Reliability Engineering and System Safety 153 (2016) 110–121

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

An efficient modularized sample-based method to estimate the first-order Sobol' index

Chenzhao Li, Sankaran Mahadevan *

Department of Civil and Environmental Engineering, Vanderbilt University, Nashville, TN, USA

ARTICLE INFO

Article history:
Received 1 June 2015
Received in revised form
28 March 2016
Accepted 20 April 2016
Available online 29 April 2016

Keyword:
Global sensitivity analysis
Sobol' index
Sample-based
Correlated variables

ABSTRACT

Sobol' index is a prominent methodology in global sensitivity analysis. This paper aims to directly estimate the Sobol' index based only on available input–output samples, even if the underlying model is unavailable. For this purpose, a new method to calculate the first-order Sobol' index is proposed. The innovation is that the conditional variance and mean in the formula of the first-order index are calculated at an unknown but existing location of model inputs, instead of an explicit user-defined location. The proposed method is modularized in two aspects: 1) index calculations for different model inputs are separate and use the same set of samples; and 2) model input sampling, model evaluation, and index calculation are separate. Due to this modularization, the proposed method is capable to compute the first-order index if only input–output samples are available but the underlying model is unavailable, and its computational cost is not proportional to the dimension of the model inputs. In addition, the proposed method can also estimate the first-order index with correlated model inputs. Considering that the first-order index is a desired metric to rank model inputs but current methods can only handle independent model inputs, the proposed method contributes to fill this gap.

1. Introduction

Uncertainty propagation problems generally involve a computational model in the form of $y=f(\boldsymbol{x},\boldsymbol{d})$ where $\boldsymbol{x}=\{x_1,\ldots,x_k\}$ is the vector of stochastic model inputs and $\boldsymbol{d}$ is the vector of deterministic inputs. Global sensitivity analysis (GSA) studies how the uncertainty in the output y can be apportioned to the uncertainty in the stochastic model inputs $\boldsymbol{x}=\{x_1,\ldots,x_k\}$, so that the importance of each stochastic model input can be ranked. Based on the result of GSA, inputs with negligible contribution can be fixed at their mean values thus reducing the number of stochastic variables. Reviews on various GSA methods can be found in [1,2]. The Sobol' sensitivity indices method based on variance decomposition is a prominent one among these methods. Usage of the Sobol' indices in different engineering problems can be found in [3–7].

Assuming that $y=f(\boldsymbol{x})$ is a real integrable function and all the model inputs $\boldsymbol{x}=\{x_1,\ldots,x_k\}$ are mutually independent, Sobol' [8] proved the following formula to decompose the variance of y:

$$V(y)=\sum_{i}^{k}V_i+\sum_{i_1=1}^{k}\sum_{i_2=i_1+1}^{k}V_{i_1i_2}+\sum_{i_1=1}^{k}\sum_{i_2=i_1+1}^{k}\sum_{i_3=i_2+1}^{k}V_{i_1i_2i_3}+\ldots+V_{12\ldots k} \quad (1)$$

where V_i indicates the variance of y caused by x_i individually, and $V_{i_1\ldots i_s}(s\geq 2)$ indicates the variance of y caused by the interaction of $\{x_{i_1},\ldots,x_{i_s}\}$.

Dividing $V(y)$ at both sides of Eq. (1) for normalization, the Sobol' index is defined as:

$$1=\sum_{i}^{k}S_i+\sum_{i_1=1}^{k}\sum_{i_2=i_1+1}^{k}S_{i_1i_2}+\sum_{i_1=1}^{k}\sum_{i_2=i_1+1}^{k}\sum_{i_3=i_2+1}^{k}S_{i_1i_2i_3}+\ldots+S_{12\ldots k} \quad (2)$$

where the index S_i measures the contribution of x_i alone to the variance of y, without interacting with any other inputs. S_i is called first-order index or main effects index. Other indices $S_{i_1\ldots i_s}(s\geq 2)$ in Eq. (2) are higher-order indices, measuring the contribution of the interaction of $\{x_{i_1},\ldots,x_{i_s}\}$.

* Corresponding author. Tel.: +1 615 322 3040.
E-mail address: sankaran.mahadevan@vanderbilt.edu (S. Mahadevan).

http://dx.doi.org/10.1016/j.ress.2016.04.012
0951-8320/

This paper focuses on calculating the first-order index S_i, which is one of the important objectives in variance-based global sensitivity analysis. The calculation of S_i is based on the following formula:

$$S_i = \frac{V_i}{V(y)} = \frac{V_{x_i}(E_{\boldsymbol{x}_{-i}}(y|x_i))}{V(y)} \tag{3}$$

where $\boldsymbol{x}_{-i}$ means all the model inputs other than x_i.

Based on Eq. (3), computing S_i analytically is nontrivial, since $E_{\boldsymbol{x}_{-i}}(\bullet)$ in Eq. (3) indicates a multi-dimensional integral. Computing S_i by Monte Carlo simulation (MCS) directly is also expensive. The numerator in Eq. (3) leads to a double-loop MCS [1]: the inner loop $E_{\boldsymbol{x}_{-i}}(y|x_i)$ computes the mean value of y using n_1 random samples of $\boldsymbol{x}_{-i}$; and the outer loop computes $V_{x_i}(E_{\boldsymbol{x}_{-i}}(y|x_i))$ by iterating the inner loop n_2 times at different values of x_i. In addition, another n_3 MCS iterations are required to compute $V(y)$ in Eq. (3). The cost of this sample-based method, defined as the total number of model evaluations to compute all S_i ($i = 1$ to k), is $kn_{dl}^2 + n_{dl}$ if $n_1 = n_2 = n_3 = n_{dl}$.This cost increases with n_{dl} and k, and is unaffordable if a single model evaluation is time-consuming or economically expensive, since n_{dl} is often of the order greater than 1000 in many practical applications.

Various algorithms have been proposed to reduce the computational cost of the Sobol' indices. These algorithms can be categorized into analytical methods and sample-based methods. In the analytical methods, the original model $y = f(\boldsymbol{x})$ is generally approximated by some surrogate model of special form, so that the multi-dimensional integral can be converted into multiple univariate integrals, which can be easily calculated analytically or numerically. Zhang and Pandey [9] approximated the original model $y = f(\boldsymbol{x})$ by a multiplication of univariate functions; then the univariate integral was calculated by Gaussian quadrature. Sudret [10] proposed that if the original model is approximated by a polynomial chaos expansion (PCE), the Sobol' index can be calculated by post-processing the PCE coefficients. Chen et al. [11] proposed another analytical method for commonly used surrogate models such as the linear regression model, Gaussian process model [12], Gaussian radial basis function model, and MARS model [13]; and analytical solution of the Sobol' index is available if the model inputs are normally or uniformly distributed. Analytical methods reduce the number of model evaluations significantly, but may require: 1) extra approximations and assumptions, and 2) extra computational cost in building the surrogate model.

Compared to the analytical methods, sample-based methods are more widely used [14–18] in engineering due to their simplicity in implementation. The basic sample-based method for GSA is the double-loop MCS, which has been explained earlier and often has prohibitive computational cost. Various efficient sample-based methods have been developed in the literature to reduce this cost. A brief review of these sample-based methods is given in Section 2. To the authors' knowledge, the computational cost (number of model evaluations) of most sample-based methods is proportional to the model input dimension k. Therefore the first objective of this paper is to develop a more efficient sample-based method whose computational cost is not proportional to k, but much less.

A key assumption of the Sobol' index is the mutual independence of model inputs. With correlated model inputs, Eqs. (1) and (2) are no longer valid. However, Saltelli [19] pointed out that the first-order index S_i is still an informed choice to rank the importance of correlated model inputs, since S_i can be defined in another way where independent model inputs are not assumed:

1. The importance of x_i at a particular location x_i can be measured by $V_{\boldsymbol{x}_{-i}}(y|x_i = \tilde{x}_i)$, i.e., smaller $V_{\boldsymbol{x}_{-i}}(y|x_i = \tilde{x}_i)$ indicates greater importance of x_i;
2. The dependence of this measurement on the location of x_i is removed by taking the average of $V_{\boldsymbol{x}_{-i}}(y|x_i = \tilde{x}_i)$, i.e. $E_{x_i}(V_{\boldsymbol{x}_{-i}}(y|x_i))$;
3. By the law of total variance $V(y) = E_{x_i}(V_{\boldsymbol{x}_{-i}}(y|x_i)) + V_{x_i}(E_{\boldsymbol{x}_{-i}}(y|x_i))$, a larger $V_{x_i}(E_{\boldsymbol{x}_{-i}}(y|x_i))$ equally indicates a greater importance of x_i;
4. The first-order index is redefined by normalization, thus $S_i = V_{x_i}(E_{\boldsymbol{x}_{-i}}(y|x_i))/V(y)$.

Saltelli's paper [20] in 2002 mentioned that there is no alternative to the expensive double-loop MCS to compute S_i with correlated model inputs. The authors have not found any efficient algorithm in more recent studies, either. Thus the second objective of this paper is to develop an efficient algorithm that can handle correlated model inputs.

The outline of this paper is as follows. Section 2 briefly reviews existing sample-based methods for GSA, and discusses their computational cost. Section 3 illustrates the proposed modularized sample-based method that reduces the computational cost and handles correlated model inputs. Section 4 uses three numerical examples to compare the proposed method with existing methods.

2. Background: sample-based methods to estimate the first-order Sobol' index

2.1. Sobol' scheme

Consider a real integrable function $y = f(\boldsymbol{x})$ where $\boldsymbol{x} = \{x_1, \ldots, x_k\}$ is the vector of independent model inputs. Denote $\boldsymbol{z} = \{z_1, \ldots, z_k\}$ as the vector of the same independent model inputs, i.e., $z_i (i = 1 \text{to} k)$ and x_i are independently and identically distributed (i.i.d.). Sobol' [8] developed the following formula to compute the first-order index:

$$V_i = \int f(\boldsymbol{x}) f(x_i, \boldsymbol{z}_{-i}) p(\boldsymbol{x}) p(\boldsymbol{z}_{-i}) d\boldsymbol{x} d\boldsymbol{z}_{-i} - E^2(y) \tag{4}$$

where $p(\bullet)$ denotes the joint probability density function (PDF) of all the arguments, and it is the product of the PDFs of individual arguments under the assumption of independent model inputs. $\boldsymbol{z}_{-i}$ are all the variables in $\boldsymbol{z}$ other than z_i.

