# Global sensitivity analysis for stochastic simulators based on generalized lambda surrogate models

X. Zhu, B. Sudret

# ETH*zürich*

# GLOBAL SENSITIVITY ANALYSIS FOR STOCHASTIC SIMULATORS BASED ON GENERALIZED LAMBDA SURROGATE MODELS

## X. Zhu and B. Sudret

CHAIR OF RISK, SAFETY AND UNCERTAINTY QUANTIFICATION
STEFANO-FRANSCINI-PLATZ 5
CH-8093 ZÜRICH

Risk, Safety & Uncertainty Quantification

Data Sheet

# Global sensitivity analysis for stochastic simulators based on generalized lambda surrogate models

Xujia Zhu [*1] and Bruno Sudret[†1]

[1]*Chair of Risk, Safety and Uncertainty Quantification, ETH Zürich, Stefano-Franscini-Platz 5, 8093 Zürich, Switzerland*

May 31, 2021

## Abstract

Global sensitivity analysis aims at quantifying the impact of input variability onto the variation of the response of a computational model. It has been widely applied to deterministic simulators, for which a set of input parameters has a unique corresponding output value. Stochastic simulators, however, have intrinsic randomness due to their use of (pseudo)random numbers, so they give different results when run twice with the same input parameters but non-common random numbers. Due to this random nature, conventional Sobol' indices, used in global sensitivity analysis, can be extended to stochastic simulators in different ways. In this paper, we discuss three possible extensions and focus on those that depend only on the statistical dependence between input and output. This choice ignores the detailed data generating process involving the internal randomness, and can thus be applied to a wider class of problems. We propose to use the generalized lambda model to emulate the response distribution of stochastic simulators. Such a surrogate can be constructed without the need for replications. The proposed method is applied to three examples including two case studies in finance and epidemiology. The results confirm the convergence of the approach for estimating the sensitivity indices even with the presence of strong heteroskedasticity and small signal-to-noise ratio.

## 1 Introduction

Computational models, a.k.a. simulators, have been extensively used to represent physical phenomena and engineering systems. They can help assess the reliability, control the risk and optimize the behavior of complex systems early at the design stage. Conventional simulators are usually deterministic, in the sense that repeated model evaluations with the same input

---

[*]zhu@ibk.baug.ethz.ch

[†]sudret@ethz.ch

parameters yield the same value of the output. In contrast, several runs of a stochastic simulator for a given set of input parameters provide different results. More precisely, the output of a stochastic simulator is a random variable following an unknown probability distribution. Hence, each model evaluation with the same input values generates a realization of the random variable. Mathematically, a stochastic simulator can be defined by

$$
\begin{aligned}
\mathcal{M}_s : \mathcal{D}_{\boldsymbol{X}} \times \Omega &\to \mathbb{R} \\
(\boldsymbol{x}, \omega) &\mapsto \mathcal{M}_s(\boldsymbol{x}, \boldsymbol{Z}(\omega)),
\end{aligned}
\tag{1}
$$

where $\boldsymbol{x}$ is the input vector that belongs to the input space $\mathcal{D}_{\boldsymbol{X}}$, and $\Omega$ denotes the probability space that represents the stochasticity. The intrinsic randomness is due to the fact that some *latent variables* $\boldsymbol{Z}(\omega)$ inside the model are not explicitly considered as a part of the input variables: given a fixed input value $\boldsymbol{x}_0$, the output of the simulator is a random variable.

In this respect, we can consider a stochastic simulator as a random field indexed by the parameters $\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}}$ [1]. For a given realization of the latent variables $\boldsymbol{z}_0$, the simulator becomes a deterministic function of $\boldsymbol{x}$. This is realized in practice by initializing the random seed to the same value before running the simulator for different $\boldsymbol{x}$' s, a trick known as *common random numbers*. The (classical) functions $\boldsymbol{x} \mapsto \mathcal{M}_s(\boldsymbol{x}, \boldsymbol{z}_0)$ will be called *trajectories* in this paper. One particular trajectory corresponds to one particular value $\boldsymbol{z}_0$.

In contrast, for a given $\boldsymbol{x}_0 \in \mathcal{D}_{\boldsymbol{X}}$, the output of the stochastic simulator is a random variable. Its distribution can be obtained by repeatedly running the simulator with $\boldsymbol{x}_0$, yet different realizations of the latent variables called *replications*.

Stochastic simulators are ubiquitous in modern engineering, finance and medical sciences. Typical examples include stochastic differential equations (e.g., financial models [2]) and agent-based models (e.g., epidemiological models [3]). To a certain extent, physical experiments can also be considered as stochastic models, because we may not be able to measure and consider all the relevant variables that can uniquely determine the experimental conditions.

In practice, the input variables may be affected by uncertainty due to noisy measurements, expert judgment or lack of knowledge. Therefore, they are modeled as random variables and grouped into a random vector $\boldsymbol{X} = (X_1, X_2, \ldots, X_M)$, which is characterized by a joint distribution $f_{\boldsymbol{X}}$. Quantification of the contribution of input variability to the output uncertainty is a major task in sensitivity analysis [4]. It allows us to identify the most important set of input variables that dominate the output variability and also to figure out non-influential variables. This information provides more insights into the simulator and can be further used for model calibrations and decision making [5].

A large number of methods have been successfully developed to perform sensitivity analysis in the context of deterministic simulators [4, 6, 7]. Among others, the variance-based sensitivity analysis, also referred to as Sobol' indices [8], is one of the most popular approaches, which relies on the analysis of variance. Several extensions of Sobol' indices to stochastic simulators

can be found in the literature, depending on the treatment of the intrinsic randomness. It is worth emphasizing that the overall uncertainty now consists of two parts, namely the inherent stochasticity in the latent variables and the uncertainty in the input parameters $\boldsymbol{X}$. Iooss and Ribatet [9] include the latent variables as a part of the input, which results in a natural extension of the classical Sobol' indices to stochastic simulators. Hart et al. [10] and Jimenez et al. [11] define the Sobol' indices as functions of the latent variables which, as a consequence, become random variables whose statistical properties can be studied. Recently, Azzi et al. [12] propose to represent the intrinsic randomness by the entropy of the response distribution and to calculate the classical Sobol' indices on the latter. All in all, relatively little attention has been devoted to sensitivity analysis for stochastic simulators.

Sensitivity analysis usually requires a large number of model evaluations for different realizations of the input vector. Due to the intrinsic randomness of stochastic simulators, an additional layer of stochasticity comes on top of the input uncertainty, which requires repeated runs with the same input parameters to fully characterize the model response. As a consequence, such analyses become intractable when the simulator is expensive to evaluate. To alleviate the computational burden, surrogate models can be constructed to mimic the original numerical model at a smaller computational cost. Large efforts have been dedicated to emulating the mean and variance function of stochastic simulators [13–15]. These two functions provide only the first two moments of the response distribution and are mostly used to estimate the Sobol' indices proposed in [9]. In recent papers [16, 17], we developed a novel surrogate model, called *generalized lambda model* (GLaM), to emulate the whole response distribution of stochastic simulators. This model uses generalized lambda distributions (GLD) [18] to flexibly approximate the response distribution, while the distribution parameters cast as functions of the inputs are approximated through polynomial chaos expansions (PCE) [19].

Based on these premises, the goal of this paper is to establish a clear framework to carry out global sensitivity analysis for stochastic simulators, and to propose efficient computational approaches based on GLaM stochastic emulators. Therefore, the original contributions of this paper are two-fold. On the one hand, we give a thorough review of the current development of global sensitivity analysis for stochastic simulators. We point out the nature and the properties of different extensions of Sobol' indices, which provides a general guideline to their usage. On the other hand, we present a unified framework based on generalized lambda models to calculate a whole variety of global sensitivity indices using this single surrogate.

The paper is organized as follows. First, we review three extensions of Sobol' indices to stochastic simulators in Section 2. In Section 3, we present the framework of GLaMs [16]. There, we recap the fitting procedure proposed in [17], where it is emphasized that there is no need for replicated runs. Then, we discuss the use of GLaMs for estimating different types of Sobol' indices. In Section 4, we illustrate the performance of GLaMs on three examples. While the first example is analytical, the second and third ones are realistic case studies in finance and epidemiology, respectively. Finally, we summarize the main findings of the paper and provide an outlook for

3

future research in Section 5.

## 2 Global sensitivity analysis of stochastic simulators

### 2.1 Sobol' indices

Variance-based sensitivity analysis has been extensively studied and successfully developed in the context of deterministic simulators. For a deterministic model $\mathcal{M}_d$, Sobol' indices quantify the contribution of each input variable $\{X_i, i = 1, \ldots, M\}$, or combination thereof, to the variance of the model output $Y = \mathcal{M}_d(\boldsymbol{X})$.

In this paper, we assume that $X_i$'s are mutually independent. Let us split the input vector into two subsets $\boldsymbol{X} = (\boldsymbol{X_u}, \boldsymbol{X_{\sim u}})$, where $\mathbf{u} \subset \{1, \ldots, M\}$ and $\sim \mathbf{u}$ is the complement of $\mathbf{u}$, i.e., $\sim \mathbf{u} = \{1, \ldots, M\} \setminus \mathbf{u}$. From the total variance theorem, the variance of the output can be decomposed as

$$\mathrm{Var}\left[Y\right] = \mathbb{E}\left[\mathrm{Var}\left[Y \mid \boldsymbol{X_u}\right]\right] + \mathrm{Var}\left[\mathbb{E}\left[Y \mid \boldsymbol{X_u}\right]\right]. \tag{2}$$

The first-order and total Sobol' indices introduced by Sobol' [8] and Homma and Saltelli [20] for the subset of input variables $\boldsymbol{X_u}$ are defined by

$$S_{\mathbf{u}} \overset{\text{def}}{=} \frac{\mathrm{Var}\left[\mathbb{E}\left[Y \mid \boldsymbol{X_u}\right]\right]}{\mathrm{Var}\left[Y\right]}, \quad S_{T_{\mathbf{u}}} \overset{\text{def}}{=} 1 - \frac{\mathrm{Var}\left[\mathbb{E}\left[Y \mid \boldsymbol{X_{\sim u}}\right]\right]}{\mathrm{Var}\left[Y\right]} = 1 - S_{\sim \mathbf{u}}. \tag{3}$$

Higher-order Sobol' indices can be defined with the help of $S_{\mathbf{u}}$. For example, the second-order or two-factor interaction Sobol' index of $X_1$ and $X_2$ is given by

$$S_{1,2} \overset{\text{def}}{=} S_{\{1,2\}} - S_1 - S_2, \tag{4}$$

where we denote $S_{\{i\}}$ by $S_i$ for the sake of simplicity.

In the context of stochastic simulators defined in Equation (1), the input variables alone do not determine the value of the output. Iooss and Ribatet [9] extend $\boldsymbol{X}$ by adding the internal source of randomness represented by latent variables $\boldsymbol{Z}$, which turns the stochastic simulator into a deterministic one. In this case, all the input variables are gathered in $(\boldsymbol{X_u}, \boldsymbol{X_{\sim u}}, \boldsymbol{Z})$, and thus the Sobol' indices in Equation (3) can be naturally extended to

$$S_{\mathbf{u}} \overset{\text{def}}{=} \frac{\mathrm{Var}\left[\mathbb{E}\left[Y \mid \boldsymbol{X_u}\right]\right]}{\mathrm{Var}\left[Y\right]}, \quad S_{T_{\mathbf{u}}} \overset{\text{def}}{=} 1 - \frac{\mathrm{Var}\left[\mathbb{E}\left[Y \mid \boldsymbol{X_{\sim u}}, \boldsymbol{Z}\right]\right]}{\mathrm{Var}\left[Y\right]}. \tag{5}$$

Note that $S_{\mathbf{u}}$ has the same expression as in the case of deterministic simulators, but $S_{T_{\mathbf{u}}}$ contains the additional variables $\boldsymbol{Z}$ [14]. Similarly, higher-order Sobol' indices corresponding to interactions among components of $\boldsymbol{X}$ are defined in the same way as deterministic simulators, whereas interactions between components of $\boldsymbol{X}$ and $\boldsymbol{Z}$ involve $\boldsymbol{Z}$ in their definition. Since this is a direct extension, the Sobol' indices defined in Equation (5) are referred to as *classical Sobol' indices* in the sequel.

Another way to extend Sobol' indices to stochastic simulators is to first eliminate the internal randomness by representing the response random variable $Y(\boldsymbol{x})$ by some summarizing statistical quantity, called here quantity of interest (QoI) denoted by $\text{QoI}(\boldsymbol{x})$, such as the mean value $m(\boldsymbol{x})$, variance $v(\boldsymbol{x})$ [9], $\alpha$-quantile $q_\alpha(\boldsymbol{x})$ [21] and differential entropy $h(\boldsymbol{x})$ [12]. As a result, the stochastic simulator is reduced to a deterministic function $\text{QoI}(\boldsymbol{x})$, and we can calculate the associated *QoI-based Sobol' indices* as follows:

$$S_{\mathbf{u}}^{\text{QoI}} \stackrel{\text{def}}{=} \frac{\text{Var}\left[\mathbb{E}\left[\text{QoI}(\boldsymbol{X})|\boldsymbol{X}_{\mathbf{u}}\right]\right]}{\text{Var}\left[\text{QoI}(\boldsymbol{X})\right]}, \quad S_{T_{\mathbf{u}}}^{\text{QoI}} \stackrel{\text{def}}{=} 1 - \frac{\text{Var}\left[\mathbb{E}\left[\text{QoI}(\boldsymbol{X})|\boldsymbol{X}_{\sim\mathbf{u}}\right]\right]}{\text{Var}\left[\text{QoI}(\boldsymbol{X})\right]}. \tag{6}$$

A third extension is defined by considering a stochastic simulator as a random field. For a fixed internal randomness $\boldsymbol{Z}(\omega) = \boldsymbol{z}$, the stochastic simulator is a deterministic function of the input variables, which corresponds to a trajectory. Hence, the associated Sobol' indices are well-defined. Yet, they are random variables because of their dependence on $\boldsymbol{Z}$ [10], which results in the *trajectory-based Sobol' indices*:

$$S_{\mathbf{u}}^{\text{traj}}(\boldsymbol{Z}) \stackrel{\text{def}}{=} \frac{\text{Var}_{\boldsymbol{X}_{\mathbf{u}}}\left[\mathbb{E}\left[Y \mid \boldsymbol{X}_{\mathbf{u}}, \boldsymbol{Z}\right]\right]}{\text{Var}\left[Y \mid \boldsymbol{Z}\right]}, \quad S_{T_{\mathbf{u}}}^{\text{traj}}(\boldsymbol{Z}) \stackrel{\text{def}}{=} 1 - \frac{\text{Var}_{\boldsymbol{X}_{\sim\mathbf{u}}}\left[Y \mid \boldsymbol{X}_{\sim\mathbf{u}}, \boldsymbol{Z}\right]}{\text{Var}\left[Y \mid \boldsymbol{Z}\right]}, \tag{7}$$

where the indices of the variance operators correspond to those variables to which these operators apply.

