lambda_4^{(i)}-1}}. \tag{14}$$

When minimizing the negative log-likelihood as a function of $\boldsymbol{a}$, analytical expressions of the derivatives of the negative log-likelihood with respect to $\boldsymbol{a}$ can be derived through implicit differentiation of Eqs. (1)-(3). The derivation is tedious, and thus details are omitted here. Since the derivatives are obtained without additional computational burden and the problem can be vectorized, derivative-based optimization methods can be efficiently applied. Since this is a nonlinear optimization problem, the initial starting point that may be required by some optimization algorithms is chosen to be the result provided by the first step, *i.e.* $\tilde{\boldsymbol{a}}$.

5. Examples

In this section, we investigate the performance of Algorithms 1 and 2 using two examples. In the first step of Algorithm 1, we apply both the method of moments and maximum likelihood estimation

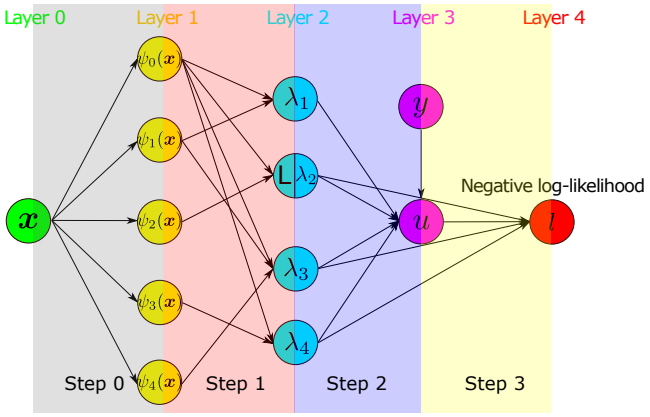


Figure 2: Flow chart of the joint models

to get $\hat{\boldsymbol{\lambda}}^{(i)}$. The associated surrogate models built from the two-step algorithm are respectively denoted by *MM* and *MLE*. Similarly, Algorithm 2 provides another two models denoted by *joint_MM* and *joint_MLE*. Since the analytical results are known for these selected examples, *i.e.* the exact expression of the PDF of $Y \mid \boldsymbol{X}$ for any $\boldsymbol{X} = \boldsymbol{x}$, the estimation error is computed as

$$\varepsilon = \mathbb{E}_{\boldsymbol{X}}\left[d_{\mathrm{HD}}\left(f_{Y|\boldsymbol{X}}(y \mid \boldsymbol{X}), \hat{f}_{Y|\boldsymbol{X}}(y \mid \boldsymbol{X})\right)\right] \tag{15}$$

$$d_{\mathrm{HD}}\left(f(y), \hat{f}(y)\right) = \frac{1}{2}\int\left(\sqrt{f(y)} - \sqrt{\hat{f}(y)}\right)^2 dy \tag{16}$$

where d_{HD} is the Hellinger distance between the true PDF $f_{Y|\boldsymbol{X}}$ and the predicted PDF $\hat{f}_{Y|\boldsymbol{X}}$. In order to calculate the expectation in Eq. (15), Monte Carlo simulation is used with 1,000 quasi-random samples generated by the Sobol' sequence in the input space. The Sobol' sequence sampler is also used to draw the experimental design (ED). To study the performance of the proposed methods, data are generated for various combinations of the size N of the experimental design and the amount of replications R per ED point. Each scenario is run independently 100 times with different ED to account for statistical uncertainty.

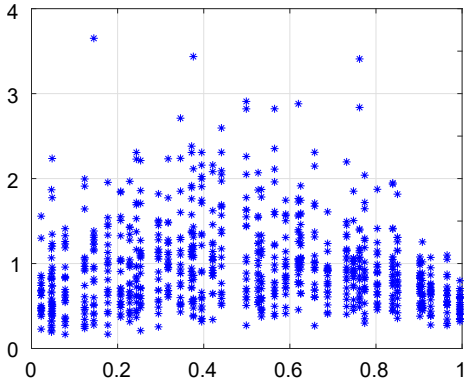
5.1. *A one-dimensional example (Example 1)*

The first example is defined as follows:

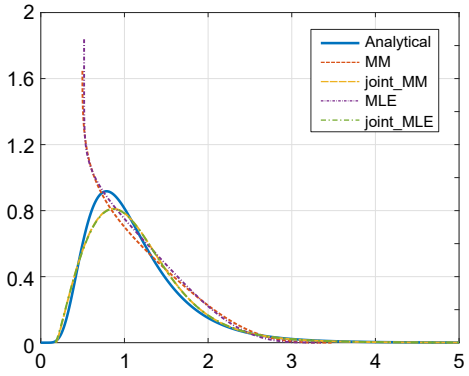
$$Y = \sin\left(\frac{2\pi}{3}X + \frac{\pi}{6}\right)\cdot(Z_1 \cdot Z_2)^{\cos(X)} \tag{17}$$

where $Z_1 \sim \mathscr{LN}(0, 0.25)$ and $Z_2 \sim \mathscr{LN}(0, 0.5)$ are lognormal latent variables, while $X \sim \mathscr{U}(0,1)$ is the single input. The response PDF given $X = x$ is a lognormal distribution $\mathscr{LN}(\mu(x), \sigma(x))$ with distribution parameters $\mu(x) = \log\left(\sin\left(\frac{2\pi}{3}x + \frac{\pi}{6}\right)\right)$ and $\sigma(x) = \cos(x)\sqrt{0.375}$.

One realization of the case with 40 ED points and 20 replications and the predicted PDF of the four surrogate models for $x = 0.5$ are shown in Figure 3. It can be observed that in this case the two models *MM* and *MLE* built using Algorithm 1 cannot capture the shape of the true distribution. In contrast, the two joint models *joint_MM* and *joint_MLE* give quite similar PDFs that have the right shape.



(a) Generated data



(b) PDF prediction for $x = 0.5$

Figure 3: Example 1, 40 ED points and 20 replications

Figures 4 to 6 show quantitative comparisons of the convergence behaviour of the four models. We observe that the two joint models generally outperform the models built from the two-step approach, especially when only a few replications are available.

In the case with only 20 replications (Figure 4), the convergence behaviour of *MM* and *MLE* shows a weak dependence on the ED size. This is because in the first step of Algorithm 1, estimators $\hat{\boldsymbol{\lambda}}^{(i)}$ from both the method of moments and the maximum likelihood estimation might be biased. Then regression used in the second step is not able to distinguish the bias from the true value. Moreover, a few replications lead to high variance of the estimators, which together with the bias explain the nonconvergent behaviour of *MM* and *MLE*. When increasing the number of replications, the bias becomes less significant with respect to the true value and the variance decreases. Therefore, in Figure 6 the error decreases with increasing size of ED for *MM* and *MLE*.

In contrast, *joint_MLE* shows a clear convergence behaviour even with small amount of replications. This is because all the available data are used to estimate the model parameters, which reduce the bias and variance. *Joint_MM* appears to provide less accurate PDF estimation than *joint_MLE* but it still performs better than both *MM* and *MLE*. In this example, using maximum likelihood estimation to obtain $\hat{\boldsymbol{\lambda}}^{(i)}$ provides more accurate results in both Algorithms 1 and 2.