Eq. (4) leads to the following estimator of V_i:

$$V_i = \frac{1}{n}\sum_{j=1}^{n} f\left(\boldsymbol{x}^j\right) f\left(x_i^j, \boldsymbol{z}_{-i}^j\right) - \left[\frac{1}{n}\sum_{j=1}^{n} f\left(\boldsymbol{x}^j\right)\right]^2 \tag{5}$$

Eq. (5) requires n_s samples of $\boldsymbol{x}$ and n_s samples of $\boldsymbol{z}$, which are sampled independently from the distributions of the model inputs. In Eq. (5), the superscript j is the index of the samples and the subscript i is the index of model inputs. For example, $\boldsymbol{x}^j$ means the j-th sample of $\boldsymbol{x}$, and $\boldsymbol{z}_{-i}^j$ means the j-th sample of $\boldsymbol{z}$ except z_i. In Eq. (5), $f(\boldsymbol{x}^j)$ implies n_s model evaluations; $f\left(x_i^j, \boldsymbol{z}_{-i}^j\right)$ implies n_s model evaluations for each model input, i.e., kn_s evaluations for all the model inputs. To improve the accuracy, generally another n_s model evaluations are needed over the samples in $\boldsymbol{z}$, and the results are used to estimate $V(y)$ together with earlier evaluations over $\boldsymbol{x}$. The first-order index is calculated as $S_i = V_i/V(y)$. The overall cost for all the first-order indices is $kn_s + 2n_s$.

Eq. (5) is the first efficient sample-based method to compute the first-order Sobol' index. Several methods have been proposed to improve its accuracy or reduce computational cost. Homma and Saltelli [21] suggested a more accurate estimator of V_i by using $\frac{1}{n}\sum_{j=1}^{n} f(\boldsymbol{x}^j) f(\boldsymbol{z}^j)$ to calculate $E^2(Y)$ instead of $\left[\frac{1}{n}\sum_{j=1}^{n} f(\boldsymbol{x}^j)\right]^2$. Thus Eq. (5) becomes [22]:

$$V_i = \frac{1}{n}\sum_{j=1}^{n} f\left(\boldsymbol{x}^j\right)\left[f\left(x_i^j, \boldsymbol{z}_{-i}^j\right) - f\left(\boldsymbol{z}^j\right)\right] \tag{6}$$

Compared to Eq. (5), Eq. (6) brings no extra model evaluation.

Sobol' and Myshetskaya [23] improved Eq. (6) further by replacing $f(\boldsymbol{x}^j)$ with $f(\boldsymbol{x}^j)-c$, where c is a constant equal or close to the true value of $E(y)$. Thus Eq. (6) becomes:

$$V_i=\frac{1}{n}\sum_{j=1}^{n}\left[f\left(\boldsymbol{x}^j\right)-c\right]\left[f\left(x_i^j,\boldsymbol{z}_{-i}^j\right)-f\left(\boldsymbol{z}^j\right)\right] \tag{7}$$

Eq. (7) brings no extra model evaluation either. In the numerical examples in Section 4, we define c as the mean value of y over all samples of $\boldsymbol{x}$ and $\boldsymbol{z}$.

In addition, another formula for V_i to improve the accuracy of small S_i is proposed by Owen [24]:

$$V_i=\frac{1}{n}\sum_{j=1}^{n}\left[f\left(\boldsymbol{x}^j\right)-f\left(w_i^j,\boldsymbol{x}_{-i}^j\right)\right]\left[f\left(x_i^j,\boldsymbol{z}_{-i}^j\right)-f\left(\boldsymbol{z}^j\right)\right] \tag{8}$$

In Eq. (8) another i.i.d of $\boldsymbol{x}$ is denoted as $\boldsymbol{w}$, and a sample set of size n_s is generated for $\boldsymbol{w}$ so that w_i^j is the j-th sample of the i-th model input in this sample set. Eq. (8) proves to be more accurate in estimating small S_i; but no accuracy improvement is observed in estimating large S_i [24]. In addition, the term $f\left(w_i^j,\boldsymbol{x}_{-i}^j\right)$ in Eq. (8) brings n_s more model evaluations to estimate a single S_i.

More sample-based methods derived from Eq. (5) can be found in [20,25–27]. This paper does not describe all these methods due to space limitations. Note that all the existing sample-based methods using the Sobol' scheme have a computational cost proportional to model inputs dimension k.

2.2. FAST scheme

The FAST (Fourier amplitude sensitivity test) scheme includes two methods: classical FAST [28] and improved FAST [29] based on random balanced design [30]. The classical FAST was introduced in the 1970s, earlier than the introduction of Sobol' index. However, FAST estimates the equivalent of the first-order index defined in Eq. (3); thus the classical FAST is considered as an algorithm to compute the first-order index.

The classical FAST method assumes that any model input x_i ($i=1\text{to}\,k$) follows the standard uniform distribution $U(0,1)$, such that the domain of the model inputs is a unit hypercube $C^k=(\boldsymbol{x}|0\le x_i\le 1; i=1\text{to}\,k)$. This can be satisfied by converting all the model inputs into their CDF space.

Instead of directly generating random samples of $\boldsymbol{x}$ to fill in the sampling space C^k, FAST utilizes a curve to explore it. This curve is defined as:

$$x_i(s)=G_i(\sin\omega_i s)\forall i=1\text{to}\,k \tag{9}$$

In Eq. (9), s varies in $[-\pi,\pi]$; ω_i is the angular frequency of x_i, set as linearly independent positive integers, and detailed strategy to select ω_i can be found in [31]; $G_i(\bullet)$ is a transfer function.

The curve in Eq. (9) explores the hypercube C^k as s changes. In other words, by generating samples of s from the uniform distribution $U(-\pi,\pi)$, the corresponding samples of $x_i(i=1\text{to}k)$ can be obtained by Eq. (9). The resultant samples of $x_i(i=1\text{to}\,k)$ should follow the uniform distribution $U(0,1)$, and the samples of x_i and $x_j(i\ne j)$ should be independent. These two objectives are achieved by the designed transfer function $G_i(\bullet)$. Different forms of $G_i(\bullet)$ have been proposed in [28,32,33].

Substituting Eq. (9) into the original model $y=f(\boldsymbol{x})$ results in a new function of s denoted as $y=f(\boldsymbol{x}(s))$, which can be expanded into a Fourier series. Then the total variance $V(y)$ and the output variance caused by x_i itself are:

$$V(y)=2\sum_{p=1}^{+\infty}\Lambda_p, V_i=2\sum_{p=1}^{+\infty}\Lambda_{p\omega_i} \tag{10}$$

where p can be any positive integer; Λ_p and $\Lambda_{p\omega_i}$ are the Fourier spectrum at frequency p and $p\omega_i$, respectively. Eq. (10) means that V_i is related to the Fourier spectrum at the selected frequency ω_i and its higher harmonics $p\omega_i$.

In numerical computation, n_F samples of s are uniformly generated from $[-\pi,\pi]$, corresponding to n_F underlying samples of $\boldsymbol{x}=x_1,\ldots,x_k$. The model $y=f(\boldsymbol{x}(s))$ is evaluated at these samples to obtain the model output values, based on which the Fourier spectrum Λ_p can be computed by a numerical integral. Usually $\Lambda_{p\omega_i}$ is computed up to $H\omega_i$, where H is usually set to 4 or 6. Fourier coefficients at frequencies higher than $H\omega_i$ can be ignored in Eq. (10).

The computational cost of classical FAST is simply n_F, since the same model evaluation $f(\boldsymbol{x}(s))$ can be used to evaluate different V_i. However, n_F is constrained to a lower limit $n_F\ge 2\max(\omega_i)+1$ [28]. According to the algorithm in [31] to select ω_i, $\max(\omega_i)$ increases with the input dimension k, thus the computational cost n_F also increases as k.

The improved FAST combines the classical FAST above with random balanced design. The improved FAST generates n_F samples of s with $\omega_i=1\,\forall i=1$ to k. Then the model $y=f(\boldsymbol{x}(s))$ is evaluated n_F times to obtain the corresponding output values, denoted as $y(s^j), j=1\text{to}\,n_F$.

To obtain V_i, the output values $y(s^j)$ are reordered as $y^R(s^j)$ such that the corresponding values of x_i are ranked in increasing order. Then V_i is calculated in the same way as for the classical FAST approach by computing Fourier spectrum.

The improved FAST has no lower limit of sample size thus its computational cost n_F is not related to the model input dimension k. In addition, this method also achieves better accuracy [1,29,34] than the classical FAST.

3. Proposed method

The motivation of the proposed method is rooted in the following challenge: with the input–output samples regarding a physics/computational model available, can we directly estimate the Sobol' index from the samples without more model evaluations? The intuitive answer should be yes, since the resultant input–output samples contain information about 1) the underlying input–output functional relationship, and 2) the underlying input/output distributions.

One GSA method based on the classical ANOVA using factorial design of experiments was proposed in [35]. If the random samples of each model input are considered as the levels of factors in the factorial design, this method gives the same result as the Sobol' index since the variance decomposition powering the Sobol' index is the same as that used in the classical ANOVA [36]. However, the factorial design in this method requires all possible combinations of the model input samples (levels) [35] and the corresponding model output samples, thus common MCS samples are not applicable.

In this paper, a new sample-based method is proposed to resolve this challenge. Instead of modifying or improving the Sobol' scheme or the FAST approach, the proposed method is developed by analyzing the inner and outer loops of MCS in calculating the first-order index.

3.1. Proposed Algorithm 1

The proposed Algorithm 1 addresses the first-order index expression of Eq. (3), whose numerator includes the inner loop $E_{\boldsymbol{x}_{-i}}(y|x_i)$ and the outer loop $V_{x_i}(\bullet)$.