## 2.2 Discussion

The three types of Sobol' indices introduced above have different nature and focus. The classical Sobol' indices defined in Equation (5) treat the latent variables $\boldsymbol{Z}$ as a set of separate input variables. As a result, indices of this type treat $\boldsymbol{Z}$ in the same way as $\boldsymbol{X}$. The first-order index $S_{\mathbf{u}}$ indicates how much the output variance can be reduced (in expectation) if we can fix the value of $\boldsymbol{X}_{\mathbf{u}}$. Besides, the classical Sobol' indices can also quantify the influence of the intrinsic randomness as well as its interactions with input variables.

The QoI-based Sobol' indices defined in Equation (6) help study a specific statistical quantity of the model response, which is a deterministic function of the inputs. Using a summary quantity to represent the random output might lead to a loss of information [10], unless this quantity itself is of interest. For example, we may want to find the variable(s) that has the largest effect(s) on the 95% quantile (i.e., $q_\alpha(\boldsymbol{X})$ for $\alpha = 0.95$) of the model response. However, the importance (ranking) of the inputs $X_i$'s can be quite different depending on the choice of the QoI.

Unlike the previous two types of indices, trajectory-based Sobol' indices presented in Equation (7) are random variables. This is because the latent variables $\boldsymbol{Z}$ and the input parameters $\boldsymbol{X}$ are treated differently: conditioned on a given $\boldsymbol{Z} = \boldsymbol{z}_0$, the stochastic simulator reduces to a deterministic function of $\boldsymbol{X}$, and we can calculate the associated (classical) Sobol' indices. To evaluate the probability distribution of these trajectory-based Sobol' indices requires that the same random seeds can be explicitly fixed in the simulator when running it for different values

of $\boldsymbol{x}$. In a sense, trajectory-based Sobol' indices emphasize the variation of the trajectory of stochastic simulators. They typically show the importance of each input variable in terms of its contribution to the variability of trajectories.

For stochastic simulators, the question "which input variable has the strongest effect?" is rather vague and cannot be answered by a single type of index. The analyst should properly pose the problem and select the appropriate sensitivity measures. If one aims at reducing the variance of the model output, the classical Sobol' indices are of interest. If one is interested in some summary QoIs (which is often the case for applications, e.g., quantiles in reliability analysis), the QoI-based indices are more appropriate. Finally, if one can control the internal randomness and is interested in the variability of the model output for fixed intrinsic stochasticity, trajectory-based indices should be selected.

To further illustrate this discussion, let's consider the stochastic simulator

$$Y(\boldsymbol{X}) = \mathcal{M}_s(\boldsymbol{X}, Z) = X_1 + X_2 \cdot Z$$

where $X_1$, $X_2$ and $Z$ are independent random variables following standard normal distributions. The classical first-order Sobol' indices are $S_1 = 0.5$ and $S_2 = 0$. Therefore, if we want to primarily reduce the variance of $Y$, $X_1$ should be investigated. For the response mean function $m(\boldsymbol{X}) = X_1$, we have $S_1^m = 1$ and $S_2^m = 0$, which indicates that $X_1$ contributes fully to the variation of the mean function. In contrast, for the variance function $v(\boldsymbol{X}) = X_2^2$, $S_1^v = 0$ and $S_2^v = 1$ reveals a different order. Regarding the trajectory-based indices, we have $S_1^{\text{traj}}(Z) = \frac{1}{1+Z^2}$ and $S_2^{\text{traj}}(Z) = \frac{Z^2}{1+Z^2}$ which are two random variables (due to the randomness in $Z$). The probability distribution of the two variables characterize how the randomness in $\boldsymbol{X}$ affects the function $\mathcal{M}_s(\boldsymbol{X}, z)$ with $z$ being fixed as a single realization of $Z$. In summary, even for this simple toy example, the various indices provide different conclusions.

It is worth remarking that the classical Sobol' indices $S_{\mathbf{u}}$ in Equation (5) share some common properties with the mean-based Sobol' indices in Eq. (6), when we consider the mean function $\text{QoI}(\boldsymbol{x}) \stackrel{\text{def}}{=} m(\boldsymbol{x}) = \mathbb{E}[Y \mid \boldsymbol{X} = \boldsymbol{x}]$:

$$S_{\mathbf{u}}^m = \frac{\text{Var}[\mathbb{E}[m(\boldsymbol{X})|\boldsymbol{X}_{\mathbf{u}}]]}{\text{Var}[m(\boldsymbol{X})]}. \tag{8}$$

According to the law of total expectation, we have

$$\mathbb{E}[Y \mid \boldsymbol{X}_{\mathbf{u}}] = \mathbb{E}[\mathbb{E}[Y \mid \boldsymbol{X}] \mid \boldsymbol{X}_{\mathbf{u}}] = \mathbb{E}[m(\boldsymbol{X}) \mid \boldsymbol{X}_{\mathbf{u}}]. \tag{9}$$

As a result, $S_{\mathbf{u}}$ can be rewritten as

$$S_{\mathbf{u}} = \frac{\text{Var}[\mathbb{E}[Y \mid \boldsymbol{X}_{\mathbf{u}}]]}{\text{Var}[Y]} = \frac{\text{Var}[\mathbb{E}[m(\boldsymbol{X}) \mid \boldsymbol{X}_{\mathbf{u}}]]}{\text{Var}[Y]}, \tag{10}$$

This implies that both classical Sobol' indices $S_{\mathbf{u}}$ and mean-based Sobol' indices $S_{\mathbf{u}}^m$ provide the same ranking, as the numerators of Equations (8) and (10) are identical. However, it is worth

emphasizing that $S_{\mathbf{u}}$ is not equal to $S_{\mathbf{u}}^m$ and they are measuring different quantities, since the denominator of Equation (10) is $\text{Var}\,[Y]$ but that of Equation (8) is $\text{Var}\,[m(\boldsymbol{X})]$.

As a summary for all three extensions, a stochastic simulator is essentially transformed into a reduced deterministic model at a certain stage. The classical Sobol' indices include the latent variables as a part of the inputs. The QoI-based Sobol' indices rely on a deterministic representation. The trajectory-based indices are random variables whose statistical properties can be studied at the cost of repeating a standard Sobol' analysis for different realizations of the latent variables separately. As a result, the three types of sensitivity indices can be estimated by modifying only slightly the standard methods based on Monte Carlo simulation developed for deterministic simulators [4].

The classical Sobol' indices involving $\boldsymbol{Z}$ (e.g., $S_{\boldsymbol{Z}}$, $S_{T_{\mathbf{u}}}$ in Eq. (5)) and the trajectory-based Sobol' indices require controlling the latent variables $\boldsymbol{Z}$. In practical computations, this is achieved by fixing the *random seed* in the computational model [10, 14]. However, for certain types of stochastic simulators, or when the data are generated by physical experiments, it may be difficult to control or even identify $\boldsymbol{Z}$. For the sake of general applicability, we focus in this paper only on Sobol' indices that can be estimated by manipulating $\boldsymbol{X}$, that is, the QoI-based Sobol' indices and, to some extent, the classical Sobol' indices.

Using Monte Carlo simulations to estimate these indices requires evaluating the simulator for various realizations of the input vector. In addition, it is generally necessary to evaluate the function $\text{QoI}(\boldsymbol{x})$ for calculating the associated QoI-based Sobol' indices. However, this function is not directly accessible due to the intrinsic randomness. Therefore this function is usually estimated by using replications; i.e., for each realization $\boldsymbol{x}$, the simulator is run many times, and $\text{QoI}(\boldsymbol{x})$ is estimated from the output samples. Both factors call for a large number of model runs, which becomes impracticable for costly models. Therefore, the use of surrogate models is unavoidable.

In the sequel, we present the generalized lambda model as a stochastic surrogate. Such a model emulates the response distribution conditioned on $\boldsymbol{X} = \boldsymbol{x}$, which fully characterizes the statistical dependence between the inputs and output. Therefore, it can be used to estimate the considered Sobol' indices.

## 3    Generalized lambda models

Generalized lambda models consist of mainly two parts: the generalized lambda distribution and polynomial chaos expansions. In this section, we briefly recap these two elements and present an algorithm to construct such a model without the need for replicated runs of the simulator. Then, we discuss how to estimate the sensitivity indices from the surrogate.

## 3.1 Generalized lambda distributions

The generalized lambda distribution is a flexible distribution family, which is designed to approximate many common distributions [18], e.g., normal, lognormal, Weibull and generalized extreme value distributions. A GLD is defined by its *quantile function* $Q(u)$ with $u \in [0, 1]$, that is, the inverse of the cumulative distribution function $Q(u) = F^{-1}(u)$. In this paper, we consider the GLD of the Freimer-Kollia-Mudholkar-Lin (FKML) family [22] with four parameters, whose quantile function is defined by

$$Q(u; \boldsymbol{\lambda}) = \lambda_1 + \frac{1}{\lambda_2} \left( \frac{u^{\lambda_3} - 1}{\lambda_3} - \frac{(1 - u)^{\lambda_4} - 1}{\lambda_4} \right), \tag{11}$$

where $\lambda_1$ is the location parameter, $\lambda_2$ is the scaling parameter, and $\lambda_3$ and $\lambda_4$ are the shape parameters. $\lambda_2$ is required to be positive to produce valid quantile functions (i.e., $Q$ being non-decreasing on $[0, 1]$). Based on the quantile function defined in Equation (11), the probability density function (PDF) of a random variable $Y$ following a GLD is given by

$$f_Y(y; \boldsymbol{\lambda}) = \frac{f_U(u)}{Q'(u; \boldsymbol{\lambda})} = \frac{\lambda_2}{u^{\lambda_3 - 1} + (1 - u)^{\lambda_4 - 1}} \mathbb{1}_{[0,1]}(u), \text{ with } u = Q^{-1}(y; \boldsymbol{\lambda}), \tag{12}$$

where $\mathbb{1}_{[0,1]}$ is the indicator function. A closed-form expression of $Q^{-1}$ is not available for arbitrary values of $\lambda_3$ and $\lambda_4$. Therefore, evaluating the PDF for a given $y$ usually requires solving the nonlinear equation (12) numerically.

GLDs cover a wide range of shapes determined by $\lambda_3$ and $\lambda_4$ [16]. For instance, $\lambda_3 = \lambda_4$ produces symmetric PDFs, and $\lambda_3 < \lambda_4$ ($\lambda_3 > \lambda_4$) results in left-skewed (resp. right-skewed) distributions. Moreover, $\lambda_3$ and $\lambda_4$ are closely linked to the support and the tail behaviors of the corresponding PDF. More precisely, $\lambda_3$ and $\lambda_4$ control the left and right tail, respectively. Whereas $\lambda_3 > 0$ implies that the PDF support is left-bounded, $\lambda_3 \leq 0$ implies that the distribution has a lower infinite support. Similarly, $\lambda_4 > 0$ implies that the PDF support is right-bounded, whereas it is $+\infty$ for $\lambda_4 \leq 0$. In addition, for $\lambda_3 < 0$ ($\lambda_4 < 0$), the left (resp. right) tail decays asymptotically as a power law. Hence, GLDs can also provide fat-tailed distributions. The reader is referred to [16, 22] for a longer presentation of GLDs.

## 3.2 Polynomial chaos expansions

Consider a deterministic model $\mathcal{M}_d$ that maps a set of input parameters $\boldsymbol{x} = (x_1, x_2, \ldots, x_M) \in \mathcal{D}_{\boldsymbol{X}} \subset \mathbb{R}^M$ to the output $y \in \mathbb{R}$. Under the assumption that $Y = \mathcal{M}_d(\boldsymbol{X})$ has finite variance, $\mathcal{M}_d$ belongs to the Hilbert space $\mathcal{H}$ of square-integrable functions with respect to the inner product $\langle u, v \rangle_{\mathcal{H}} = \mathbb{E}\left[u(\boldsymbol{X})v(\boldsymbol{X})\right] = \int_{\mathcal{D}_{\boldsymbol{X}}} u(\boldsymbol{x})v(\boldsymbol{x})f_{\boldsymbol{X}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$. If the joint distribution $f_{\boldsymbol{X}}$ satisfies certain conditions [23], the simulator $\mathcal{M}_d$ admits a spectral representation in terms of orthogonal polynomials:

$$Y = \mathcal{M}_d(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^M} c_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{X}), \tag{13}$$

where $\psi_{\boldsymbol{\alpha}}$ is a multivariate polynomial basis function indexed by $\boldsymbol{\alpha} \in \mathbb{N}^M$, and $c_{\boldsymbol{\alpha}}$ denotes the associated coefficient. The orthogonal basis can be obtained by using tensor products of univariate polynomials, each of which is orthogonal with respect to the probability measure $f_{X_i}(x_i)\mathrm{d}x_i$ of the $i$-th variable $X_i$:

$$\psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) = \prod_{j=1}^{M} \phi_{\alpha_j}^{(j)}(x_j). \tag{14}$$

Details about the construction of this generalized polynomial chaos expansion can be found in [24, 25].