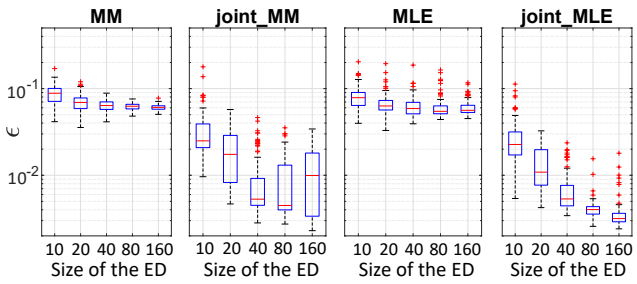


Figure 4: Example 1, 20 replications

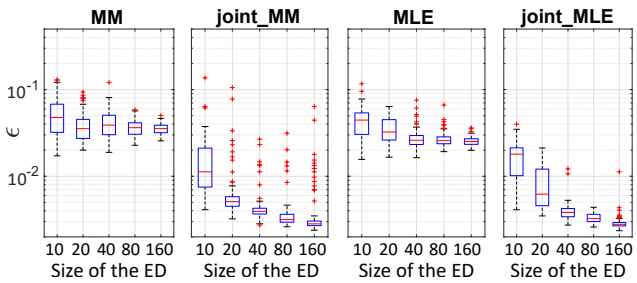


Figure 5: Example 1, 40 replications

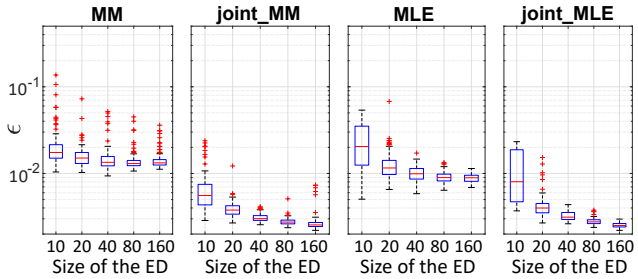


Figure 6: Example 1, 80 replications

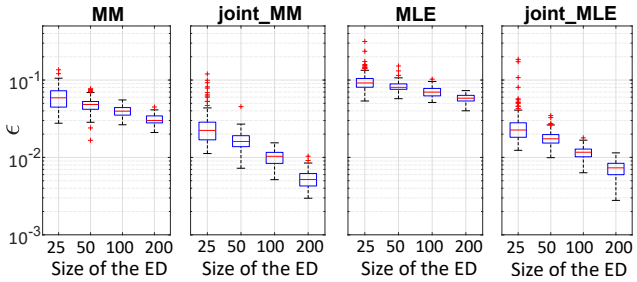


Figure 7: Example 2, 25 replications

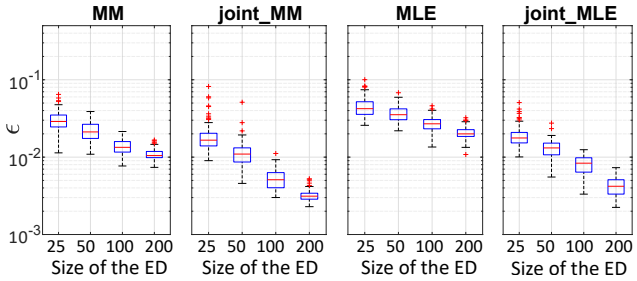


Figure 8: Example 2, 50 replications

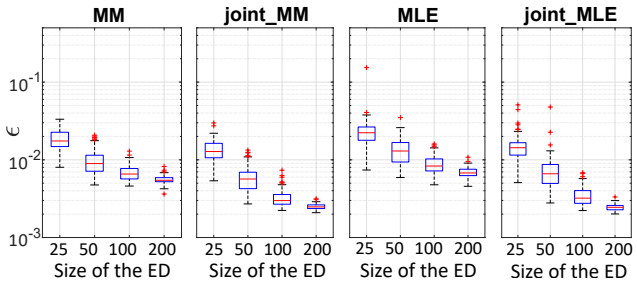


Figure 9: Example 2, 100 replications

5.2. *A five-dimensional example (Example 2)*

The second example is defined as follows:

$$Y(\boldsymbol{X},Z) = \mu(\boldsymbol{X}) + \sigma(\boldsymbol{X}) \cdot Z \tag{18}$$

$$\mu(\boldsymbol{x}) = 3 - \sum_{j=1}^{5} j x_j + \frac{1}{5}\sum_{j=1}^{5} j x_j^3 + \frac{1}{15}\log\left(1 + \sum_{j=1}^{5} j(x_j^2 + x_j^4)\right) + x_1 \cdot x_2 - x_5 \cdot x_3 + x_2 \cdot x_4 \tag{19}$$

$$\sigma(\boldsymbol{x}) = \exp\left(\frac{1}{4}\sum_{j=1}^{5} x_j\right) \tag{20}$$

where $Z \sim \mathcal{N}(0,1)$ is the latent variable that represents the source of randomness and $X_j \sim \mathcal{U}(0,1)$ are input parameters. Given $\boldsymbol{X} = \boldsymbol{x}$, $Y(\boldsymbol{x})$ is a Gaussian random variable with mean $\mu(\boldsymbol{x})$ and standard deviation $\sigma(\boldsymbol{x})$.

Similar to the previous example, it is observed from Figures 7 to 9 that the two joint models show better performance than those built by Algorithm 1. Both *MM* and *MLE* converge rather slowly with respect to the ED size in the case of small amount of replications (Figure 7). In contrast, the two joint models are less sensitive to the number of replications. Unlike the first example, the parametric estimation methods employed to get $\hat{\boldsymbol{\lambda}}^{(i)}$ do not affect the behaviour of Algorithms 1 and 2.

In this section, only the error measure based on the Hellinger distance is reported for convergence studies. Nevertheless, quantitative comparisons of the other quantities such as mean and 95% quantile show similar convergence behaviour.

6. CONCLUSIONS

The aim of this paper was to build efficient and accurate surrogate models for stochastic simulators with a relatively small number of model evaluations. Generalized lambda distributions are used to flexibly approximate the output PDF, and the distribution parameters are represented by polynomial chaos expansions. To construct surrogate models in a non-intrusive manner, we first proposed a two-step algorithm which consists of solving two consecutive problems and which allows the use of PCE in the standard regression way.