Consider a model of $y=f(\boldsymbol{x})$ where $\boldsymbol{x}=\{x_1,\ldots,x_k\}$. We divide $\boldsymbol{x}$ into two sets: $\boldsymbol{x}=\{x_i,\boldsymbol{x}_{-i}\}$ where $\boldsymbol{x}_{-i}$ are the inputs other than x_i. $E_{\boldsymbol{x}_{-i}}(y|x_i)$ is a function of x_i, and it can be proved that the mean value of y over $\boldsymbol{x}=\{x_i,\boldsymbol{x}_{-i}\}$ is equal to the average of $E_{\boldsymbol{x}_{-i}}(y|x_i)$ over

x_i:

$$E_{x_i}\left(E_{\boldsymbol{x}_{-i}}(y|x_i)\right) = \int E_{\boldsymbol{x}_{-i}}(y|x_i)p(x_i)dx_i$$
$$= \int \left(\int f(x_i,\boldsymbol{x}_{-i})p(\boldsymbol{x}_{-i}|x_i)d\boldsymbol{x}_{-i}\right)p(x_i)dx_i$$
$$= \int\int f(x_i,\boldsymbol{x}_{-i})p(x_i,\boldsymbol{x}_{-i})dx_i d\boldsymbol{x}_{-i} = E(y) \quad (11)$$

Eq. (11) is called the law of total expectation and can be found in [37]. In Eq. (11), if x_i is constrained into a closed and bounded interval Φ, the distribution of x_i (and $\boldsymbol{x}_{-i}$ for correlated inputs) will change but Eq. (11) is still valid. In this case, based on the extreme value theorem [38], if $E_{\boldsymbol{x}_{-i}}(y|x_i)$ is a continuous function of x_i in Φ, it must have a maximum value $\max_{x_i\in\Phi}\left(E_{\boldsymbol{x}_{-i}}(y|x_i)\right)$ and a minimum value $\min_{x_i\in\Phi}\left(E_{\boldsymbol{x}_{-i}}(y|x_i)\right)$ in Φ. The mean value of $E_{\boldsymbol{x}_{-i}}(y|x_i)$, i.e., $E_\Phi\left(E_{\boldsymbol{x}_{-i}}(y|x_i)\right)$, is between these maximum and minimum values. Due to $x_i\in\Phi$, we denote the mean value of y as $E_\Phi(y)$. With $E_\Phi(y)=E_\Phi\left(E_{\boldsymbol{x}_{-i}}(y|x_i)\right)$ proved in Eq. (11), we obtain

$$\min_{x_i\in\Phi}\left(E_{\boldsymbol{x}_{-i}}(y|x_i)\right) \le E_\Phi(y) \le \max_{x_i\in\Phi}\left(E_{\boldsymbol{x}_{-i}}(y|x_i)\right) \quad (12)$$

Furthermore, since $E_{\boldsymbol{x}_{-i}}(y|x_i)$ is a continuous function in Φ, the intermediate value theorem [38] proves that

$$\exists x_i^*\in\Phi \text{such that} E_\Phi(y)=E_{\boldsymbol{x}_{-i}}(y|x_i^*) \quad (13)$$

Eq. (13) leads to the proposed Algorithm 1 if we design the interval Φ based on stratified sampling. As the mathematical basis of Eqs. (12) and (13), the extreme value theorem and the intermediate value theorem can be found in the Appendix of this paper.

Stratified sampling generates samples in equal probability intervals to represent the distribution of a random variable x. Fig. 1 (a) shows one strategy [1] of stratified sampling: 1) divide the CDF of x_i into M intervals such that these intervals have the same length; 2) generate one sample u^l (the red dots in Fig. 1(a), and $l=1\text{to}M$) from each CDF interval and obtain samples of x_i (the green dots in Fig. 1) by CDF inversion $x_i^l=P^{-1}(u^l)$, where $P^{-1}(\bullet)$ is the inverse CDF of x_i. If we take the bounds of these intervals of the CDF as the inputs of $P^{-1}(\bullet)$, the sampling space of x_i is actually divided into M equally probable intervals $\Phi^l(l=1\text{to}M)$, as shown in Fig. 1(b), and x_i^l is actually a random sample generated within Φ^l.

Consider the inner loop $E_{\boldsymbol{x}_{-i}}(yx_i)$ in Eq. (3) first. Assuming $\Phi=\Phi^l$, Eq. (13) proves that $\exists x_i^{l*}\in\Phi^l$ such that $E_{\boldsymbol{x}_{-i}}(y|x_i^{l*})=E_{\Phi^l}(y)$, where $E_{\Phi^l}(y)$ is the mean value of y with $x_i\in\Phi^l$. In other words, calculating $E_{\Phi^l}(y)$ is equivalent to fixing x_i at an unknown but existing point $x_i^{l*}\in\Phi^l$ and calculating the conditional mean value $E_{\boldsymbol{x}_{-i}}(y|x_i=x_i^{l*})$.

The outer loop $V_{x_i}(\bullet)$ requires fixing x_i at different locations, and these selected locations are samples from the distribution of x_i. Based on stratified sampling, the set of these unknown but existing points $\boldsymbol{x}_i^*=\{x_i^{1*},\ldots,x_i^{M*}\}$ from the equally probable intervals $\boldsymbol{\Phi}=\{\Phi^1,\ldots,\Phi^M\}$ can represent the distribution of x_i. As $E_{\Phi^l}(y)=E_{\boldsymbol{x}_{-i}}(y|x_i=x_i^{l*})$, the computation of S_i in the proposed Algorithm 1 is expressed as

$$S_i=\frac{V_{\boldsymbol{\Phi}}\left(E_{\Phi^l}(y)\right)}{V(y)} \quad (14)$$

where it numerator is the variance of $\left\{E_{\Phi^1}(y),E_{\Phi^2}(y),\ldots,E_{\Phi^M}(y)\right\}$. The steps to realize Algorithm 1 are listed in Section 3.3.

3.2. Proposed Algorithm 2

Based on the law of total variance

$$V(y)=E_{x_i}\left(V_{\boldsymbol{x}_{-i}}(yx_i)\right)+V_{x_i}\left(E_{\boldsymbol{x}_{-i}}(yx_i)\right) \quad (15)$$

Eq. (3) can be rewritten as

$$S_i=1-\frac{E_{x_i}\left(V_{\boldsymbol{x}_{-i}}(yx_i)\right)}{V(y)} \quad (16)$$

The proposed Algorithm 2 is regarding this equivalent first-order Sobol' index expression in Eq. (16), whose numerator implies an expensive double-loop Monte Carlo simulation including the inner loop $V_{\boldsymbol{x}_{-i}}(yx_i)$ and the outer loop $E_{x_i}(\bullet)$. Its inner loop part $V_{\boldsymbol{x}_{-i}}(yx_i)$ is a function of x_i. Assume $x_i\in\Phi$, where Φ can be the entire sampling space of x_i or only a small interval. Based on the extreme value theorem, $V_{\boldsymbol{x}_{-i}}(yx_i)$ must have a maximum value and a minimum value in Φ. The mean value of $V_{\boldsymbol{x}_{-i}}(y|x_i)$, i.e., $E_\Phi\left(V_{\boldsymbol{x}_{-i}}(yx_i)\right)$ for $x_i\in\Phi$, is between these maximum and minimum values:

$$\min_{x_i\in\Phi}\left(V_{\boldsymbol{x}_{-i}}(y|x_i)\right) \le E_\Phi\left(V_{\boldsymbol{x}_{-i}}(yx_i)\right) \le \max_{x_i\in\Phi}\left(V_{\boldsymbol{x}_{-i}}(y|x_i)\right) \quad (17)$$

Then the intermediate value theorem proves that

$$\exists x_i^{\#}\in\Phi s.\text{t.}V_{\boldsymbol{x}_{-i}}(y|x_i^{\#})=E_\Phi\left(V_{\boldsymbol{x}_{-i}}(yx_i)\right) \quad (18)$$

With $x_i\in\Phi$, we rewrite the law of total variance in Eq. (15) as:

$$V_\Phi(y)=E_\Phi\left(V_{\boldsymbol{x}_{-i}}(yx_i)\right)+V_\Phi\left(E_{\boldsymbol{x}_{-i}}(yx_i)\right) \quad (19)$$

where the subscript Φ means all the terms are constrained to $x_i\in\Phi$. Substituting Eq. (19) into Eq. (18) and assuming $\Phi=\Phi^l$ as one of the equally probable intervals in stratifying sampling, we can have

$$\exists x_i^{l\#}\in\Phi^l s.\text{t.}V_{\boldsymbol{x}_{-i}}\left(y|x_i^{l\#}\right)=V_{\Phi^l}(y)-V_{\Phi^l}\left(E_{\boldsymbol{x}_{-i}}(yx_i)\right) \quad (20)$$

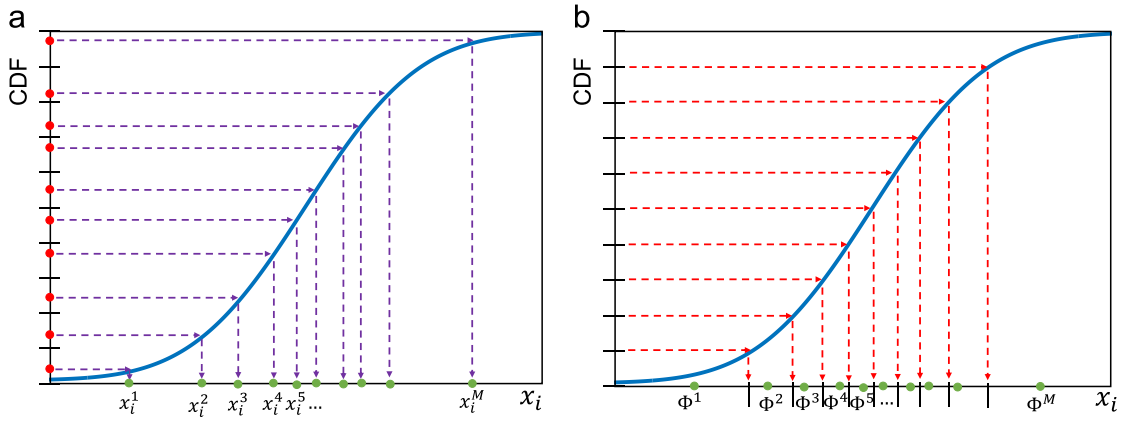

Fig. 1. Stratified sampling and equally probably intervals. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

where $x_i^{I\#}$ is an unknown but existing point in Φ^I. Note that now $V_{\Phi^I}(y)$ is the variance of y given $x_i \in \Phi^I$ and $V_{\Phi}(E_{\boldsymbol{x}_{-i}}(yx_i))$ is the variance of $E_{\boldsymbol{x}_{-i}}(yx_i)$ given $x_i \in \Phi^I$.