The PCE defined in Equation (13) contains an infinite sum of terms. However, in practice, it is only feasible to use a finite series as an approximation. To this end, truncation schemes are adopted to select a set of basis functions defined by a finite subset $\mathcal{A} \subset \mathbb{N}^M$ of multi-indices. A typical scheme is the hyperbolic (a.k.a. $q$-norm) truncation scheme [26] given by

$$\mathcal{A}^{p,q,M} = \left\{ \boldsymbol{\alpha} \in \mathbb{N}^M : \|\boldsymbol{\alpha}\|_q \overset{\text{def}}{=} \left( \sum_{i=1}^{M} |\alpha_i|^q \right)^{\frac{1}{q}} \leq p \right\}, \tag{15}$$

where $p$ is the maximum degree of polynomials, and $q \leq 1$ defines the quasi-norm $\|\cdot\|_q$. Note that with $q = 1$, we obtain the full basis of total degree less than $p$, which corresponds to the *standard truncation scheme*.

## 3.3 Formulation of generalized lambda models

Because of the flexibility of GLDs, we assume that the response random variable $Y(\boldsymbol{x})$ of a stochastic simulator for a given input vector $\boldsymbol{x}$ can be well approximated by a GLD. Hence, the associated distribution parameters $\boldsymbol{\lambda}$ are functions of the input variables:

$$Y(\boldsymbol{x}) \sim \mathrm{GLD}\left(\lambda_1(\boldsymbol{x}), \lambda_2(\boldsymbol{x}), \lambda_3(\boldsymbol{x}), \lambda_4(\boldsymbol{x})\right). \tag{16}$$

Under appropriate conditions discussed in Section 3.2, each component of $\boldsymbol{\lambda}(\boldsymbol{x})$ can be represented by a series of orthogonal polynomials. Because $\lambda_2(\boldsymbol{x})$ is required to be positive (see Section 3.1), the associated polynomial chaos representation is built on the natural logarithmic transform $\log(\lambda_2(\boldsymbol{x}))$. This results in the following approximations:

$$\lambda_l(\boldsymbol{x}) \approx \lambda_s^{\mathrm{PC}}(\boldsymbol{x};\boldsymbol{c}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}_l} c_{l,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}), \quad l = 1, 3, 4, \tag{17}$$

$$\lambda_2(\boldsymbol{x}) \approx \lambda_2^{\mathrm{PC}}(\boldsymbol{x};\boldsymbol{c}) = \exp\left( \sum_{\boldsymbol{\alpha} \in \mathcal{A}_2} c_{2,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) \right), \tag{18}$$

where $\mathcal{A}_l$ ($l = 1, 2, 3, 4$) is a finite set of selected basis functions for $\lambda_l$, and $c_{l,\boldsymbol{\alpha}}$'s are the coefficients. For the purpose of clarity, we explicitly express $\boldsymbol{c}$ in the PC approximations $\lambda_l^{\mathrm{PC}}(\boldsymbol{x};\boldsymbol{c})$ to emphasize that $\boldsymbol{c}$ are the model parameters yet to be estimated from data.

### 3.4 GLaM constructions

We assume that our costly stochastic simulator is evaluated once for each point $\boldsymbol{x}^{(i)}$ of the experimental design $\mathcal{X}$, and the associated model response $y^{(i)}$ is collected in $\mathcal{Y}$:

$$\mathcal{X} = \left\{ \boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)} \right\}, \quad \mathcal{Y} = \left\{ \mathcal{M}_s \left( \boldsymbol{x}^{(1)}, \omega^{(1)} \right), \dots, \mathcal{M}_s \left( \boldsymbol{x}^{(N)}, \omega^{(N)} \right) \right\} \tag{19}$$

As already mentioned (and as emphasized by the notation $\omega^{(i)}$), no replications are required, and we do not control the random seed. To construct a GLaM from the available data $(\mathcal{X}, \mathcal{Y})$, both the truncated sets $\boldsymbol{\mathcal{A}}$ of basis functions and the coefficients $\boldsymbol{c}$ shall be determined. In this section, we summarize the method proposed in [17], which is designed to achieve both purposes without the need for replications.

Sometimes prior knowledge is available to set the basis functions. For example, when working with a standard linear regression problem, the data is supposed to be generated by

$$Y = \beta_0 + \boldsymbol{X}\boldsymbol{\beta} + \epsilon, \tag{20}$$

where $\epsilon$ has mean zero and is independent of $\boldsymbol{X}$. This case can be treated within the GLaM framework as follows: $\mathcal{A}_1$ contains the constant and linear term, and $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$ contain only a constant term. Note that such a GLaM allows estimating the distribution of $\epsilon$, which is not required to be normal, whereas the usual linear regression framework assumes normally distributed $\epsilon$.

However, in general there is no prior knowledge that would help select $\boldsymbol{\mathcal{A}}$. Thus, we make the following assumptions to find appropriate hyperbolic truncation schemes defined in Equation (15) for each $\lambda_i, i = 1, \dots, 4$:

(A1) The response distribution of $Y(\boldsymbol{x})$ can be well-approximated by a generalized lambda distribution;

(A2) The shape of this distribution smoothly varies as a function of $\boldsymbol{x}$, so that the parameters $\lambda_i(\boldsymbol{x})$ can be well approximated by a low-order PCE

Because the shape of a GLD is controlled by $\lambda_3$ and $\lambda_4$, the associated hyperbolic truncation schemes $\mathcal{A}^{p,q,M}$ can be set with a small value of $p$, say $p = 1$.

Moreover, the parameters $\lambda_1(\boldsymbol{x})$ and $\lambda_2(\boldsymbol{x})$ mainly affect the variation of the mean $m(\boldsymbol{x})$ and of the variance $v(\boldsymbol{x})$ as a function of the input $\boldsymbol{x}$, respectively. As a result, they may require possibly larger degree $p$. To this end, we modify the feasible generalized least-squares (FGLS) [27] to find suitable truncation schemes for the mean and variance function modeled as

$$m(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}_m} c_{m,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}), \quad v(\boldsymbol{x}) = \exp \left( \sum_{\boldsymbol{\alpha} \in \mathcal{A}_v} c_{v,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) \right).$$

Basically, FGLS iterates between a weighted least-square problem (WLS) to fit the mean function, and an ordinary least-square (OLS) analysis to estimate the variance function.

The details of the modified FGLS are presented in Algorithm 1. In this algorithm, the inputs $\boldsymbol{p}_1$ and $\boldsymbol{q}_1$ stand for the set of candidate degrees and $q$-norms that are tested to expand $\lambda_1(\boldsymbol{x})$, respectively. The same notation apply to $\boldsymbol{p}_2$ and $\boldsymbol{q}_2$ for $\lambda_2(\boldsymbol{x})$. Indeed, because of the low cost of least-square analysis, various combinations of $p$ and $q$ are tested for both $\lambda_1(\boldsymbol{x})$ and $\lambda_2(\boldsymbol{x})$.

More precisely, AOLS denotes *adaptive ordinary least-squares* with degree and $q$-norm adaptivity [28, 29]. This algorithm first builds a series of PCEs, each of which is obtained by applying ordinary least-squares with a truncation scheme $\mathcal{A}^{p,q,M}$ defined by a combination of $p \in \boldsymbol{p}$ and $q \in \boldsymbol{q}$. Then, it selects the PCE, therefore the associated truncation scheme, with the smallest *leave-one-out* errors (see [29] for details). WLS denotes the use of weighted least-squares, which takes the estimated variance $\hat{\boldsymbol{v}}$ as weight to re-estimate $\boldsymbol{c}_m$. In this procedure, the truncation set $\mathcal{A}_m$ for $m(\boldsymbol{x})$ is selected only once (before the loop), whereas a set of truncation schemes $\{\mathcal{A}_v^i : i = 1, \ldots, N_{\text{FGLS}}\}$ is obtained.

We finally select the one with the smallest leave-one-out error as the final truncated set $\mathcal{A}_v$ for $v(\boldsymbol{x})$. The number of iterations $N_{\text{FGLS}}$ is defined by the user, typically $N_{\text{FGLS}} = 5\text{--}10$. After applying Algorithm 1, we set $\mathcal{A}_1 = \mathcal{A}_m$ and $\mathcal{A}_2 = \mathcal{A}_v$.

---

**Algorithm 1** Modified feasible generalized least-squares

1: Input: $(\mathcal{X}, \mathcal{Y})$, $\boldsymbol{p}_1$, $\boldsymbol{q}_1$, $\boldsymbol{p}_2$, $\boldsymbol{q}_2$
2: Output: truncated sets for the mean and variance function–$\mathcal{A}_m$ and $\mathcal{A}_v$
3: $\mathcal{A}_m$, $\hat{\boldsymbol{c}}_m \leftarrow \text{AOLS}\,(\mathcal{X}, \mathcal{Y}, \boldsymbol{p}_1, \boldsymbol{q}_1)$
4: **for** $i \leftarrow 1, \ldots, N_{\text{FGLS}}$ **do**
5:     $\hat{\boldsymbol{m}} \leftarrow \sum_{\boldsymbol{\alpha} \in \mathcal{A}_m} c_{m,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\mathcal{X})$
6:     $\tilde{\boldsymbol{r}} \leftarrow 2\log\left(|\mathcal{Y} - \hat{\boldsymbol{m}}|\right)$
7:     $\mathcal{A}_v^i$, $\hat{\boldsymbol{c}}_v$, $\varepsilon_{\text{LOO}}^i \leftarrow \text{AOLS}\,(\mathcal{X}, \tilde{\boldsymbol{r}}, \boldsymbol{p}_2, \boldsymbol{q}_2)$
8:     $\hat{\boldsymbol{v}} \leftarrow \exp\left(\sum_{\boldsymbol{\alpha} \in \mathcal{A}_v} c_{v,\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\mathcal{X})\right)$
9:     $\hat{\boldsymbol{c}}_m \leftarrow \text{WLS}\,(\mathcal{X}, \mathcal{Y}, \mathcal{A}_m, \hat{\boldsymbol{v}})$
10: **end for**
11: $i^* = \arg\min\left\{\varepsilon_{\text{LOO}}^i : i = 1, \ldots, N_{\text{FGLS}}\right\}$ and $\mathcal{A}_v \leftarrow \mathcal{A}_v^{i^*}$

---

Once the basis functions are selected, we use the maximum (conditional) likelihood estimator to estimate $\boldsymbol{c}$:

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c} \in \mathcal{C}} \mathsf{L}\,(\boldsymbol{c})\,, \tag{21}$$

where $\mathsf{L}\,(\boldsymbol{c})$ is the conditional negative log-likelihood

$$\mathsf{L}\,(\boldsymbol{c}) = \sum_{i=1}^{N} -\log\left(f^{\text{GLD}}\left(y^{(i)}; \boldsymbol{\lambda}^{\text{PC}}\left(\boldsymbol{x}^{(i)}; \boldsymbol{c}\right)\right)\right)\,, \tag{22}$$

with $f^{\text{GLD}}$ being the probability density function of the GLD defined in Equation (12).

The advantages of the proposed estimator are twofold. On the one hand, the simulator is required to be evaluated only once (but not limited to one) on each point of the experimental design. Thereby, replications are not necessary (yet possible), and the method is versatile in this respect. On the other hand, if the underlying computational model can be exactly represented by a GLaM for a specific choice of $c$, the maximum likelihood estimator is *consistent* (see proof in [17]).

In practice, the evaluation of $\mathsf{L}(c)$ is not straightforward because the PDF of generalized lambda distributions does not have an explicit form: it is necessary to solve nonlinear equations as shown in Eq. (12). Nevertheless, the nonlinear function $Q(u; \boldsymbol{\lambda})$ is monotonic and defined on $[0, 1]$. Therefore, we proposed using the bisection method [30] to efficiently solve the nonlinear equations.

## 3.5 Sensitivity analysis with GLaMs

### 3.5.1 Introduction

The various Sobol' indices introduced in Equation (5) and (6) can be estimated by sampling from the conditional distribution $Y \mid \boldsymbol{X_{\mathsf{u}}}$. Because of the specific format of the GLaM definition, such a sampling can be easily performed.

The generalized lambda distribution parameterizes the quantile function $Q(u; \boldsymbol{\lambda})$ (see Equation (11)), which can be seen as the inverse probability integral transform. In other words, the random variable $Q(U; \boldsymbol{\lambda})$ with $U \sim \mathcal{U}(0, 1)$ follows GLD($\boldsymbol{\lambda}$). As a result, sampling from a GLD is straightforward. We define the function $Q_{\mathrm{GLaM}} : (u, \boldsymbol{x}) \in [0, 1] \times \mathcal{D}_{\boldsymbol{X}} \mapsto \mathbb{R}$ by

$$Q_{\mathrm{GLaM}}(u; \boldsymbol{x}) = Q(u; \boldsymbol{\lambda}^{\mathrm{PC}}(\boldsymbol{x})) = \lambda_1^{\mathrm{PC}}(\boldsymbol{x}) + \frac{1}{\lambda_2^{\mathrm{PC}}(\boldsymbol{x})} \left( \frac{u^{\lambda_3^{\mathrm{PC}}(\boldsymbol{x})} - 1}{\lambda_3^{\mathrm{PC}}(\boldsymbol{x})} - \frac{(1 - u)^{\lambda_4^{\mathrm{PC}}(\boldsymbol{x})} - 1}{\lambda_4^{\mathrm{PC}}(\boldsymbol{x})} \right). \quad (23)$$

$Q_{\mathrm{GLaM}}(U; \boldsymbol{x})$ is a so-called *GLaM stochastic emulator* where $U \sim \mathcal{U}(0, 1)$ serves as a latent variable that introduces the internal source of randomness. Precisely, $Q_{\mathrm{GLaM}}(U; \boldsymbol{x})$ is a random variable following the surrogate response PDF for $\boldsymbol{X} = \boldsymbol{x}$, and $Q_{\mathrm{GLaM}}(u; \boldsymbol{x})$ provides its corresponding $u$-quantile. In other words, GLaM is a simple stochastic surrogate model with only one latent variable (namely, $U$); this surrogate behaves similarly to the original stochastic simulator in terms of the response distribution for any $\boldsymbol{x}$.