Due to the nature of Algorithm 1, it is observed that this approach is sensitive to the number of replications. In order to build accurate surrogate models even when not many replications are available, we proposed in a second part the joint modelling framework Algorithm 2. The latter has

shown to have better performance according to the analytical examples described in Section 5.

More tests should be be considered in future research to study the applicability of the methods to data gathered from real problems, *e.g.* wind turbine simulation. Besides, in the framework of joint modelling, the main role played by replications is to select the basis functions of $\boldsymbol{\lambda}(\boldsymbol{x})$ and to find initial starting points for optimization. Therefore, replications are not indispensable in the joint modelling framework if the basis functions of each distribution parameter are known or preselected. In the future research, we plan to improve the joint modelling framework by using advanced statistical methods to avoid the need of replications, and thus to reduce the total number of the model evaluations.

7. REFERENCES

Blatman, G. and Sudret, B. (2011). "Adaptive sparse polynomial chaos expansion based on Least Angle Regression." *J. Comput. Phys.*, 230, 2345–2367.

Browne, T., Iooss, B., Le Gratiet, L., Lonchampt, J., and Rémy, E. (2016). "Stochastic simulators based optimization by Gaussian process metamodels – application to maintenance investments planning issues." *Quality Reliab. Eng. Int.*, 32(6), 2067–2080.

Chalabi, Y., Scott, D. J., and Würtz, D. (2011). "The generalized lambda distribution as an alternative to model financial returns." *Eidgenössische Technische Hochschule and University of Auckland, Zurich and Auckland. Retrieved from Rmetrics*.

Corlu, C. G. and Meterelliyoz, M. (2016). "Estimating the parameters of the generalized lambda distribution: Which method performs best?." *Comm. Stat. Simul. Comput.*, 45(7), 2276–2296.

Efromovich, S. (2010). "Dimension reduction and adaptation in conditional density estimation." *J. Amer. Stat. Assoc.*, 105, 761–774.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). "Least angle regression." *Annals of Statistics*, 32, 407–499.

Fan, J. and Gijbels, I. (1996). *Local Polynomial Modelling and Its Applications*, Vol. 66 of *Monographs on Statistics and Applied Probability*. Chapman and Hall.

Freimer, M., Kollia, G., Mudholkar, G. S., and Lin, C. T. (1988). "A study of the generalized Tukey lambda family." *Comm. Stat. Theor. Meth.*, 17, 3547–3567.

Ghanem, R. and Spanos, P. (2003). *Stochastic Finite Elements: A Spectral Approach*. Courier Dover Publications, Mineola, 2nd edition.

Hall, P., Racine, J., and Li, Q. (2004). "Cross-validation and the estimation of conditional probability densities." *J. Amer. Stat. Assoc.*, 99, 1015–1026.

Hastie, T. and Tibshirani, R. (1990). *Generalized Additive Models*, Vol. 43 of *Monographs on Statistics and Applied Probability*. Chapman and Hall.

Karian, Z. A. and Dudewicz, E. J. (2000). *Fitting Statistical Distributions: The Generalized Lambda Distribution and Generalized Bootstrap Methods*. Chapman and Hall/CRC.

Karian, Z. A. and Dudewicz, E. J. (2010). *Handbook of Fitting Statistical Distributions with R*. Taylor & Francis.

McCullagh, P. and Nelder, J. (1989). *Generalized Linear Models*, Vol. 37 of *Monographs on Statistics and Applied Probability*. Chapman and Hall.

Moutoussamy, V., Nanty, S., and Pauwels, B. (2015). "Emulators for stochastic simulation codes." *ESAIM: Math. Model. Num. Anal.*, 48, 116–155.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press, Cambridge, Massachusetts, Internet edition.

Su, S. (2007). "Numerical maximum log likelihood estimation for generalized lambda distributions." *Comput. Stat. Data Anal.*, 51(8), 3983–3998.

Sudret, B. (2015). *Polynomial chaos expansions and stochastic finite element methods*. Risk and Reliability in Geotechnical Engineering, Taylor and Francis, Chapter 6, 265–300.

Tsybakov, A. B. (2009). *Introduction to Nonparametric Estimation*. Springer Series in Statistics. Springer, Cambridge, Massachusetts.

Xiu, D. and Karniadakis, G. E. (2002). "The Wiener-Askey polynomial chaos for stochastic differential equations." *SIAM J. Sci. Comput.*, 24(2), 619–644.