The outer loop $E_{x_i}(\bullet)$ requires fixing x_i at different locations, and these selected locations are samples from the distribution of x_i. Based on stratified sampling, the set of these unknown but existing points $\boldsymbol{x}_i^{\#} = \{x_i^{1\#}, \ldots, x_i^{M\#}\}$ from the equally probable intervals $\boldsymbol{\Phi} = \{\Phi^1, \ldots, \Phi^M\}$ can represent the distribution of x_i. Based on Eqs. (16) and (20), computation of S_i is expressed as

$$S_i = 1 - \frac{E_{\boldsymbol{\Phi}}(V_{\boldsymbol{x}_{-i}}(y|x_i^{I\#}))}{V(y)} = 1 - \frac{E_{\boldsymbol{\Phi}}\left(V_{\Phi^I}(y) - V_{\Phi^I}(E_{\boldsymbol{x}_{-i}}(y|x_i))\right)}{V(y)}$$
$$= 1 - \frac{E_{\boldsymbol{\Phi}}\left(V_{\Phi^I}(y)\right)}{V(y)} + \frac{E_{\boldsymbol{\Phi}}\left(V_{\Phi^I}(E_{\boldsymbol{x}_{-i}}(y|x_i))\right)}{V(y)} \tag{21}$$

On the right-hand side of Eq. (21), the first term is a known constant 1; the second term can be directly computed using the Monte Carlo samples, following the steps given later in Fig. 2; the third term is still a challenge but we can prove that this term can be ignored by rewriting it as:

$$\frac{E_{\boldsymbol{\Phi}}\left(V_{\Phi^I}(E_{\boldsymbol{x}_{-i}}(yx_i))\right)}{V(y)} = E_{\boldsymbol{\Phi}}\left(\frac{V_{\Phi^I}(E_{\boldsymbol{x}_{-i}}(yx_i))}{V_{\Phi^I}(y)}\frac{V_{\Phi^I}(y)}{V(y)}\right) \tag{22}$$

In Eq. (22), the term $V_{\Phi^I}(E_{\boldsymbol{x}_{-i}}(yx_i))/V_{\Phi^I}(y) = S_i^{\Phi^I}$ is nothing but the first-order sensitivity of x_i as it is restricted to the interval Φ^I. We always have $S_i^{\Phi^I} < S_i$ since the uncertainty of x_i has been reduced significantly by restricting it in Φ^I such that its sensitivity index will be much lower. The other term in Eq. (22) $V_{\Phi^I}(y)/V(y)$ is the ratio of 1) the variance of y as x_i is restricted to the interval Φ^I and 2) the overall variance of y.

If x_i has a high sensitivity index S_i close to one, restricting it to Φ^I will reduce the variance of y significantly such that $V_{\Phi^I}(y)/V(y)$ will be close to zero; meanwhile $S_i^{\Phi^I}$ will be also smaller than S_i. Overall, their product will be close to zero.

If x_i has a low sensitivity index S_i closer to zero, restricting it to Φ^I will NOT reduce the variance of y significantly such that $V_{\Phi^I}(y)/V(y)$ will be close to 1; however, we always have $S_i^{\Phi^I} < S_i$ so that $S_i^{\Phi^I}$ is closer to zero. Overall, their product will be close to zero.

In sum, no matter whether S_i is closer to zero or one, Eq. (22) is always a small value close to zero, and this value will reduce further as the number of intervals M increases, since in that case Φ will be narrower so that . Thus Eq. (21) can be approximated as

$$S_i \approx 1 - \frac{E_{\boldsymbol{\Phi}}\left(V_{\Phi^I}(y)\right)}{V(y)} \tag{23}$$

Eq. (23) is the proposed Algorithm 2, and the steps to realize it are listed in Section 3.3.

3.3. Implementation and benefits of modularization

The innovation in the proposed methods is that the inner loop $E_{\boldsymbol{x}_{-i}}(yx_i)$ or $V_{\boldsymbol{x}_{-i}}(yx_i)$ is not conditioned on an explicit sample of x_i selected by the user, but on an unknown but existing point. The first-order index S_i is obtained without knowing the value of this existing point. The benefits of the proposed methods can be observed from the following steps to realize Eqs. (14) and (23):

1. Generate n_M random samples of $\boldsymbol{x}$;
2. Obtain corresponding values of y by evaluating $y = f(\boldsymbol{x})$, and estimate $V(y)$ using all samples of y;
3. Divide the domain of x_i into M equally probable intervals, as shown in Fig. 1;
4. Assign the samples of y into divided intervals based on one-to-one mapping between the samples of x_i and samples of y;
5. For Algorithm 1, estimate $E_{\Phi^I}(y)$ as the sampling mean of y in each interval; for Algorithm 2, estimate $V_{\Phi^I}(y)$ as the sampling variance of y in each interval;
6. For Algorithm 1, estimate $V_{\boldsymbol{\Phi}}\left(E_{\Phi^I}(y)\right)$ as the sampling variance of $E_{\Phi^I}(y)$ in step 5; for Algorithm 2, estimate $E_{\boldsymbol{\Phi}}\left(V_{\Phi^I}(y)\right)$ as the sampling mean of $V_{\Phi^I}(y)$ in step 5;
7. $S_i = V_{\boldsymbol{\Phi}}\left(E_{\Phi^I}(y)\right)/V(y)$ for Algorithm 1 and $S_i = 1 - E_{\boldsymbol{\Phi}}\left(V_{\Phi^I}(y)\right)/V(y)$ for Algorithm 2.

The steps to realize the proposed method are also illustrated in Fig. 2, where samples in different equally probable intervals are represented in different colors. These steps indicate that the proposed methods are modularized in two aspects. First, Steps 3 and 4 show that the samples of $\boldsymbol{x}_{-i}$ are not used in calculating the index S_i for x_i, so that index calculations for different model inputs are separated. Therefore *the calculation of S_i purely depends on the samples of x_i and y, and can be achieved even if the samples of $\boldsymbol{x}_{-i}$ are missing*. Second, model inputs sampling, model evaluation, and index calculation are separate processes. The computational cost of most existing sample-based methods is proportional to the model inputs dimension k because each input needs *new samples* to calculate its Sobol' index. In comparison, the *computational cost of the proposed method is not proportional to k because each input uses the same samples to calculate its Sobol' index*. Therefore in the proposed method the accuracy of the resultant Sobol' index only relies on the number of samples n_M and the selected value of M, but not dependent on k.

Another benefit brought by this modularization is that the input–output samples in step 1 can be from other uncertainty quantification activities. *It provides a solution of sensitivity analysis when input–output samples are available but the underlying model is not available or too expensive for re-running*.

One very important benefit of the proposed methods is that the derivation of these proposed algorithms does not assume

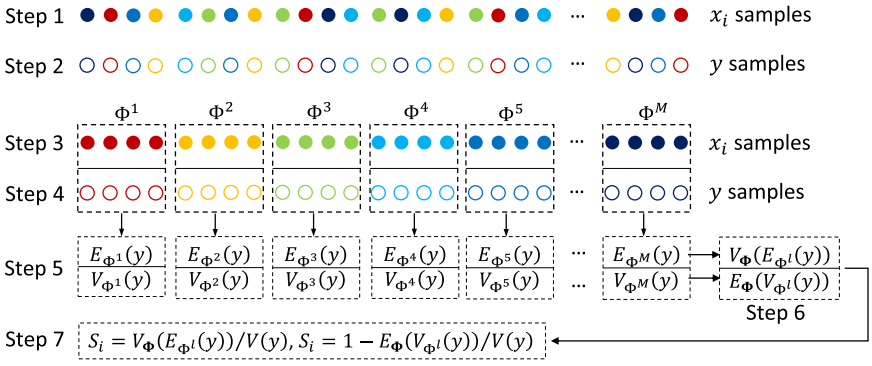

Fig. 2. Steps to realize the proposed method.

independent model inputs. Thus the proposed methods can handle both independent and correlated model inputs. To the authors' knowledge, the proposed method is the only available alternative so far to the costly double-loop MCS method when the model inputs are correlated.

3.4. Accuracy comparison: Algorithm 1 vs. Algorithm 2

For a given set of input–output samples, the factor affecting the implementation of the proposed method is M, the number of equally probable intervals used to stratify the samples. This section identifies the effect of M on the proposed algorithms and compares their accuracy. The algorithm found to be better will then be used to compare against existing methods.

First, Algorithms 1 and 2 are compared by an illustrative example of a thick cantilever beam shown in Fig. 3. This example computes the beam's tip deflection along the y-axis using the Timoshenko beam theory [39]:

$$u=\frac{P}{6EI}\left[(4+5\nu)\frac{h^2L}{4}+2L^3\right] \tag{24}$$

where $I=bh^3/12$. The statistics of other model inputs in Eq. (24) are listed in Table 1.

The proposed two algorithms are used to calculate the first-order index using 10^4 MCS samples. Results for different interval numbers M are shown in Figs. 4 and 5, where the "True value" is estimated by the costly double-loop MCS method with $n_{dl}=10^4$.

Fig. 4 clearly shows that Algorithm 1 tends to overestimate the first-order index, especially at large M values; while Fig. 5 shows that Algorithm 2 is more robust and reveals adequate accuracy at different values of M. This observation can be explained by analyzing the numerical errors in implementing Algorithms 1 and 2.