Equation (23) emulates the conditional quantile function $Q_{Y|\boldsymbol{X}}(u; \boldsymbol{x})$ of the original model, so calculating Sobol' indices $S_{\mathsf{u}}$ of the deterministic function $Q_{\mathrm{GLaM}}(u; \boldsymbol{x})$ can directly provide the classical Sobol' indices defined in Equation (5). Note that $Q_{\mathrm{GLaM}}$ also allows us to calculate classical Sobol' indices involving $U$, e.g., $S_U$. However, since the surrogate approximates only the response distribution but cannot produce the trajectories, these Sobol' indices are not representative of those of the original model, e.g., $S_{\boldsymbol{Z}}$ that requires estimating $\mathrm{Var}\left[\mathbb{E}\left[Y \mid \boldsymbol{Z}\right]\right]$ (according to Equation (5)), where the inner expectation $\mathbb{E}\left[Y \mid \boldsymbol{z}\right]$ is taken over a trajectory.

For QoI-based Sobol' indices in Equation (6), if the quantity of interest $q_{\mathrm{GLaM}}(\boldsymbol{x})$ can be directly calculated from generalized lambda distributions, we can just treat it as a classical surrogate

model of QoI$(\boldsymbol{x})$. This is the case for the mean $m(\boldsymbol{x})$ and the variance $v(\boldsymbol{x})$ (see Section 6.1 for details). In addition, if QoI$(\boldsymbol{x})$ is a $u$-quantile of the response distribution, Equation (23) is used directly.

Finally, if it is impossible to evaluate analytically $q_{\mathrm{GLaM}}(\boldsymbol{x})$, we generate a large sample set from Equation (23) by sampling $U \sim \mathcal{U}(0,1)$, and then use the sample statistic $\hat{q}_{\mathrm{GLaM}}(\boldsymbol{x})$ as a surrogate model for QoI$(\boldsymbol{x})$.

### 3.5.2 Monte Carlo estimates

Because both $Q_{\mathrm{GLaM}}(u;\boldsymbol{x})$ and $q_{\mathrm{GLaM}}(\boldsymbol{x})$ are deterministic, we can use methods based on Monte Carlo simulations [20] to estimate the considered Sobol' indices. Here, we illustrate the estimator suggested by Janon et al. [31] for classical Sobol' indices estimations.

We first define two random variables $Y = Q_{\mathrm{GLaM}}(U; \boldsymbol{X}_{\mathbf{u}}, \boldsymbol{X}_{\sim\mathbf{u}})$ and $Y_{\mathbf{u}} = Q_{\mathrm{GLaM}}(\tilde{U}; \boldsymbol{X}_{\mathbf{u}}, \tilde{\boldsymbol{X}}_{\sim\mathbf{u}})$, where $\tilde{U}$ and $\tilde{\boldsymbol{X}}_{\sim\mathbf{u}}$ are independent copies of $U$ and $\boldsymbol{X}_{\sim\mathbf{u}}$. This indicates that $Y_{\mathbf{u}}$ is correlated to $Y$ by using the same set of random variables $\boldsymbol{X}_{\mathbf{u}}$ as argument. In addition, $Y$ and $Y_{\mathbf{u}}$ follow the same distribution, and thus they share the same moments, e.g., $\mathbb{E}[Y] = \mathbb{E}[Y_{\mathbf{u}}]$, $\mathbb{E}[Y^2] = \mathbb{E}[Y_{\mathbf{u}}^2]$. Following Janon et al. [31], $S_{\mathbf{u}}$ defined in Equation (5) can be re-written as:

$$S_{\mathbf{u}} = \frac{\mathrm{Cov}[Y, Y_{\mathbf{u}}]}{\mathrm{Var}[Y]} = \frac{\mathbb{E}[Y\,Y_{\mathbf{u}}] - (\mathbb{E}[Y])^2}{\mathbb{E}[Y^2] - (\mathbb{E}[Y])^2} = \frac{\mathbb{E}[Y\,Y_{\mathbf{u}}] - \left(\frac{1}{2}\mathbb{E}[Y + Y_{\mathbf{u}}]\right)^2}{\frac{1}{2}\mathbb{E}[Y^2 + Y_{\mathbf{u}}^2] - \left(\frac{1}{2}\mathbb{E}[Y + Y_{\mathbf{u}}]\right)^2}. \tag{24}$$

We generate $N_{\mathrm{MC}}$ realizations of $Y$ and $Y_{\mathbf{u}}$ by sampling (independently) $\boldsymbol{X}_{\mathbf{u}}$, $\boldsymbol{X}_{\sim\mathbf{u}}$, $U$, $\tilde{\boldsymbol{X}}_{\sim\mathbf{u}}$, and $\tilde{U}$. The expectations in Equation (24) can be estimated by sample statistics, which leads to

$$\hat{S}_{\mathbf{u}} = \frac{\frac{1}{N_{\mathrm{MC}}}\sum_{i=1}^{N_{\mathrm{MC}}} y^{(i)} y_{\mathbf{u}}^{(i)} - \left(\frac{1}{2N_{\mathrm{MC}}}\sum_{i=1}^{N_{\mathrm{MC}}}\left(y^{(i)} + y_{\mathbf{u}}^{(i)}\right)\right)^2}{\frac{1}{2N}\sum_{i=1}^{N_{\mathrm{MC}}}\left(\left(y^{(i)}\right)^2 + \left(y_{\mathbf{u}}^{(i)}\right)^2\right) - \left(\frac{1}{2N_{\mathrm{MC}}}\sum_{i=1}^{N_{\mathrm{MC}}} y^{(i)} + y_{\mathbf{u}}^{(i)}\right)^2}, \tag{25}$$

where $y^{(i)} = Q_{\mathrm{GLaM}}\left(u^{(i)}; \boldsymbol{x}_{\mathbf{u}}, \boldsymbol{x}_{\sim\mathbf{u}}\right)$ and $y_{\mathbf{u}}^{(i)} = Q_{\mathrm{GLaM}}\left(\tilde{u}^{(i)}; \boldsymbol{x}_{\mathbf{u}}, \tilde{\boldsymbol{x}}_{\sim\mathbf{u}}\right)$ are the $i$-th realizations of $Y$ and $Y_{\mathbf{u}}$, respectively.

For QoI-based Sobol' indices defined in Equation (6), we follow the same procedure by replacing $Q_{\mathrm{GLaM}}$ by $q_{\mathrm{GLaM}}$.

### 3.5.3 PCE-based estimates

As discussed in Section 3.5.1, estimating the considered two types of Sobol' indices of a GLaM surrogate model is reduced to studying two deterministic functions $Q_{\mathrm{GLaM}}(u; \boldsymbol{x})$ and $q_{\mathrm{GLaM}}(\boldsymbol{x})$. According to the definition in Section 2.1, $\boldsymbol{X}$ has mutually independent components, which are also independent of $U \sim \mathcal{U}(0,1)$. Both functions can be represented by polynomial chaos

expansions (see Section 3.2):

$$Q_{\mathrm{GLaM}}(u; \boldsymbol{x}) \approx Q_{\mathrm{GLaM}}^{\mathrm{PC}}(u; \boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}^Q} c_{\boldsymbol{\alpha}}^Q \psi_{\boldsymbol{\alpha}}^Q(u, \boldsymbol{x}), \qquad \text{and}$$

$$q_{\mathrm{GLaM}}(\boldsymbol{x}) \approx q_{\mathrm{GLaM}}^{\mathrm{PC}}(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}^q} c_{\boldsymbol{\alpha}}^q \psi_{\boldsymbol{\alpha}}(\boldsymbol{x}),$$

(26)

where $\mathcal{A}^Q \subset \mathbb{N}^{M+1}$ and $\mathcal{A}^q \subset \mathbb{N}^M$ are the truncated sets defining the basis functions $\psi_{\boldsymbol{\alpha}}(\boldsymbol{x})$'s and $\psi_{\boldsymbol{\alpha}}^Q(u, \boldsymbol{x})$'s, respectively, as discussed in Equation (14). Note that each multi-index in $\mathcal{A}^Q$ has a dimension $M + 1$ because of the additional variable $u$, and the univariate basis functions of $u$ in $\psi_{\boldsymbol{\alpha}}^Q(u, \boldsymbol{x})$ are Legendre polynomials [24]. The advantage of using a PCE surrogate is that its Sobol' indices (of any order) can be analytically calculated by post-processing its coefficients [32].

Several methods have been developed to construct PCEs for deterministic functions with given basis functions, such as the projection method [19] and ordinary least-squares [33]. To both determine the truncated set and estimate the associated coefficients, we opt for the hybrid-LAR algorithm [29]. This method selects the most important basis functions among a candidate set, before ordinary least-squares is used to compute the coefficients. The selection procedure of the algorithm is based on *least angle regression* (LAR) [34].

Practically, we first generate $N_{\mathrm{PC}}$ samples by sampling $\boldsymbol{X}$ and $U$. They are used to evaluate the target function $Q_{\mathrm{GLaM}}(u; \boldsymbol{x})$ or $q_{\mathrm{GLaM}}(\boldsymbol{x})$ to obtain the associated model responses. Then, we apply the hybrid-LAR algorithm with the generated data to construct the PCE surrogate. Finally, the Sobol' indices are calculated by post-processing the PC coefficients.

In the following examples, we use the PCE-based estimates for the Sobol' indices of the GLaM surrogate model, instead of performing Monte Carlo simulations, as the accuracy of the former turned out to be extremely good.

# 4 Examples

In this section, we illustrate the performance of GLaMs for global sensitivity analysis on an analytical example and two case studies. We focus on the classical first-order Sobol' indices and QoI-based total Sobol' indices. The choice of the QoI depends on the focus of the example. To characterize the examples, we define the *signal-to-noise ratio* of a stochastic simulator by

$$\mathrm{SNR} = \frac{\mathrm{Var}\left[\mathbb{E}\left[Y \mid \boldsymbol{X}\right]\right]}{\mathbb{E}\left[\mathrm{Var}\left[Y \mid \boldsymbol{X}\right]\right]} = \frac{\mathrm{Var}\left[m(\boldsymbol{X})\right]}{\mathrm{Var}\left[Y\right] - \mathrm{Var}\left[m(\boldsymbol{X})\right]}.$$

(27)

This quantity gives the ratio between the variance of $Y$ explained by the mean function $m(\boldsymbol{x})$ and the remaining variance.

We use Latin hypercube sampling [35] to generate the experimental design. The stochastic simulator is evaluated only once on each combination of input parameters. The associated output values are used to construct surrogates with the proposed estimation procedure introduced in Section 3.4.

To assess the overall surrogate quality, we define the error measure

$$\varepsilon_Q \overset{\text{def}}{=} \frac{\mathbb{E}\left[\left(Q_{Y|\boldsymbol{X}}(U;\boldsymbol{X}) - Q_{\text{GLaM}}(U;\boldsymbol{X})\right)^2\right]}{\text{Var}\left[Q_{Y|\boldsymbol{X}}(U;\boldsymbol{X})\right]} = \frac{\mathbb{E}\left[\left(Q_{Y|\boldsymbol{X}}(U;\boldsymbol{X}) - Q_{\text{GLaM}}(U;\boldsymbol{X})\right)^2\right]}{\text{Var}\left[Y\right]} \quad (28)$$

where $Q_{Y|\boldsymbol{X}}(u;\boldsymbol{x})$ is the conditional quantile function of the model, $Q_{\text{GLaM}}(u;\boldsymbol{x})$ is that of the GLaM following the definition in Equation (23), and $U \sim \mathcal{U}(0,1)$. This error has a form similar to the Wasserstein distance between probability measures [36]. In addition, we also define an error measure to assess the accuracy of estimating the quantity of interest $\text{QoI}(\boldsymbol{x})$ whose approximation by GLaM is denoted by $q_{\text{GLaM}}(\boldsymbol{x})$

$$\varepsilon_q \overset{\text{def}}{=} \frac{\mathbb{E}\left[(\text{QoI}(\boldsymbol{X}) - q_{\text{GLaM}}(\boldsymbol{X}))^2\right]}{\text{Var}\left[\text{QoI}(\boldsymbol{X})\right]}. \quad (29)$$

The expectations in Equation (28) and Equation (29) are estimated by averaging the error over a test set $\mathcal{X}_{\text{test}}$ of size $10^5$.

Experimental designs of various size $N$ are investigated to study the convergence of the proposed method. For each size, 50 independent realizations of these experimental designs are carried out to account for statistical uncertainty in the random design. As a consequence, estimates for each scenario are represented by box plots.

## 4.1 A three-dimensional toy example

The first example is defined as follows:

$$Y(\boldsymbol{x},\omega) = \sin(x_1) + 7\sin^2(x_2) + \exp\left(\frac{x_1}{\pi} + x_3 Z(\omega)\right) \quad (30)$$

where $X_1, X_2 \sim \mathcal{U}(0,2\pi)$, $X_3 \sim \mathcal{U}(0.25,0.75)$ are independent input variables, and $Z \sim \mathcal{N}(0,1)$ denotes the latent variable that introduces the intrinsic randomness. The response distribution is a shifted lognormal distribution: the shift is equal to $\sin(x_1) + 7\sin^2(x_2)$ and the lognormal distribution is parameterized by $\mathcal{LN}\left(\frac{x_1}{\pi}, x_3\right)$. As a result, this stochastic simulator has a nonlinear location function and a strong heteroskedastic effect: the variance varies between 0.069 and 72.35. Besides, this example has a mild signal-to-noise ratio $\text{SNR} = 1.4$. This implies that the input variables can explain around 58% of the total variance of $Y$ (i.e., $S_{\{1,2,3\}} = 0.58$).