In Algorithm 1, assume that the true mean value of y in the l-th interval Φ^l is $\mu^l(i=1\text{ to }M)$ while the estimated sampling mean value is $\bar{y}^l$. We denote $\bar{y}^l=\mu^l+d^l$ where d^l is the bias due to limited samples in Φ^l. At given M, the best estimate for $V_{x_i}(E_{\boldsymbol{x}_{-i}}(yx_i))$ is

$$\hat{V}=\frac{1}{M-1}\sum_{l=1}^{M}\left(\mu^l-\overline{\mu}\right)^2 \tag{25}$$

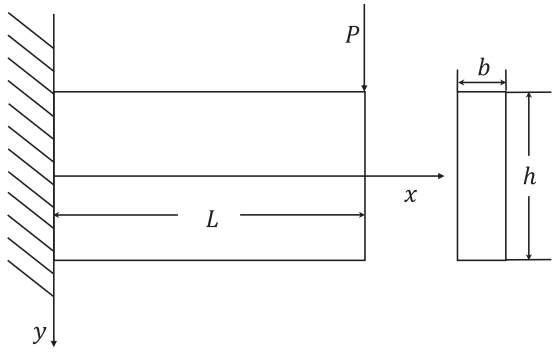

Fig. 3. Thick cantilever beam.

where $\overline{\mu}$ is the mean value of μ^l, i.e., $\overline{\mu}=(\sum_{l=1}^{M}\mu^l)/M$. This $\hat{V}$ approximates the desired $V_{x_i}(E_{\boldsymbol{x}_{-i}}(yx_i))$ well if M is large enough. However, $\hat{V}$ can be obtained only if $\mu^l\forall l=1\text{to}M$ is estimated correctly, which cannot be achieved due to the numerical errors. Denoting $\overline{\bar{y}}=(\sum_{l=1}^{M}\bar{y}^l)/M$ and $\overline{d}=(\sum_{l=1}^{M}d^l)/M$, the actual estimate of $V_{x_i}(E_{\boldsymbol{x}_{-i}}(Yx_i))$ is

$$\tilde{V}=\frac{1}{M-1}\sum_{l=1}^{M}\left(\bar{y}^l-\overline{\bar{y}}\right)^2=\frac{1}{M-1}\sum_{l=1}^{M}\left(\mu^l+d^l-\overline{\mu}-\overline{d}\right)^2 \tag{26}$$

The bias of $\tilde{V}$ from $\hat{V}$ is

$$\begin{aligned}\tilde{V}-\hat{V}&=\frac{1}{M-1}\left[\sum_{l=1}^{M}\left(\mu^l+d^l-\overline{\mu}-\overline{d}\right)^2-\sum_{l=1}^{M}\left(\mu^l-\overline{\mu}\right)^2\right]\\&=\frac{1}{M-1}\left[\sum_{l=1}^{M}\left(2\mu^l+d^l-2\overline{\mu}-\overline{d}\right)\left(d^l-\overline{d}\right)\right]\\&=\frac{2(\mu^l-\overline{\mu})}{M-1}\sum_{l=1}^{M}\left(d^l-\overline{d}\right)+\frac{1}{M-1}\sum_{l=1}^{M}\left(d^l-\overline{d}\right)^2\\&=\frac{1}{M-1}\sum_{l=1}^{M}\left(d^l-\overline{d}\right)^2>0\end{aligned} \tag{27}$$

Eq. (27) clearly indicates that $\tilde{V}$ is a positively biased estimate of $\hat{V}$, where $\hat{V}$ is used to approximate the desired term $V_{x_i}(E_{\boldsymbol{x}_{-i}}(yx_i))$. Due to the square term in the last line of Eq. (27), this bias tends to increase as M becomes large. This explains why Algorithm 1 overestimates the first-order indices in Fig. 4 and why this overestimation increases with M.

In Algorithm 2, assume that the true variance of y in the l-th interval Φ^l is $S^l(i=1\text{to}M)$ while the estimated sampling mean value is V^l. We denote $V^l=S^l+\delta^l$ where δ^l is the bias due to limited sample in Φ^l. At given M, the best estimate of $E_{x_i}(V_{\boldsymbol{x}_{-i}}(yx_i))$ is $\hat{E}=\left(\sum_{l=1}^{M}S^l\right)/M$ while the actual estimate is $\tilde{E}=\left(\sum_{l=1}^{M}V^l\right)/M$. The bias of $\tilde{E}$ from $\hat{E}$ is

$$\tilde{E}-\hat{E}=\frac{1}{M}\sum_{l=1}^{M}\left(V^l-S^l\right)=\frac{1}{M}\sum_{l=1}^{M}\delta^l \tag{28}$$

The bias from Eq. (28) is around zero since δ^l can be randomly positive or negative. Then $\tilde{E}$ is an unbiased estimate of $\hat{E}$, where $\hat{E}$ is used to approximate the desired term $E_{x_i}(V_{\boldsymbol{x}_{-i}}(yx_i))$. This explains why Algorithm 2 estimates the first-order index accurately at different values of M. In conclusion, Algorithm 2 is more accurate and robust than Algorithm 1. Note that Eqs. (27) and (28) which compare the accuracy of Algorithms 1 and 2 are general; the cantilever beam example was only for illustrative purposes.

In this cantilever beam example, Fig. 5 proves the robustness of Algorithm 2 at different values of M. Section 4.4 will give a detailed discussion on the selection of M based on another three numerical examples and provide an empirical instruction in selecting M.

3.5. Extension of the proposed method

Theoretically, the proposed Algorithms 1 or 2 can be extended to estimate higher-order Sobol' index. For example, the formula for

Table 1
Statistics of model inputs in the cantilever beam example.

Input	Load	Young's modulus	Poisson's ratio	Width	Height	Length
Symbol/Unit	P/kN	E/GPa	ν	b/mm	h/mm	L/mm
Distribution type	Normal	Lognormal	Lognormal	Lognormal	Lognormal	Lognormal
Mean	2.5	200	0.225	1.0	3.0	3.5
COV	0.1	0.05	0.05	0.05	0.05	0.05

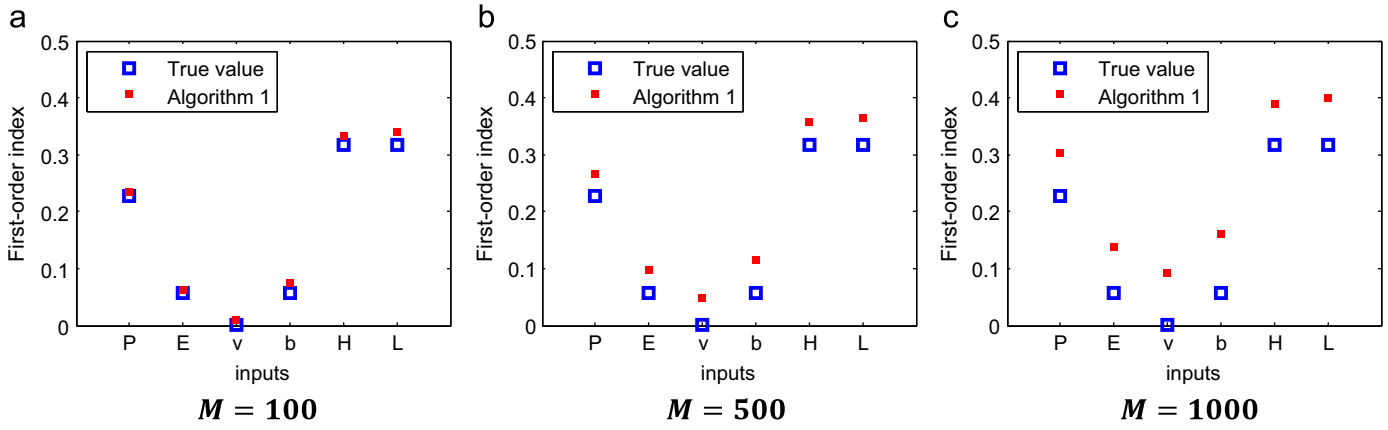

Fig. 4. Algorithm 1, Cantilever beam example.

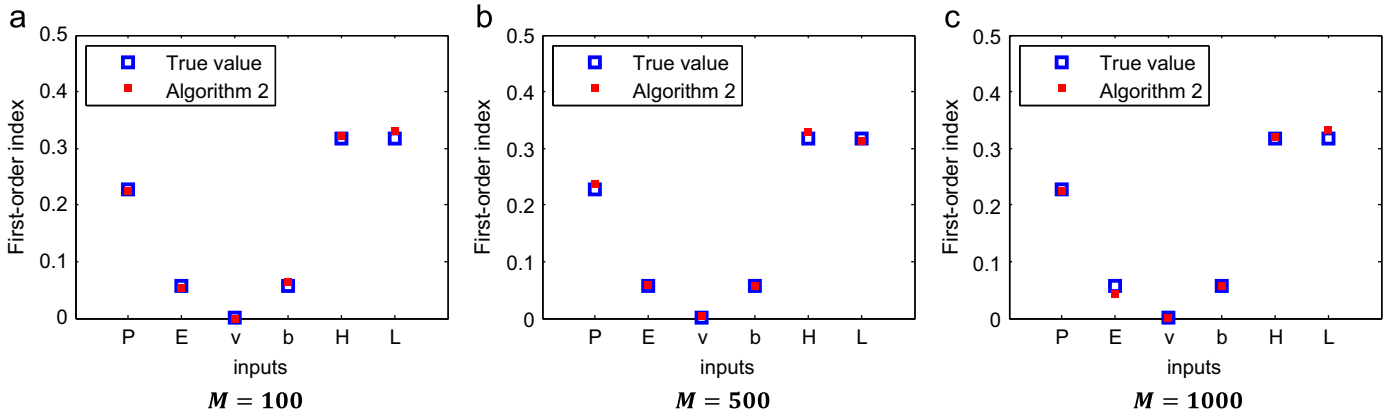

Fig. 5. Algorithm 2, Cantilever beam example.

the second-order Sobol' index of x_i and x_j $(i \neq j)$ is [1]

$$S_{ij} = \frac{V(E(y|x_i, x_j))}{V(y)} - S_i - S_j \tag{29}$$

where S_i and S_j are given by the proposed method.

Similar to Algorithm 1, $V(E(yx_i, x_j))$ in Eq. (29) can be estimated by: 1) dividing $x_i$andx_j into equally probable intervals; and 2) $V(E(yx_i, x_j)) = V_{\mathbf{\Phi}}\left(E_{\Phi^l}(y)\right)$ where $\Phi^l (l = 1\text{to}M)$ represents a 2-dimensional equally probable interval, instead of a 1-dimensional interval in calculating S_i. In general, Sobol' index of order D requires calculating $V(E(y|x_{i_1}, x_{i_2}, \ldots, x_{i_D}))$, meaning that Φ^l is an interval in a D-dimensional sampling space. The required number of intervals to fill this space, i.e., the value of M, increases with D. For a given number of samples, this means less samples in a single interval and increased numerical error in estimating $V_{\mathbf{\Phi}}\left(E_{\Phi^l}(y)\right)$. In the worst case, some intervals may not contain any sample at all. In conclusion, extending the proposed method to higher-order indices is theoretically possible, but much larger numbers of samples are needed for accurate results. Therefore, this paper only focuses on the first-order index.

3.6. Summary

Section 3 proposed two new algorithms to calculate the first-order Sobol' index. The main innovation is that the conditional mean value $V_{\boldsymbol{x}_{-i}}(yx_i)$ or the conditional variance $E_{\boldsymbol{x}_{-i}}(yx_i)$ is no more conditioned on a user-defined location but an unknown existing location of x_i. This innovation enables the proposed algorithms to directly estimate the Sobol' index from the input–output samples, and reuse the same samples to compute the indices of different input. This section also proves that Algorithm 2 is more accurate and robust than Algorithm 1. Therefore in the next section Algorithm 2 is selected to compete with existing methods. The proposed method has the following advantages:

1. Less computational effort than most existing sample-based methods in Section 2, since its computational cost is not proportional to the model input dimension.
2. Handling correlated model inputs, which is an advantage over both the existing sample-based methods and the existing analytical methods such as the M-DRM algorithm in [9].
3. Capability to compute the first-order index if input–output samples have been generated but the underlying model is not available or too expensive for re-running, and this is also an advantage over both the existing sample-based methods and the existing analytical methods. In fact, the first-order Sobol' index of x_i can be computed by the proposed method even if the samples of $\boldsymbol{x}_{-i}$ are missing.

The only parameter to be tuned in the proposed method is the number of equally probable intervals. The selection of this parameter will be discussed in Section 3.4.

Note that analytical methods such as the M-DRM algorithm [9] are more efficient and use less model evaluations than the proposed method. However, these methods need a mathematical model that connects the input to the output so that the users can run the functions at some specific values; whereas our proposed method works directly with the input–output samples, which might have been

collected from tests or field observations. As pointed out in the abstract, the main focus of this paper is to extract Sobol' index from the samples directly. Another difference is that the analytical methods need independent inputs, whereas our method is applicable also with correlated inputs.

4. Numerical examples

The objective of this section is to compare the performance of the proposed method against existing sample-based methods. The proposed Algorithm 2 is used in the comparison since Section 3.4 has shown that this algorithm is more accurate than Algorithm 1. Three numerical examples are used for comparison: 1) a low-dimensional classical non-smooth function; 2) a high-dimensional linear function; and 3) a cantilever beam problem with correlated model inputs. The comparison is conducted under the same computational cost, i.e., the same number of model evaluations.

For examples 1 and 2, the selected existing methods are 1) Sobol' method in Eq. (7), and 2) improved FAST. As a representative of the Sobol' scheme, Sobol' method in Eq. (7) is selected due to its higher accuracy than Eqs. (5) and (6), and lower computational cost than Eq. (8). The improved FAST method is selected due to its advantage of higher accuracy and lower cost than the classical FAST.

For example 3, the selected existing method is the costly double-loop MCS since other advanced methods are only suitable for independent model inputs.

Except the improved FAST, other methods (Sobol' method, double-loop MCS, and proposed method) in this section require random samples of model inputs. To achieve a comparison of best possible performance, this section uses Latin hypercube sampling to generate these random samples. Latin hypercube sampling fills the model input sampling space more evenly and improves the computational accuracy at given cost [2,29,30].

4.1. Example 1: Low-dimensional non-smooth function

The classical non-smooth function proposed by Sobol' [8] and widely used in the literature [23,32,33,40] is considered in Example 1, as

$$y=\prod_{i=1}^{k}\frac{|4x_i-2+a_i|}{a_i+1} \tag{30}$$

where $x_i (i=1\text{to}\,k)$ are independent model inputs, each following a standard uniform distribution $U(0,1)$; and $a_i (i=1\text{to}\,k)$ are user-defined constants. An analytical expression of the first-order index is available for this function:

$$V(y)=-1+\prod_{i=1}^{k}\left[\frac{1}{3(a_i+1)^2}+1\right] \quad S_i=\frac{1}{V(y)}\cdot\frac{1}{3(a_i+1)^2} \tag{31}$$

Eq. (31) indicates that a smaller value of a_i corresponds to a larger first-order index. Here we define a 4-dimentional function ($k=4$) with $a_i=i$ such that $S_1>S_2>S_3>S_4$.

Comparison of the three methods is shown in Fig. 6. The true values are based on Eq. (31). And the 95% confidence intervals for the three methods are based on 1000 runs. For each method, a single run should spend the same computational cost of model evaluations to achieve a fair comparison. The computational cost of Sobol' method is kn_s+2n_s where $k=4$ in this example and n_s is number of samples to calculate a single index S_i. Here we use $n_s=100$ thus the computational cost of the Sobol' method is 600 model evaluations. To achieve a fair comparison, we also use $n_M=600$ samples in the improved FAST method and the proposed method. In this example the number of equally probable intervals is $M=\lfloor\sqrt{n_M}\rfloor=24$, i.e., the square root of n_M rounded to the nearest integer. A detailed discussion on the selection of M can be found in Section 4.4.

Sobol' method is expected to be less accurate, since it will only use 100 samples to compute the first-order index of each individual variable, but the other two methods use all the 600 samples to compute the first-order index of each individual variable. This is confirmed by the wider confidence interval for the Sobol' method in Fig. 6.

In contrast to the Sobol' method, the improved FAST and the proposed method reduce the confidence interval by over 50%. However, the improved FAST tends to slightly overestimate the first-order indices in this example. This is probably due to the limited Fourier spectrum order ($H=6$ here). The indices estimated by the proposed method (Algorithm 2) show excellent agreement with the true values.

Note that analytical methods may solve the same problem using less functional evaluations. For instance, the M-DRM algorithm [9] can compute the Sobol' indices of a higher order ($k=8$) non-smooth function with only 81 model evaluations. The example here is to test the validity of the proposed method and prove its advantage in reducing computational cost and improving accuracy compared to other sample-based methods. Compared to analytical methods, the advantages of the proposed methods are: 1) no approximation in the model of interest; 2) calculation of Sobol' indices if input–output samples are available but the model is not; and 3) handling problems with correlated model inputs.

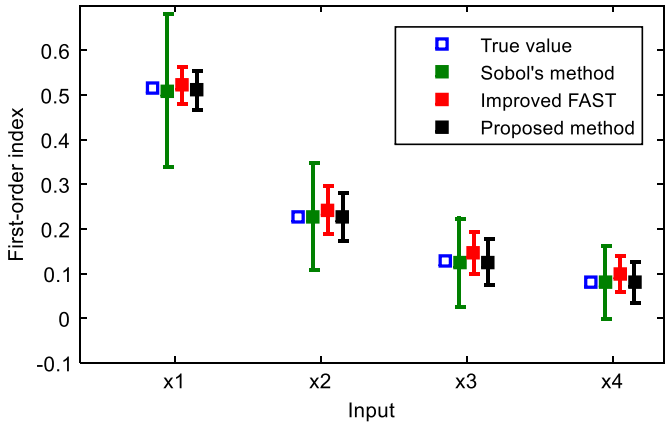

Fig. 6. First-order index of the non-smooth function.

4.2. Example 2: High-dimensional linear function

The computational cost in the improved FAST and the proposed method is not proportional to the model input dimension. This advantage is more prominent in high-dimensional problems. Consider a 50-dimensional linear function $y=\sum_{i=1}^{50} b_i x_i$ where $b_i=1+i/50$ and x_i are independent model inputs of standard normal distribution. For this example, the true value of the first-order index has analytical solution $S_i=b_i^2/\sum_{i=1}^{50} b_i^2$.

The results of the three methods are shown in Fig. 7, where the 95% confidence intervals for the three methods are based on 1000 runs. For each method, a single run should use the same computational cost of model evaluations to achieve a fair comparison. The computational cost of Sobol' method is kn_s+2n_s where $k=50$ in this example and n_s is number of samples to calculate a single index S_i. Here we use $n_s=200$ thus the computational cost of the Sobol' method is 10400 model evaluations. To achieve a fair comparison, we also use $n_M=$ 10,400 samples in the improved FAST method and the proposed method. Similar to the non-

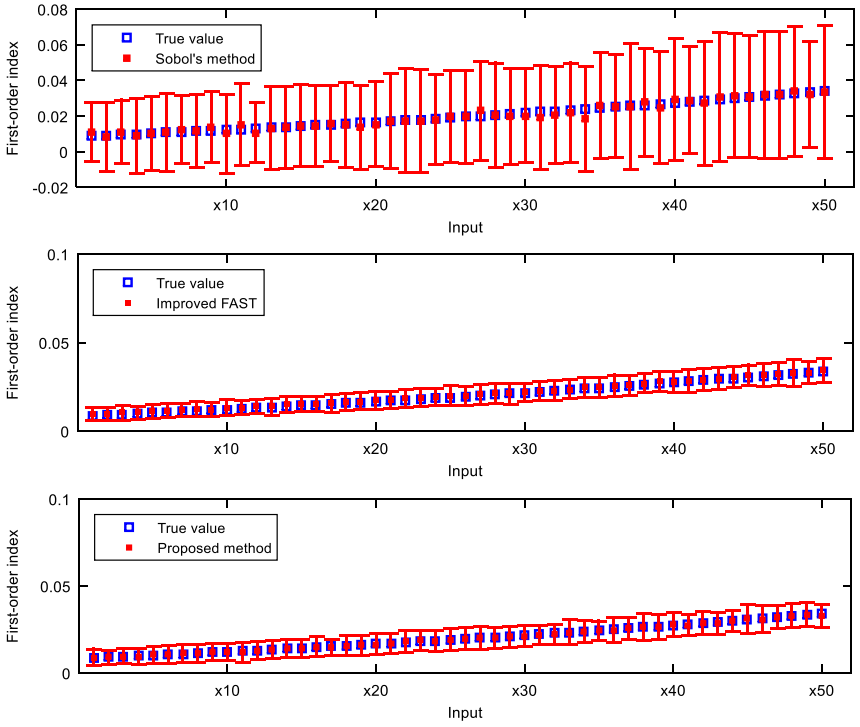

Fig. 7. First-order index of the linear function.

Table 2
Statistics of model inputs.

Model input	P/kN	E/GPa	ν	b/mm	h/mm	L/mm
Mean value	2.5	200	0.225	1.0	3	3.5
Standard deviation	0.25	20	0.0225	0.1	0.3	0.35

Table 3
Correlation matrix of model inputs.

Model input	P/kN	E/GPa	ν	b/mm	h/mm	L/mm
P/kN	1.000	0.174	0.451	0.082	−0.134	0.004
E/GPa	0.174	1.000	−0.800	0.059	−0.125	−0.082
ν	0.451	−0.800	1.000	−0.004	0.033	0.080
b/mm	0.082	0.059	−0.004	1.000	−0.105	−0.400
h/mm	−0.134	−0.125	0.033	−0.105	1.000	0.279
L/mm	0.004	−0.082	0.080	−0.400	0.279	1.000

smooth function example, the number of equally probable intervals is $M = \lfloor \sqrt{n_M} \rfloor = 102$. A detailed discussion on the selection of M can be found in Section 4.4.

In Fig. 7, the improved FAST and the proposed method show comparable performance. The improved FAST method still slightly overestimates the first-order indices in this example. The improved FAST method and the proposed method reduce the confidence interval width by around 80% in contrast to Sobol' method.