Figure 1 compares the PDFs predicted by a GLaM built on an experimental design of $N = 1,000$ with the reference response PDFs of the simulator. The results show that the developed algorithm correctly identifies the shape of the underlying shifted lognormal distribution. Moreover, the PDF supports and tails are also accurately approximated.

We consider the differential entropy $h(\boldsymbol{x})$ [12] as the QoI in this example. Because the analytical response distribution and entropy are known, we investigate the convergence of GLaM in terms of the conditional quantile function estimation Equation (29) and the entropy estimation
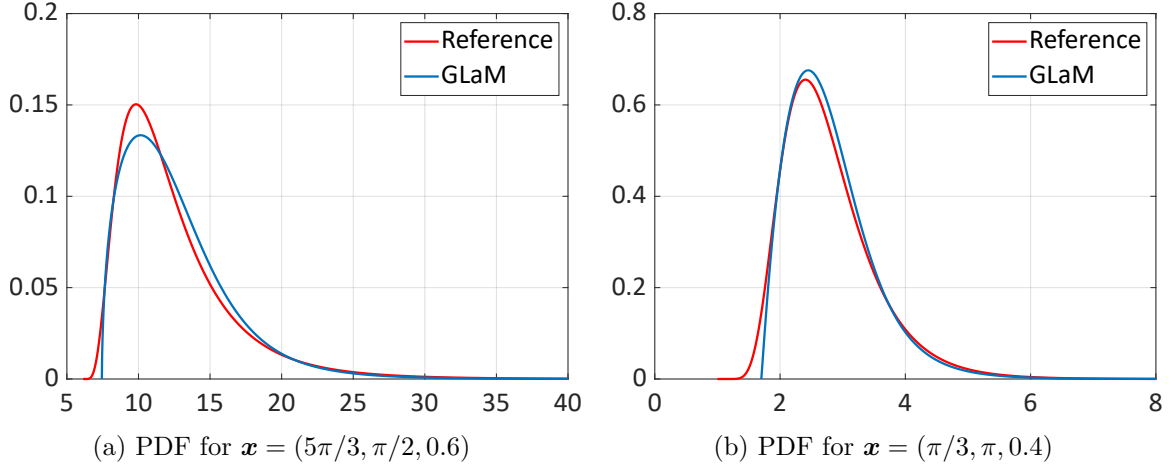
(a) PDF for $\boldsymbol{x} = (5\pi/3, \pi/2, 0.6)$        (b) PDF for $\boldsymbol{x} = (\pi/3, \pi, 0.4)$

Figure 1: Toy example – Emulated response PDFs, $N = 1{,}000$

.

Equation (28). The size of experimental design varies among $N \in \{250; 500; 1{,}000; 2{,}000; 4{,}000\}$. Note that the entropy of a GLD does not have a closed form. Therefore, we use $10^4$ Monte Carlo samples to estimate this quantity of a GLaM for each $\boldsymbol{x}$ in the test set.
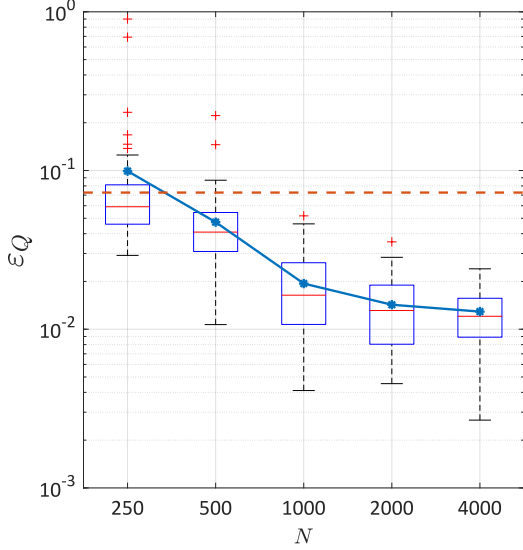
In addition, we consider another model where we approximate the response distribution with a normal distribution. The mean and variance (as functions of $\boldsymbol{x}$) for such an approximation are chosen as the true mean and variance of the original. In other words, this model represents the "oracle" of Gaussian-type mean-variance models.

The results are summarized in Figure 2. The proposed method exhibits a clear convergence with respect to $N$ for both $Q(u; \boldsymbol{x})$ and $h(\boldsymbol{x})$ estimations. We observe in Figure 2a that the decay of $\varepsilon_Q$ has two regimes separated by $N = 1{,}000$. For a small $N$, the error coming from the use of finite samples dominates the estimate accuracy. When we consider a large data set, the error mainly comes from the model mispecification, because the stochastic simulator cannot be exactly represented by a GLaM. This phenomenon is not significant for the entropy estimation, which demonstrates a relatively consistent decay.
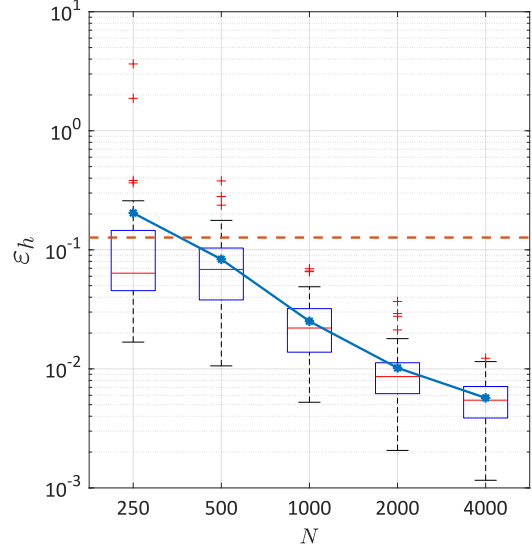
The accuracy of the oracle normal approximation is reported with red dash lines in Figure 2. The error shown is only due to model misspecifications (because the true response distribution is not Gaussian) since we use the underlying true mean and variance. For both measures, the medians of the errors of GLaMs built on $N = 250$ model runs are smaller than those of the normal approximation. For $N \geq 1{,}000$, the GLaM clearly outperforms the oracle of Gaussian-type mean-variance models. This example illustrates the limits of such Gaussian-type models in practice.

Finally, the errors of GLaMs are below $0.05$ for $N \geq 1{,}000$ indicating that the surrogate is able to explain over 95% of the variance of the target functions.

For sensitivity analysis, we focus on the classical first-order and the entropy-based total Sobol' indices. Figure 3 and Figure 4 show the convergence of GLaMs for estimating these quantities

16

(a) Conditional quantile function estimation

(b) Entropy estimation

Figure 2: Toy example – Convergence study. The blue lines denote the errors averaged over 50 repetitions of the full analysis. The red dash lines are the corresponding errors of the model assuming that the response distribution is normal with the true mean and variance

of each input variable. The reference values are derived from Equation (30). As shown by the two figures, this toy example is designed to have $X_2$ as the most important variable according to the classical first-order Sobol' indices, which also indicates that $X_2$ contributes the most to the variance of the mean function $m(\boldsymbol{X})$. In contrast, it has zero effect to the entropy. In comparison, $X_1$ is the dominant variable for the variation of the entropy $h(\boldsymbol{X})$. Because $X_3$ mainly controls the shape of the response distribution (especially the right tail), it has a minor first-order effect to the mean function, which leads to a very small value of $S_3$. In contrast, the entropy depends on the distribution shape, and thus $S_{T_3}^h$ is not negligible. The results reveal that GLaMs capture this characteristic and yield accurate estimates for both classical Sobol' indices and entropy-based Sobol' indices.

Similar to Figure 2, we also reported the sensitivity indices calculated by using Gaussian approximations with the true mean and variance. Because the classical first-order indices depend only on the mean and variance functions, the oracle Gaussian model gives the exact values. Therefore, we showed only the results for the entropy-based Sobol' indices in Figure 4. It is clear that the Gaussian approximation with the true mean and variance demonstrates a significant bias. In contrast, GLaMs show nearly no bias and approximate much more accurately the reference values.

## 4.2 Heston model

In this example, we perform the global sensitivity analysis for a Heston model used in mathematical finance [37]. The Heston model describes the evolution of a stock price $Y_t$. It is an extension of
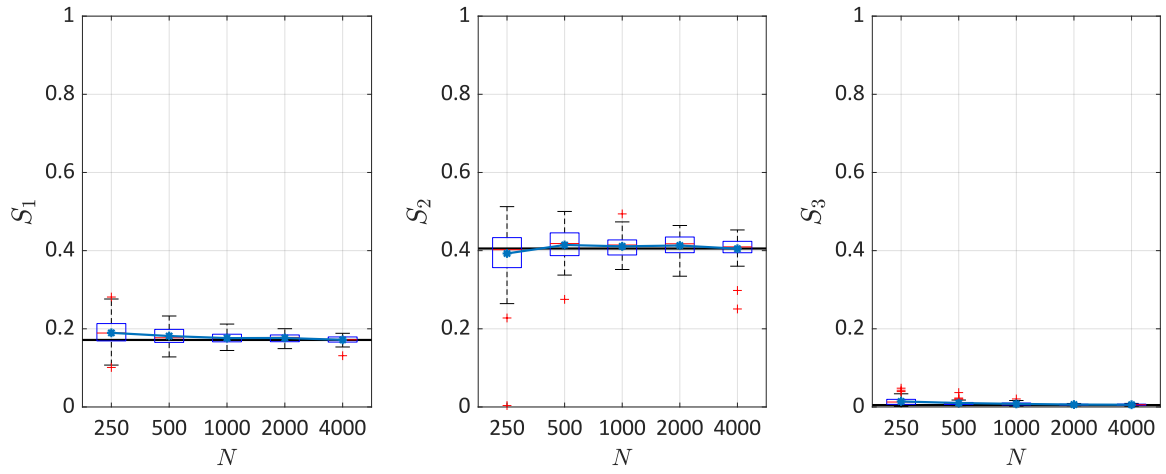
Figure 3: Estimation of the classical first-order Sobol' indices. The black lines are the reference values, and the blue lines denote the average values of 50 repetitions of the full analysis.
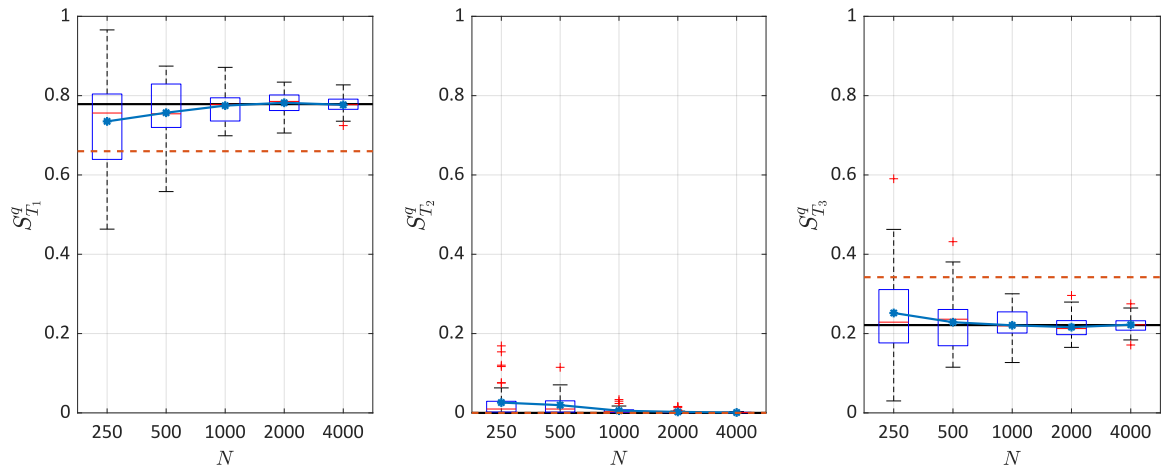


Figure 4: Estimation of the entropy-based total Sobol' indices. The black lines are the reference values, and the blue lines denote the average values of 50 repetitions of the full analysis. The red dash lines correspond to the indices calculated from the normal approximation using the true mean and variance

the geometric Brownian motion by modeling the volatility as a stochastic process $v_t$, instead of considering it as constant. Hence, the Heston model is a stochastic volatility model and consists of two coupled stochastic differential equations:

$$\begin{aligned}
\mathrm{d}Y_t &= \mu Y_t \mathrm{d}t + \sqrt{v_t} Y_t \mathrm{d}W_t^1, \\
\mathrm{d}v_t &= \kappa(\theta - v_t)\mathrm{d}t + \sigma\sqrt{v_t}\mathrm{d}W_t^2,
\end{aligned} \tag{31}$$

with

$$\mathbb{E}\left[\mathrm{d}W_t^1 \, \mathrm{d}W_t^2\right] = \rho\mathrm{d}t, \tag{32}$$

where $W_t^1$ and $W_t^2$ are two Wiener processes with correlation coefficient $\rho$, which introduce the intrinsic randomness of the stochastic model. The model parameters $\boldsymbol{x} = (\mu, \kappa, \theta, \sigma, \rho, v_0)$ are summarized in Table 1. The range of the last five input variables are selected based on the parameters calibrated from real data (S&P 500 and Eurostoxx 50) [38]. The range of the first variable $\mu$ is set to $[0, 0.1]$ to take the uncertainty of the expected return rate into account. Without loss of generality, we set $Y_0 = 1$.