4.3. Example 3: Cantilever beam with correlated inputs

Examples 1 and 2 show that the improved FAST and the proposed method perform equally well and outperform the Sobol' method. Model inputs are independent in examples 1 and 2. However, advanced methods such as the improved FAST method are no more valid for correlated model inputs and the costly double-loop MCS is the only existing option. The proposed method provides an alternative to compute the first-order index with correlated model inputs. The example in this section illustrates this unique benefit of the proposed method. Other benefits have been discussed in Section 3.3.

Consider the cantilever beam example in Section 3.4 again. Here the model inputs are assumed to follow correlated normal distributions. Mean values and standard deviations of the model inputs are listed in Table 2 and their correlation matrix is shown in Table 3.

The results of the double-loop MCS method and the proposed method are shown in Fig. 8, where the 95% confidence intervals for

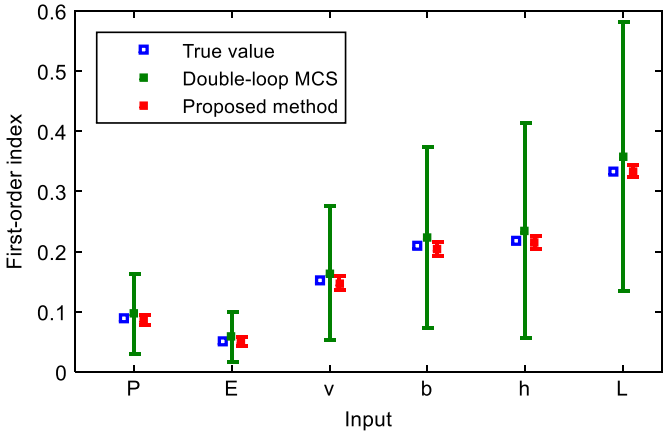

Fig. 8. First-order index of the cantilever beam example with correlated inputs.

the two methods are based on 1000 runs. For each method, a single run should use the same computational cost of model evaluations to achieve a fair comparison. The computational cost of the double-loop MCS method is $kn_{dl}^2 + n_{dl}$ where $k = 6$ in this example and n_{dl} is number of samples to calculate a single index S_i. Here we use $n_{dl} = 50$ thus the computational cost of the double-loop MCS method is 15050 model evaluations. To achieve the fair comparison, we also use $n_M = 15{,}050$ MCS samples in the

proposed method. Similar to the other two examples, the number of equally probable intervals is $M = \lceil \sqrt{n_M} \rceil = 123$. A detailed discussion on the selection of M can be found in Section 4.4.

The true values in Fig. 8 are approximated by an extreme expensive double-loop MCS with $n_{dl} = 10^4$, whose total cost is more than 6×10^8 model evaluations. Fig. 8 shows that: 1) the proposed method is very accurate for correlated model inputs; and 2) compared to the double-loop MCS, the proposed method narrows the confidence intervals by 80–95% for the same number of model evaluations.

4.4. Selection of M

At a given number of input–output samples (n_M is fixed), the only parameter to be tuned in the proposed method is M, the number of equally probably intervals. A lager M tends to improve the accuracy in the outer loop of $E_{\mathbf{\Phi}}(V_{\Phi^I}(y))$ in Eq. (23); but also reduce the accuracy in the inner loop since the average number of samples $n = n_M/M$ to compute $V_{\Phi^I}(y)$ in each individual interval will be decreased. Therefore a tradeoff between M and n is to be decided. This section aims to compare different selections of M using the three numerical examples above and provides an heuristic guideline in selecting M. This discussion constitutes of the following steps:

1. Set different values of M. The medium value is $M = \lceil \sqrt{n_M} \rceil$ to achieve a balance of $M = n$; the lowest values is 5 meaning only 5 intervals; and the highest value is $M = n_M/5$ meaning only around 5 samples in each interval.
2. Calculate the confidence intervals (CI) of the first-order indices at different values of M.
3. Compare the accuracy at different values of M based on the location and width of the CIs.

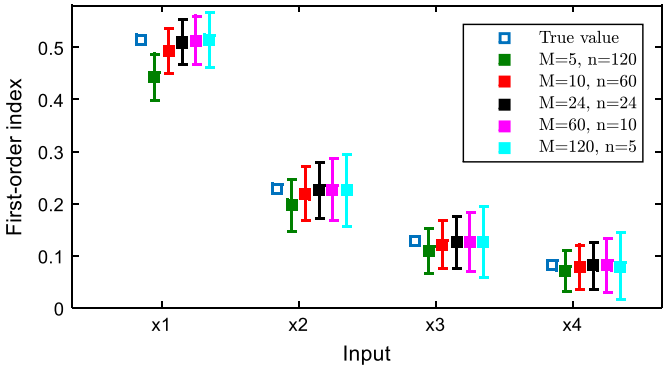

Fig. 9. Selection of M in the non-smooth function example.

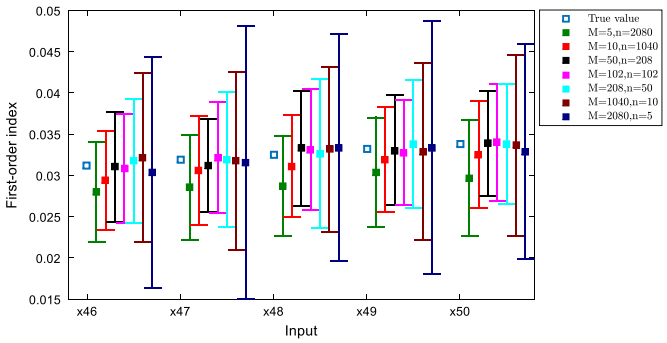

Fig. 10. Selection of M in high-dimensional linear function example.

The comparison for the non-smooth function example is shown in Fig. 9. As explained in Section 4.1, the total number of input–output samples is $n_M = 600$. As shown in the legend of Fig. 9, 5 values of M are used where the medium value is $M = \lceil \sqrt{n_M} \rceil = 24$. Fig. 9 indicates that: 1) the result by $M = 5\&n = 120$ is biased from the true value, especially for x_1; 2) the result by $M = 120\&n = 5$ has wider CIs than others; and 3) the results by other values of M are comparable good. In sum, this example requires $M \geq 10$ and $n \geq 10$.

The comparison for the high-dimensional linear function example is shown in Fig. 10, and only the last five inputs are included due to limited space. As explained in Section 4.2, the total number of input–output samples is $n_M = 10400$. As shown in the legend of Fig. 10, 7 values of M are used where the medium value is $M = \lceil \sqrt{n_M} \rceil = 102$. Fig. 10 indicates that: 1) the result by $M = 5$ is biased from the true value significantly; 2) the result by $M = 10$ is biased slightly but still acceptable; 3) the results by $n = 5$ have significant wider CIs; 4) the result by $n = 10$ also have wider CIs but still acceptable; and 5) the results by other values of M are comparable. In sum, this example requires $M \geq 10\&n \geq 10$ but $M \geq 50\&n \geq 50$ is recommended.

The comparison for the cantilever beam with correlated inputs is shown in Fig. 11, and only the inputs with first-order index larger than 0.1 is listed. As explained in Section 4.3, the total number of input–output samples is $n_M = 15050$. As shown in the legend of Fig. 11, 7 values of M are used where the medium value is $M = \lceil \sqrt{n_M} \rceil = 123$. Fig. 11 indicates that: 1) the results by $M = 5, 10$ or $n = 5, 10$ are biased from the true value, especially for L; and 2) the results by other values of M are comparable, and the result by $M = n = 123$ is slightly better. In sum, this example requires $M \geq 50$ and $n \geq 50$.

Based on the comparisons above, the authors conclude that the minimum requirement for the proposed algorithm is $M \geq 10\&n \geq 10$; but $M \geq 50\&n \geq 50$ is recommended. Actually a simple strategy is $M = \lceil \sqrt{n_M} \rceil$ to achieve a balance of $M = n$, and this strategy has been used in all the examples in this paper. Note that this guidance is purely heuristic, and formally optimizing the value of M may be explored in future.

4.5. Example 4: Input–output function NOT available

The proposed algorithm can estimate the first-order Sobol' index as long as the input–output samples have already been collected, even if the underlying function is NOT available or cannot be re-evaluated. This situation often happens in the industry when an analyst supplies only the input–output data, but does not provide the computational model due to proprietary

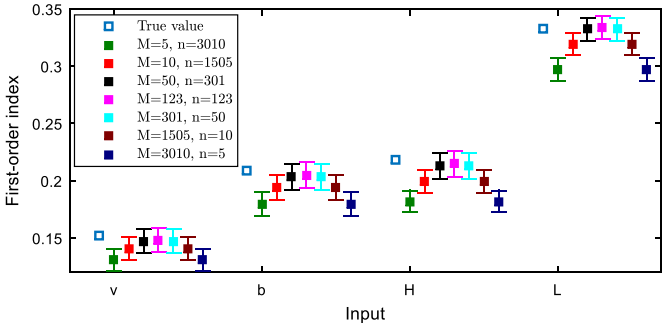

Fig. 11. Selection of M in the cantilever beam example.

Table 4
First-order Sobol' index.

Variable	x_1	x_2	x_3
First-order index	0.42	0.23	0.00

reasons. The reason that we used computational models in the earlier examples was to be able to compare the accuracy of our method with existing methods. Here we demonstrate the case of sensitivity analysis with only input–output data, assuming the computational model is not available.

We generated 2500 Monte Carlo input–output samples using the Ishigami function $Y = \sin x_1 + a \sin^2 x_2 + b x_3^4 \sin x_1$ [10]. These samples can be downloaded via the URL https://github.com/VandyChris/Global-Sensitivity-Analysis/blob/master/Ishigami.csv.

Now suppose that only these samples are available, and that the actual function is not available, and even the distributions of $x_i (i = 1, 2, 3)$ are not known. In this situation, the proposed algorithm can directly estimate the first-order Sobol' indices from the available samples following the steps in Fig. 2, but the existing algorithms discussed in Section 2 cannot. Using the proposed Algorithm 2 and $M = 50$, the result is obtained as shown in Table 4. (Of course, if the function is known, then it is possible to verify the accuracy of this result. We have verified that the result using the above analytical function is exactly the same. However, the purpose of this example is to demonstrate that the proposed method can calculate the first-order Sobol' indices using only the input–output samples. An alternative approach is to build a regression model based on the samples, and then use the regression model for GSA using any of the other existing methods; in that case, the regression error should also be accounted for).