Table 1: Parameters of the Heston model

| Variable | Description | Distribution |
|----------|-------------|--------------|
| $\mu$ | Expected return rate | $\mathcal{U}(0, 0.1)$ |
| $\kappa$ | Mean reversion speed of the volatility | $\mathcal{U}(0.3, 2)$ |
| $\theta$ | Long term mean of the volatility | $\mathcal{U}(0.02, 0.07)$ |
| $\sigma$ | Volatility of the volatility | $\mathcal{U}(0.2, 0.4)$ |
| $\rho$ | Correlation coefficient between $\mathrm{d}W_t^1$ and $\mathrm{d}W_t^2$ | $\mathcal{U}(-1, -0.5)$ |
| $v_0$ | Volatility at time 0 | $\mathcal{U}(0.02, 0.07)$ |

In this example, we are interested in the stock price after one year, i.e., $Y_t(\boldsymbol{x})$ with $t = 1$. A closed form solution to Equation (31) is generally not available. To get samples of $Y_1(\boldsymbol{x})$, we simulate the entire time evolution of $Y_t$ and $v_t$ for a given $\boldsymbol{x}$ using Euler integration scheme with $\Delta t = 0.001$ over the time interval $[0, 1]$. Note that when simulating the bivariate process $(Y_t, v_t)$, a problem may happen: since $v_t$ follows a Cox–Ingersoll–Ross process [38], the simulation scheme can generate negative values for $v_t$. To overcome the problem, we apply the full truncation scheme, which replaces the update of $v_t$ by $\max(v_t, 0)$ [38].

Figure 5 shows two response PDFs predicted by a surrogate built upon $N = 2{,}000$ model runs. The reference histograms are obtained from $10^4$ repeated model runs with the same input parameters. We observe that the variance of the response distribution is not constant, e.g., 0.065 and 0.027 for the two illustrated PDFs. Moreover, the PDF shape varies: it changes from symmetric to left-skewed distributions depending on the model parameters. This would be difficult to approximate with a simple distribution family such as normal or lognormal. In contrast, GLaMs are able to accurately capture this shape variation, because of the flexibility of generalized lambda distributions.
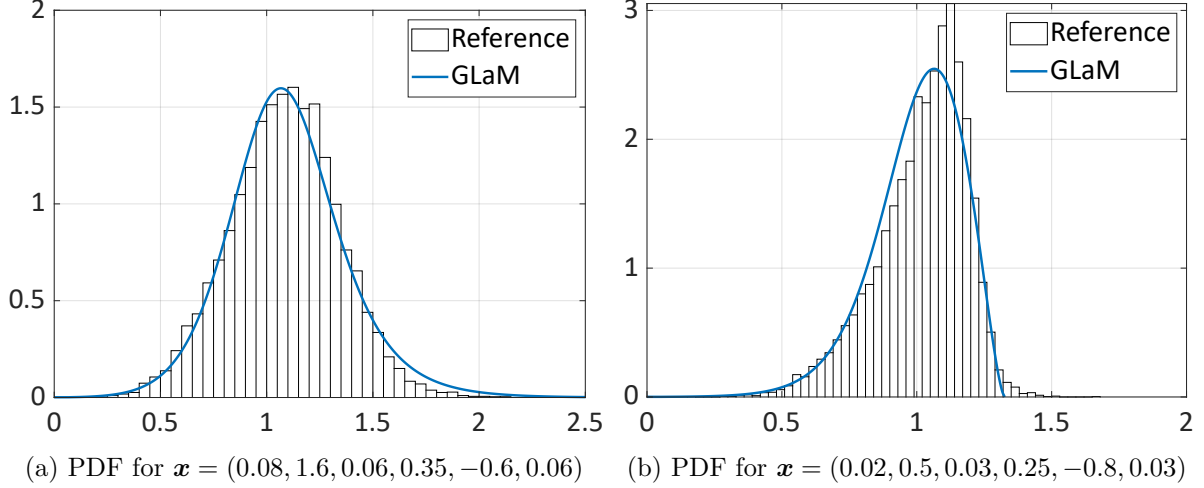
(a) PDF for $\boldsymbol{x} = (0.08, 1.6, 0.06, 0.35, -0.6, 0.06)$     (b) PDF for $\boldsymbol{x} = (0.02, 0.5, 0.03, 0.25, -0.8, 0.03)$

Figure 5: Heston model – Emulated response PDFs, $N = 2,000$

Even though a closed form distribution of $Y_1(\boldsymbol{x})$ does not exist, the mean function $m(\boldsymbol{x}) = \mathbb{E}[Y_1(\boldsymbol{x})]$ can be derived analytically:

$$m(\boldsymbol{x}) = \exp(x_1) = \exp(\mu). \tag{33}$$

As a result, we use $\varepsilon_m$ defined in Equation (29) with $\mathrm{QoI}(\boldsymbol{x}) = m(\boldsymbol{x})$ to assess the convergence of the surrogate. In addition, we also consider the expected payoff of an European call option. The payoff $C(\boldsymbol{x})$ and the expected payoff $m_C(\boldsymbol{x})$ of an European call option are defined by

$$
\begin{aligned}
C(\boldsymbol{x}) &= \max\{0, Y_1(\boldsymbol{x}) - K\}, \\
m_C(\boldsymbol{x}) &= \mathbb{E}[C(\boldsymbol{x})],
\end{aligned} \tag{34}
$$

where $K$ is the *strike price* and set to 1 in the following analysis. In finance, $m_C$ not only is important for making investment decisions but also has a very similar form to the option price [39]. For the Heston model, numerical methods based on the Fourier transform have been developed to calculate the expected payoff without the need for Monte Carlo simulations [37]. For the GLaM surrogate, this quantity can also be calculated numerically (see Section 6.1). As a second performance index, we compute the associated error denoted by $\varepsilon_C$ (Equation (29)) for the convergence study.

Figure 6 shows box plots of the errors $\varepsilon_m$ and $\varepsilon_c$ for $N \in \{500; 1,000; 2,000; 4,000; 8,000; 16,000\}$. Both $\varepsilon_m$ and $\varepsilon_c$ are relatively large for $N \leq 2,000$. This is mainly due to the fact that the variability of the model response is dominated by the intrinsic randomness: the model parameters $\boldsymbol{X}$ altogether are only able to explain about 2% of the variance of the output (i.e., $S_{\{1,\ldots,6\}} = 0.02$). In other words, the stochastic simulator has a very small signal-to-noise ratio $\mathrm{SNR} = 0.02/(1 - 0.02) \approx 0.02$. Since GLDs are flexible, a few data scattered in a moderately high dimensional space may not provide enough information of the response distribution variation. We observe that for $N \leq 1,000$, the selection procedure proposed in Algorithm 1 can choose $\lambda_1^{\mathrm{PC}}(\boldsymbol{x})$ and $\lambda_2^{\mathrm{PC}}(\boldsymbol{x})$ being only constant. Such a model is too simple and thus fails to capture the

20

variations of the scalar quantities. Consequently, it is necessary to have enough data to achieve an accurate estimate: when increasing the size of $N$ of the LHS design, we observe a clear decay of the errors.



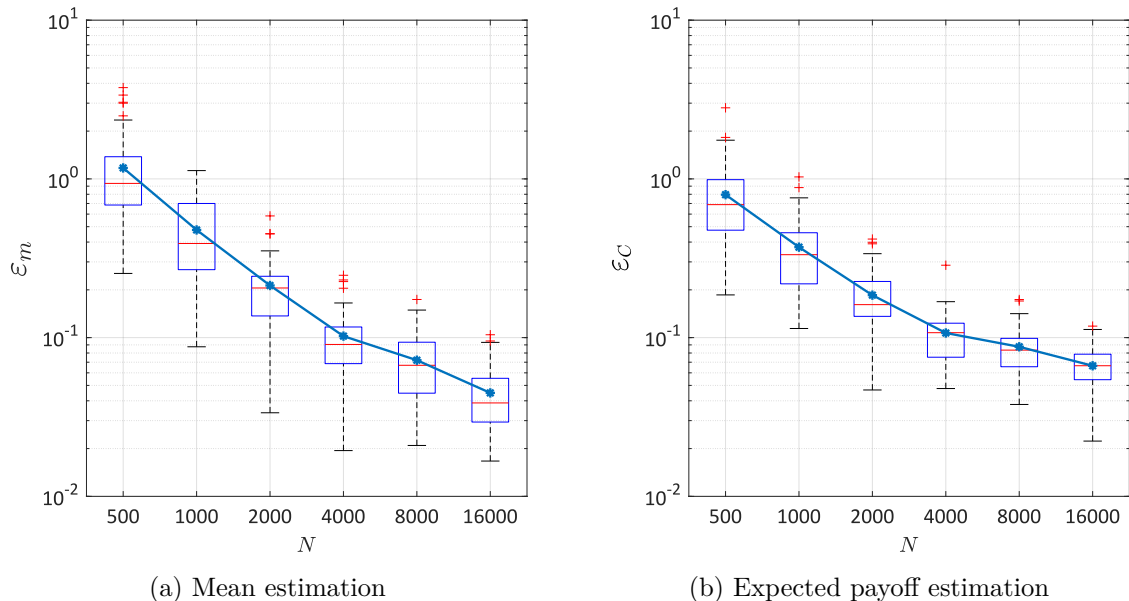(a) Mean estimation

(b) Expected payoff estimation

Figure 6: Heston model – Convergence study. The blue lines denote the errors averaged over 50 repetitions of the full analysis.

We now study the convergence for the Sobol' indices estimations. According to Equation (33), the mean function depends only on the first input variable $X_1$, which contributes little (2%) to the total variance of $Y_1(\boldsymbol{X})$. This implies that the classical Sobol' indices are not informative (they are either 0 or very close to 0). However, we cannot ignore the variability of the input variables because the response distribution demonstrates a clear dependence on the input parameters, as shown in Figure 5. Therefore, we focus on the accuracy of the expected-payoff-based total Sobol' indices, denoted by $S_{T_{\mathbf{u}}}^C$.

As a second quantity of interest, we also calculate the total Sobol' indices associated to the 95%-superquantile, referred to as $S_{T_{\mathbf{u}}}^{\mathrm{sq}}$. Superquantiles are known as the *conditional value-at-risk*, which is an important risk measure in finance [40]. The $\alpha$-superquantile of a random variable $Y$ is defined by

$$\mathrm{sq}_\alpha = \mathbb{E}\left[Y \mid Y \geq q_\alpha\right], \tag{35}$$

where $q_\alpha$ is the $\alpha$-quantile of $Y$. This quantity corresponds to the conditional expectation of $Y$ being larger than its $\alpha$-quantile. For the Heston model, this quantity does not have an analytical closed form, whereas $\mathrm{sq}_\alpha$ of a GLD can be derived analytically (see Section 6.1).

We use $10^5$ Monte Carlo samples to evaluate (numerically) the function $m_C(\boldsymbol{x})$ to obtain a reference value for each $S_{T_{\mathbf{u}}}^C$. To calculate the Sobol' indices associated to the 95%-superquantile $\mathrm{sq}_{95}(\boldsymbol{X})$, it is necessary to evaluate the function $\mathrm{sq}_{95}(\boldsymbol{x})$. Because it cannot be analytically derived for the Heston model, we use $10^4$ replications to calculate the sample 95%-superquantile

$\hat{\mathrm{sq}}_{95}(\boldsymbol{x})$ as an estimate for $\mathrm{sq}_{95}(\boldsymbol{x})$. Then, we treat it as a deterministic function and use $10^4$ samples to estimate each Sobol' index. This indicates that a total number of $7 \times 10^8$ model runs are performed to obtain the six reference 95%-superquantile-based total Sobol' indices. Because only $10^4$ samples are used to estimate each $S_{T_\mathbf{u}}^{\mathrm{sq}}$, we use bootstraps [41] to calculate the 95% confidence interval to account for the uncertainty of the Monte Carlo simulation.



Figure 7: Estimation of the expected-payoff-based total Sobol' indices. The black lines are the reference values, and the blue lines denote the average values of 50 repetitions.

Figures 7 and 8 confirm and quantify the convergence of GLaMs to estimate $S_{T_\mathbf{u}}^C$ and $S_{T_\mathbf{u}}^{\mathrm{sq}}$. For the expected payoff $m_C(\boldsymbol{x})$, the first variable $\mu$ is the most important. The estimation of its total effect converges from below the reference value, and we observe a bias in the estimate. Nevertheless, with $N$ large enough ($\geq 4{,}000$), the GLaM can always correctly identify its importance (the bias is 0.072 for $\geq 8{,}000$ and 0.055 for $\geq 16{,}000$), and each classical first-order Sobol' index of the other five variables converge to the reference line. The 95%-superquantile suggests a different ranking: $\mu$, $\theta$, $\rho$ and $v_0$ (corresponding to the first, third, fifth and sixth input variable, respectively) have similar total effects, which are superior to those of $\kappa$ and $\sigma$ (i.e., the second and fourth input variables). In addition, none of the input variables has nearly 0 total effect. The GLaM surrogate model accurately reproduces the phenomena. Moreover, the estimates generally vary around the reference values, and larger $N$ results in narrower spread of the box plots.

As a conclusion, GLaM surrogates allow us to represent accurately the QoI of the Heston model and carry out a detailed sensitivity analysis at the cost of $\mathcal{O}(10^4)$ runs of the stochastic simulator. Note that in this example the leave-one-out errors of the polynomial chaos expansions built on the GLaM surrogates are of the order of $o(10^{-4})$, which justifies the use of PCE-based Sobol' indices.
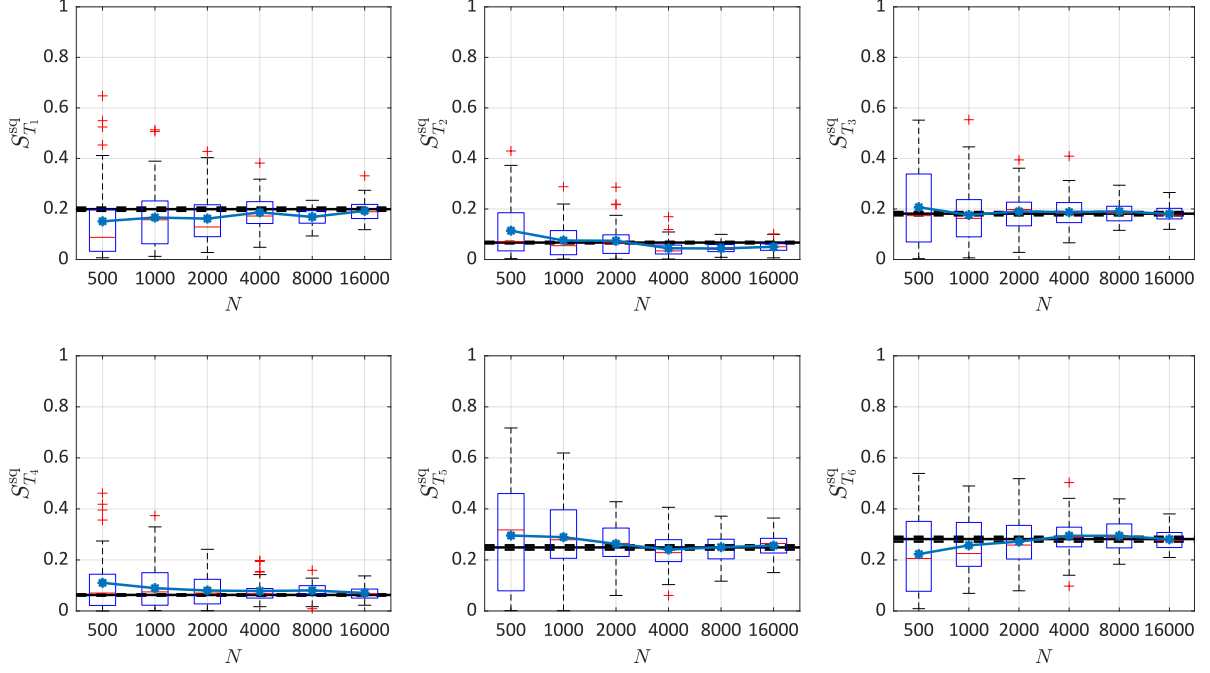
Figure 8: Estimation of the 95%-superquantile-based total Sobol' indices. The blue lines denote the average value of 50 repetitions of the full analysis. The black lines are the reference values, and the dashed lines correspond to the 95% confidence intervals.

## 4.3 Stochastic SIR model

In this example, we apply the proposed method to a *stochastic Susceptible-Infected-Recovered* (SIR) model in epidemiology [3]. This model simulates the spread of an infectious disease, which can help conduct appropriate epidemiological intervention to minimize the social and ethical impacts during the outbreak.

In a SIR model, a population of size $P_t$ at time $t$ can be partitioned into three groups: susceptible, infected and recovered during the outbreak of an epidemic. Susceptible individuals are those who can get infected by contacting an infectious person. Infected individuals are suffering from the disease and are contagious. They can recover (therefore classified as recovered) and become immune to future infections. The number of individuals within each group is denoted by $E_t$, $I_t$ and $R_t$, respectively. Without differentiating individuals, these three quantities characterize the configuration of the population at a given time $t$. Hence, their evolutions represent the spread of the epidemic. In this study, we consider a fixed population without newborns and deaths, i.e., the total population size is constant, $P_t = P$. As a result, $E_t$, $I_t$ and $R_t$ satisfy the constraint $E_t + I_t + R_t = P$, and thus only the time evolution of $E_t$ and $I_t$ is necessary to represent the disease evolution.

Without going into detailed assumptions of the model, we illustrate the system dynamics in Figure 9, where the black icons represent susceptible individuals, the red icons indicate infected persons, and the blue icons are those recovered. Suppose that at time $t$ the population has the
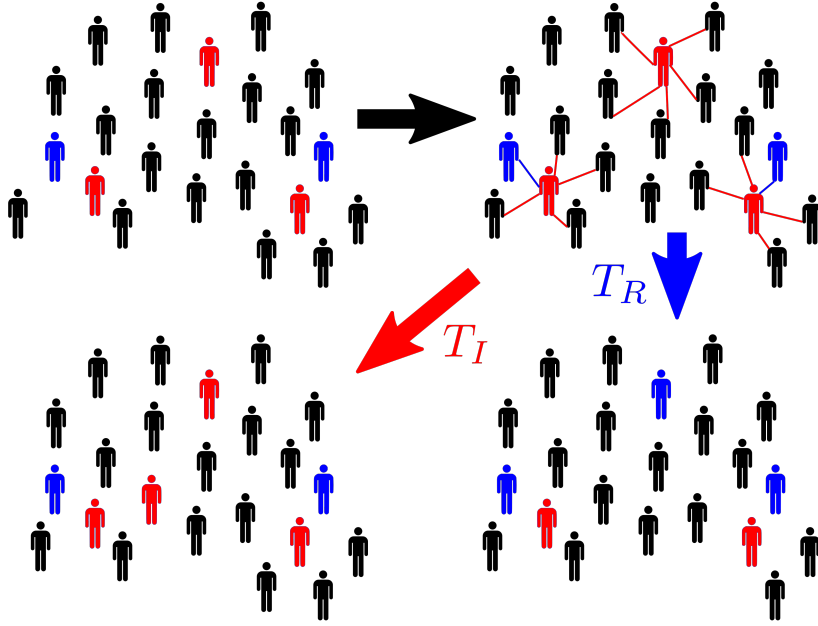
Figure 9: Dynamics of the stochastic SIR model: black icons denote susceptible individuals, red icons represent infected individuals, and blue icons are those recovered.

configuration $(E_t, I_t)$ (top left figure of Figure 9). Infected individuals can meet susceptible individuals, or they may receive essential treatments and recover from the disease. Hence, the next configuration has two possibilities: (1) $C_I$, where one susceptible individual is infected; (2) $C_R$, where one infected person recovers. The population state evolving either to $C_I$ or $C_R$ depends on two random variables, $T_I$ and $T_R$, which denote the respective time to move to the associated candidate configuration. Both random variables follow an exponential distribution, yet with different parameters:

$$T_I \sim \text{Exp}(\lambda_I), \quad \lambda_I = \beta \frac{E_t I_t}{P}, \tag{36}$$

$$T_R \sim \text{Exp}(\lambda_R), \quad \lambda_R = \gamma I_t, \tag{37}$$

where $\beta$ indicates the contact rate of an infected individual, and $\gamma$ is the recovery rate. If $T_I > T_R$, $C_I$ becomes the next configuration at $t + T_I$ with $S_{t+T_I} = E_t - 1$ and $I_{t+T_I} = I_t + 1$, and vice versa. This update step iterates until time $T$ when $I_T = 0$. Because the population size is finite and the recovered individuals will not get infected again, the total number of updates is finite ($\leq P$). This number is not a constant due to the updating process, indicating that the amount of latent variables of this simulator is also random. Note that the evolution procedure described here corresponds to the *Gillespie algorithm* [42].

In this case study, we set $P = 2,000$. $\boldsymbol{x} = (E_0, I_0, \beta, \gamma)$ is the vector of input parameters. To account for different scenarios, the input variables $\boldsymbol{X}$ are modeled as $X_1 \sim \mathcal{U}(1,600\,,1,800)$, $X_2 \sim \mathcal{U}(20, 200)$ and $X_3, X_4 \sim \mathcal{U}(0.5, 0.7)$. The uncertainty in the first two variables can be interpreted as lack of knowledge of the initial condition. While the last two variables are affected by possible interventions, such as social distancing measures that can reduce the contact rate $\beta$

24

and increasing medical resources that improves the recovery rate $\gamma$. We are interested in the total number of newly infected individuals during the outbreak, i.e., $E_T - E_0$. The signal-to-noise ratio of this stochastic model is estimated to be SNR $\approx 6.7$, which is relatively large.

Figure 10 shows the response PDF estimation of the surrogate model built on an experimental design of $N = 1,000$. The reference histograms are calculated from $10^4$ repeated model runs with the same input values. We observe that the response distribution changes from right-skewed to left-skewed distributions (so we can also find symmetric distributions in between), which is correctly represented by the surrogate. In addition, GLaMs also accurately approximate the bulk and the support of the response PDF.



(a) PDF for $\boldsymbol{x} = (1750, 150, 0.55, 0.65)$  (b) PDF for $\boldsymbol{x} = (1650, 80, 0.65, 0.55)$
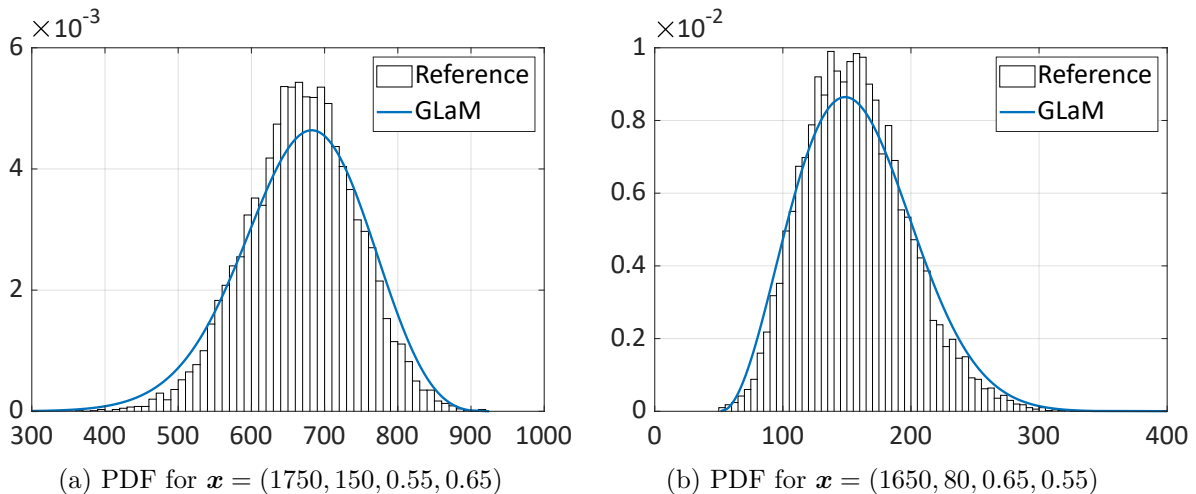
Figure 10: Stochastic SIR model – Emulated response PDFs, $N = 1,000$

In this example, we investigate the convergence of GLaMs for estimating the classical first-order Sobol' indices and the standard-deviation-based total Sobol' indices, denoted by $S^\sigma_{T_{\mathbf{u}}}$. To calculate the reference values, we use $10^5$ Monte Carlo samples for each classical Sobol' index. Regarding the standard-deviation-based Sobol' indices, we calculate the sample standard deviation $\hat{\sigma}(\boldsymbol{x})$ based on $10^4$ replications. Then, we apply Monte Carlo simulations with $10^4$ samples to estimate the associated Sobol' indices. The total cost to get reference values is thus equal to $5 \times 10^8$. As in the previous example, we use bootstraps to calculate the 95% confidence intervals.

Figures 11 and 12 show the results of the convergence study. In terms of the classical Sobol' indices, the GLaM yields accurate estimates even when $N = 250$: the box plots scatter around the reference values with a small variability. Among the four input variables, the second one $I_0$ that corresponds to the number of infected individuals at time 0 is the most important. It is followed by the contact rate and the recovery rate, which show similar first-order effect. As a result, performing medical test to better determine $I_0$ would be the most effective way to reduce the variance of the output. In contrast, Figure 12 suggests that controlling the contact rate and recovery rate would be the best measure to reduce the variation of $\sigma(\boldsymbol{X})$. For estimating the associated Sobol' indices, the GLaM converges within the 95% confidence intervals of the Monte
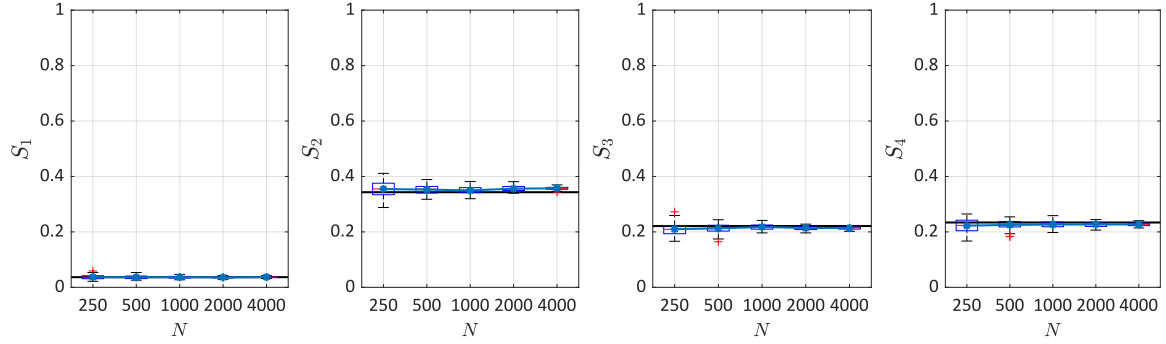
Figure 11: Estimation of the classical first-order Sobol' indices. The black lines are the reference values, and the blue lines denote the average values of 50 repetitions of the full analysis.
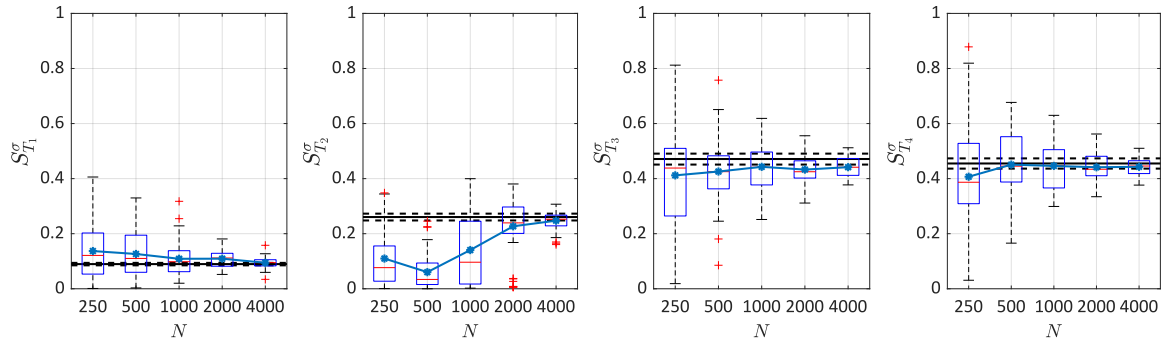


Figure 12: Estimation of the standard-deviation-based total Sobol' indices. The blue lines denote the average values of 50 repetitions of the full analysis. The black lines are the reference values, and the dashed lines correspond to the 95% confidence intervals.

Carlo estimates, and the spread of the box plots decreases significantly with $N$ increasing.

Finally, we remark that when higher-order Sobol' indices are of interest, Monte Carlo simulations require additional runs of the original model. For example, $4 \times 10^8$ more model evaluations should be performed to obtain the reference values for the standard-deviation-based second-order Sobol' indices. This results in a large amount of total model runs, which become impracticable even with cheap models. In contrast, GLaM surrogates can be used without additional cost: the PCE-based method presented in Section 3.5.3 provides analytical higher-order indices by post-processing the PC coefficients. In this example, the leave-one-out errors of PCE built on the GLaM surrogates are of the order $\mathcal{O}(10^{-3})$, which justifies the use of PCE estimates. Based on the surrogate model, we observe that the largest standard-deviation-based second-order Sobol' indices, which translate parameters interactions, are $I_0$ and $\beta$, $I_0$ and $\gamma$. Both have a value 0.09, while the other second-order interactions are very small. As illustrated in Figure 12, $I_0$ has a total effect $S^{\sigma}_{T_2} = 0.26$. Moreover, it has a relatively small first-order effect $S^{\sigma}_2 = 0.05$. This implies that $I_0$ mainly affects the variance of $\sigma(\boldsymbol{X})$ through its interactions with $\beta$ and $\gamma$.

## 5 Conclusions

In this paper, we discuss the nature and focus of three extensions of Sobol' indices to stochastic simulators: classical Sobol' indices, QoI-based Sobol' indices and trajectory-based Sobol' indices. The first two types are of interest because of their versatility and applicability to a broad class of problems. We propose to use the generalized lambda model as a stochastic emulator to estimate the considered indices. This surrogate model aims at emulating the entire response distribution, instead of focusing only on some scalar statistical quantities, e.g., mean and variance. More precisely, it relies on using the four-parameter generalized lambda distribution to approximate the response distribution. The associated distribution parameters as functions of the input are represented by polynomial chaos expansions. Such a surrogate can be constructed without the need for replications, and thus it is not restricted to a special data structure.

Because of the special formulation of GLaM, the considered sensitivity indices can be estimated by directly working with deterministic functions. This allows applying the methods developed for deterministic simulators, namely estimators based on Monte Carlo simulations and on polynomial chaos expansions. In this paper, we suggest the latter to post-process the surrogate model to achieve high computational efficiency.

The performance of the proposed method for estimating various Sobol' indices is illustrated on three examples with different signal-to-noise ratios. The toy example is designed to have a strong heteroskedastic effect. It shows the general convergent behavior of GLaMs for approximating the conditional quantile functions and estimating the entropy of the response distributions. The second example is a Heston model from mathematical Finance. This case study has a very small signal-to-noise ratio and demonstrates a shape variation of the response PDF. The surrogate

27

generally yields accurate estimate of the Sobol' indices associated to the expected payoff and the 95%-superquantile. The last example is a stochastic SIR model in epidemiology, in which GLaMs exhibit robust estimates of the classical Sobol' indices and the standard-deviation-based Sobol' indices. All three examples have a different ranking of the input variables depending on the type of Sobol' indices, which is correctly captured by GLaMs when comparing to reference values obtained by extremely costly Monte Carlo simulations. Fairly accurate results are obtained at a cost of $\mathcal{O}\left(10^4\right)$ runs of the simulator compared to reference values based on $\mathcal{O}\left(10^8\right)$ runs by a brute force approach.

In future work, we plan to develop algorithms to improve GLaMs for small data sets. Besides, we will investigate GLaMs for estimating distribution-based sensitivity indices [43, 44]. The estimation of these indices usually requires a large number of model runs to infer the conditional PDF, which can be easily obtained from GLaMs. In addition, appropriate contrast measures between distributions, such as the Wasserstein metric, can be developed for sensitivity analysis in the context of stochastic simulators. Finally, developing sensitivity indices for stochastic simulators with dependent input variables will allow engineers to tackle a broader group of problems.

# Acknowledgements

# 6   Appendix

## 6.1   Some properties of GLDs

The mean and variance of a GLD can be calculated by

$$m = \lambda_1 - \frac{1}{\lambda_2}\left(\frac{1}{\lambda_3+1} - \frac{1}{\lambda_4(\boldsymbol{x})+1}\right), \tag{38}$$

$$v = \frac{(d_2 - d_1^2)}{\lambda_2^2}, \tag{39}$$

where $\{d_k : k = 1, 2\}$ are defined by

$$
\begin{aligned}
d_1 &= \frac{1}{\lambda_3}\,\mathrm{B}(\lambda_3+1, 1) - \frac{1}{\lambda_4}\,\mathrm{B}(1, \lambda_4+1),\\
d_2 &= \frac{1}{\lambda_3^2}\,\mathrm{B}(2\lambda_3+1, 1) - \frac{2}{\lambda_3\lambda_4}\,\mathrm{B}(\lambda_3+1, \lambda_4+1) + \frac{1}{\lambda_4^2}\,\mathrm{B}(1, 2\lambda_4+1),
\end{aligned}
\tag{40}
$$

with B denoting the beta function.

The expected payoff defined in Equation (41) of a GLD with the strike price $K$ is given by

$$m_C \overset{\text{def}}{=} \mathbb{E}\left[\max\{Y - K, 0\}\right]$$
$$= \left(\lambda_1 - \frac{1}{\lambda_2 \lambda_3} + \frac{1}{\lambda_2 \lambda_4} - K\right)(1 - u_K) + \frac{1}{\lambda_2}\left(\frac{1 - u_K^{\lambda_3 + 1}}{\lambda_3(\lambda_3 + 1)} - \frac{(1 - u_K)^{\lambda_4 + 1}}{\lambda_4(\lambda_4 + 1)}\right). \tag{41}$$

where $u_K$ is the solution of the nonlinear equation:

$$Q(u_K; \boldsymbol{\lambda}) = K. \tag{42}$$

The $\alpha$-superquantile $\text{sq}_\alpha$ defined in Equation (35) of a GLD has a closed-form:

$$\text{sq}_\alpha \overset{\text{def}}{=} \mathbb{E}\left[Y \mid Y > q_\alpha\right]$$
$$= \lambda_1 - \frac{1}{\lambda_2 \lambda_3} + \frac{1}{\lambda_2 \lambda_4} + \frac{1}{(1 - \alpha)\lambda_2}\left(\frac{1 - \alpha^{\lambda_3 + 1}}{\lambda_3(\lambda_3 + 1)} - \frac{\alpha^{\lambda_4 + 1}}{\lambda_4(\lambda_4 + 1)}\right). \tag{43}$$

# References

[1] S. Azzi, Y Huang, B. Sudret, and J. Wiart. Surrogate modeling of stochastic functions – application to computational electromagnetic dosimetry. *Int. J. Uncertainty Quantification*, 9(4):351–363, 2019.

[2] A. J. McNeil, R. Frey, and P. Embrechts. *Quantitative Risk Management: Concepts, Techniques, and Tools*. Princeton Series in Finance. Princeton University Press, Princeton, New Jersey, 2005.

[3] T. Britton. Stochastic epidemic models: a survey. *Math. Biosci.*, 225:24–35, 2010.

[4] A. Saltelli, K. Chan, and E. M. Scott, editors. *Sensitivity analysis*. J. Wiley & Sons, 2000.

[5] E. de Rocquigny, N. Devictor, and S. Tatantola. *Uncertainty in Industrial Practice: A guide to Quantitative Uncertainty Management*. J. Wiley & Sons, New York, 2008.

[6] J.C. Helton, J.D. Johnson, C.J. Sallaberry, and C.B. Storlie. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliab. Eng. Sys. Safety*, 91(10–11):1175–1209, 2006.

[7] E. Borgonovo. *Sensitivity analysis – An Introduction for the Management Scientist*. Springer, 2017.

[8] I. M. Sobol'. Sensitivity estimates for nonlinear mathematical models. *Math. Modeling & Comp. Exp.*, 1:407–414, 1993.

[9] B. Iooss and M. Ribatet. Global sensitivity analysis of computer models with functional inputs. *Reliab. Eng. Sys. Safety*, 94:1194–1204, 2009.

[10] J.L. Hart, A. Alexanderian, and P.A. Gremaud. Efficient computation of Sobol' indices for stochastic models. *SIAM J. Sci. Comput.*, 39:1514–1530, 2016.

[11] M. N. Jimenez, O. P. Le Maître, and O. M. Knio. Nonintrusive polynomial chaos expansions for sensitivity analysis in stochastic differential equations. *SIAM J. Uncer. Quant.*, 5:212–239, 2017.

[12] S. Azzi, B. Sudret, and J. Wiart. Sensitivity analysis for stochastic simulators using differential entropy. *Int. J. Uncertainty Quantification*, 10:25–33, 2020.

[13] M. Dacidian and R. J. Carroll. Variance function estimation. *J. Amer. Stat. Assoc.*, 400:1079–1091, 1987.

[14] A. Marrel, B. Iooss, S. Da Veiga, and M. Ribatet. Global sensitivity analysis of stochastic computer models with joint metamodels. *Stat. Comput.*, 22:833–847, 2012.

[15] M. Binois, R. Gramacy, and M. Ludkovski. Practical heteroskedastic Gaussian process modeling for large simulation experiments. *J. Comput. Graph. Stat.*, 27:808–821, 2018.

[16] X. Zhu and B. Sudret. Replication-based emulation of the response distribution of stochastic simulators using generalized lambda distributions. *Int. J. Uncertainty Quantification*, 10(3):249–275, 2020.

[17] X. Zhu and B. Sudret. Emulation of stochastic simulators using generalized lambda models. *Submitted to SIAM/ASA J. Unc. Quant.*, 2021. (ArXiv: 2007.00996).

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

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

[20] T. Homma and A. Saltelli. Importance measures in global sensitivity analysis of non linear models. *Reliab. Eng. Sys. Safety*, 52:1–17, 1996.

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

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

[23] O. G. Ernst, A. Mugler, H. J. Starkloff, and E. Ullmann. On the convergence of generalized polynomial chaos expansions. *ESAIM: Math. Model. and Num. Anal.*, 46:317–339, 2012.

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

[25] B. Sudret. *Polynomial chaos expansions and stochastic finite element methods*, chapter 6, pages 265–300. Risk and Reliability in Geotechnical Engineering. Taylor and Francis, 2015.

[26] G. Blatman and B. Sudret. An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. *Prob. Eng. Mech.*, 25:183–197, 2010.

[27] J.M. Wooldridge. *Introductory Econometrics: A Modern Approach*. Cengage Learning, 5th edition, 2013.

[28] S. Marelli and B. Sudret. UQLab user manual – Polynomial chaos expansions. Technical report, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, 2019. Report # UQLab-V1.3-104.

[29] G. Blatman and B. Sudret. Adaptive sparse polynomial chaos expansion based on Least Angle Regression. *J. Comput. Phys.*, 230:2345–2367, 2011.

[30] R. L. Burden, J. D. Faires, and A. M. Burden. *Numerical analysis*. Cengage Learning, 2015.

[31] A. Janon, T. Klein, A. Lagnoux, M. Nodet, and C. Prieur. Asymptotic normality and efficiency of two Sobol' index estimators. *ESAIM: Probab. Stat.*, 18:342–364, 2014.

[32] B. Sudret. Global sensitivity analysis using polynomial chaos expansions. *Reliab. Eng. Sys. Safety*, 93:964–979, 2008.

[33] M. Berveiller, B. Sudret, and M. Lemaire. Stochastic finite elements: a non intrusive approach by regression. *Eur. J. Comput. Mech.*, 15(1-3):81–92, 2006.

[34] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Ann. Statist.*, 32:407–499, 2004.

[35] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.

[36] C. Villani. *Optimal transport, old and new*. Cambridge Series in Statistical and Probabilistic Mathematics. Springer, Cambridge, 2000.

[37] S. L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The Review of Financial Studies*, 6:327–343, 1993.

[38] F.D. Rouah. *The Heston Model in Matlab and C#*. J. Wiley & Sons, New Jersey, 2013.

[39] S. Shreve. *Stochastic Calculus for Finance II*. Springer, New York, 2004.

[40] C. Acerbi. Spectral measures of risk: A coherent representation of subjective risk aversion. *J. Bank. Finance*, 26:1505–1518, 2002.

[41] B Efron. Bootstrap methods: another look at the Jackknife. *Ann. Stat.*, 7(1):1–26, 1979.

[42] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81:2340–2361, 1977.

[43] E. Borgonovo. A new uncertainty importance measure. *Reliab. Eng. Sys. Safety*, 92:771–784, 2007.

[44] Y. Huoh. *Sensitivity Analysis of Stochastic Simulators with Information Theory*. PhD thesis, University of California, Berkeley, 2013.