5. Summary

This paper is particularly interested in directly extracting Sobol' index from Monte Carlo samples. To solve this problem, this paper showed that the conditional variance and mean in the expression of the first-order Sobol' index can be computed at an unknown but existing location of model inputs, instead of an explicit user-defined location. This concept leads to the proposed method which is modularized in two aspects: 1) separate the index calculations for different model inputs; and 2) model inputs sampling, model evaluation, and index calculation are separate processes. The modularization brings several benefits: 1) The computational cost of the proposed method is not proportional to the number of model inputs; 2) The proposed method can be used when only legacy input–output data or field data are available but the underlying model is not available, which is our main focus; 3) The calculation of S_i purely depends on the samples of x_i and y, and can be achieved even if the samples of $\boldsymbol{x}_{-i}$ are missing; and most importantly 4) The proposed method is able to compute the first-order index with correlated model inputs.

The proposed method includes two algorithms. Algorithm 1 computes the inner loop $E_{\boldsymbol{x}_{-i}}(y|x_i)$ first and then the outer loop $V_{x_i}(\bullet)$; while Algorithm 2 computes the inner loop $V_{\boldsymbol{x}_{-i}}(y|x_i)$ first and the then outer loop $E_{x_i}(\bullet)$. Section 3.4 proves that Algorithm 2 provides higher accuracy while Algorithm 1 is positively biased due to numerical error.

Algorithm 2 is used in two numerical examples with independent model inputs to compare with two existing methods: 1) the widely used Sobol' method, and 2) the improved FAST method. The latter one also has a computational cost that is not proportional to the model inputs dimension; and it is the best previously available algorithm for independent model inputs to the authors' knowledge. The results show that the proposed method has comparable accuracy with improved FAST and outperforms the Sobol' method. Algorithm 2 is also used in a third numerical example with correlated model inputs and seen to significantly outperform the double-loop MCS method; the improved FAST method cannot handle correlated model inputs.

The benefits brought by the proposed method imply strong promise for practical implementation such as test design [41,42], dimension reduction, feature selection, etc. Nowadays in areas such as transportation and social networks, obtaining data can be much easier than extracting the underlying models. Since the proposed method is highly efficient and only requires data, it is especially useful in ranking and identifying important variables, no matter whether the variables are correlated or not.

Acknowledgments

The research in this paper is partially supported by funds from Sandia National Laboratories through Contract no. BG-7732 (Technical Monitor: Dr. Angel Urbina). This support is gratefully acknowledged. The authors also thank Saideep Nannapaneni and Dr. Zhen Hu at Vanderbilt University for valuable discussions.

Appendix

Extreme value theorem: A continuous real-valued function f on a non-empty compact space must attain a maximum and a minimum.

Eq. (12) applies this theorem to the function $f: x_i \to E_{\boldsymbol{x}_{-i}}(y|x_i)$ in a nonempty compact space Φ.

Intermediate value theorem: Let $f: X \to Y$ be a continuous function where X is a connected space. If a and b are two points in X and u is a point in Y lying between $f(a)$ and $f(b)$, then there exists c in X such that $u = f(c)$.

Eq. (12) has proved that $E(y)$ is between the minimum and the maximum of the function $f: x_1 \to E_{\boldsymbol{x}_{-i}}(y|x_i)$ where $x_i \in \Phi$. Eq. (13) applies the intermediate value theorem to $f: x_i \to E_{\boldsymbol{x}_{-i}}(y|x_i)$ to prove the existence of x_i^* in Φ such that $E_{\Phi}(y) = E_{\boldsymbol{x}_{-i}}(y|x_i^*)$.

References

[1] Saltelli A, Ratto M, Andres T, Campolongo F, Cariboni J, Gatelli D, et al. Global sensitivity analysis: the primer. John Wiley & Sons; 2008.
[2] Helton JC, Johnson JD, Sallaberry CJ, Storlie CB. Survey of sampling-based methods for uncertainty and sensitivity analysis. Reliab Eng Syst Saf 2006;91:1175–209. http://dx.doi.org/10.1016/j.ress.2005.11.017.
[3] Hu Z, Du X. Mixed efficient global optimization for time-dependent reliability analysis. J Mech Des 2015;137:051401. http://dx.doi.org/10.1115/1.4029520.
[4] Hu Z, Du X. Time-dependent reliability analysis with joint upcrossing rates. Struct Multidiscip Optim 2013;48:893–907. http://dx.doi.org/10.1007/s00158-013-0937-2.
[5] Li C, Mahadevan S. Robust test resource allocation using global sensitivity analysis. In: Proceedings of the 18th AIAA non-deterministic approaches conference, Reston (Virginia): American Institute of Aeronautics and Astronautics; 2016. doi: ⟨http://dx.doi.org/10.2514/6.2016-0952⟩.
[6] Li C, Mahadevan S. Relative contributions of aleatory and epistemic uncertainty sources in time series prediction. Int J Fatigue 2016;82:474–86. http://dx.doi.org/10.1016/j.ijfatigue.2015.09.002.
[7] Nannapaneni S, Mahadevan S. Uncertainty quantification in performance evaluation of manufacturing processes. In: Proceedings of the 2014 IEEE international conference on big data (Big Data), IEEE; 2014. p. 996–1005. doi: ⟨http://dx.doi.org/10.1109/BigData.2014.7004333⟩.
[8] Sobol' IM. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math Comput Simul 2001;55:271–80. http://dx.doi.org/10.1016/S0378-4754(00)00270-6.
[9] Zhang X, Pandey MD. An effective approximation for variance-based global sensitivity analysis. Reliab Eng Syst Saf 2014;121:164–74. http://dx.doi.org/10.1016/j.ress.2013.07.010.
[10] Sudret B. Global sensitivity analysis using polynomial chaos expansions. Reliab Eng Syst Saf 2008;93:964–79. http://dx.doi.org/10.1016/j.ress.2007.04.002.
[11] Chen W, Jin R, Sudjianto A. Analytical variance-based global sensitivity analysis in simulation-based design under uncertainty. J Mech Des 2005;127:875. http://dx.doi.org/10.1115/1.1904642.
[12] Xu P, Su X, Mahadevan S, Li C, Deng Y. A non-parametric method to determine basic probability assignment for classification problems. Appl Intell 2014. http://dx.doi.org/10.1007/s10489-014-0546-9.
[13] Friedman JH. Multivariate adaptive regression splines. Ann Stat 1991;19:1–67.

[14] Mara T, Joseph O. Comparison of some efficient methods to evaluate the main effect of computer model factors. J Stat Comput Simul 2008;78:167–78. http://dx.doi.org/10.1080/10629360600964454.
[15] Li C, Mahadevan S. Uncertainty quantification and output prediction in multi-level problems. In: Proceedings of the 16th AIAA non-deterministic approaches conference, Reston (Virginia): American Institute of Aeronautics and Astronautics; 2014. doi: ⟨http://dx.doi.org/10.2514/6.2014-0124⟩.
[16] Tissot J, Prieur C. A randomized Orthogonal Array-based procedure for the estimation of first- and second-order Sobol' indices. J Stat Comput Simul 2014:1–24.
[17] Li C, Mahadevan S. Global sensitivity analysis for system response prediction using auxiliary variable method. In: Proceedings of the 17th AIAA Non-deterministic approaches conference, Reston (Virginia): American Institute of Aeronautics and Astronautics; 2015. doi: ⟨http://dx.doi.org/10.2514/6.2015-0661⟩.
[18] Li C, Mahadevan S. Role of calibration, validation, and relevance in multi-level uncertainty integration. Reliab Eng Syst Saf 2016;148:32–43. http://dx.doi.org/10.1016/j.ress.2015.11.013.
[19] Saltelli A, Tarantola S. On the relative importance of input factors in mathematical models. J Am Stat Assoc 2002;97:702–9. http://dx.doi.org/10.1198/016214502388618447.
[20] Saltelli A. Making best use of model evaluations to compute sensitivity indices. Comput Phys Commun 2002;145:280–97. http://dx.doi.org/10.1016/S0010-4655(02)00280-1.
[21] Homma T, Saltelli A. Importance measures in global sensitivity analysis of nonlinear models. Reliab Eng Syst Saf 1996;52:1–17. http://dx.doi.org/10.1016/0951-8320(96)00002-6.
[22] Sobol' IM, Tarantola S, Gatelli D, Kucherenko SS, Mauntz W. Estimating the approximation error when fixing unessential factors in global sensitivity analysis. Reliab Eng Syst Saf 2007;92:957–60. http://dx.doi.org/10.1016/j.ress.2006.07.001.
[23] Sobol' IM, Myshetskaya EE. Monte Carlo estimators for small sensitivity indices. Monte Carlo Methods Appl 2008;13:455–65.
[24] Owen A. Better estimation of small sobol sensitivity indices. ACM Trans Model Comput Simul 2013;23:11. http://dx.doi.org/10.1145/2457459.2457460.
[25] Wu Q-L, Cournède P-H, Mathieu A. An efficient computational method for global sensitivity analysis and its application to tree growth modelling. Reliab Eng Syst Saf 2012;107:35–43. http://dx.doi.org/10.1016/j.ress.2011.07.001.
[26] Janon A, Klein T, Lagnoux AA, Nodet M, Prieur C. Asymptotic normality and efficiency of two Sobol index estimators. ESAIM Probab Stat 2014;18:342–64.
[27] Weirs VG, Kamm JR, Swiler LP, Tarantola S, Ratto M, Adams BM, et al. Sensitivity analysis techniques applied to a system of hyperbolic conservation laws. Reliab Eng Syst Saf 2012;107:157–70. http://dx.doi.org/10.1016/j.ress.2011.12.008.
[28] Saltelli A, Tarantola S, Chan K. A quantitative model-independent method for global sensitivity analysis of model output. Technometrics 1999;41:39–56. http://dx.doi.org/10.1080/00401706.1999.10485594.
[29] Tarantola S, Koda M. Random balance designs for the estimation of first order global sensitivity indices. Procedia – Soc Behav Sci 2006;2:7753–4. http://dx.doi.org/10.1016/j.sbspro.2010.05.212.
[30] Satterthwaite FE. Random Balance Experimentation. Technometrics 1959.
[31] Cukier RI, Levine HB, Shuler KE. Nonlinear sensitivity analysis of multiparameter model systems. J Comput Phys 1978;26:1–42. http://dx.doi.org/10.1016/0021-9991(78)90097-9.
[32] Cukier RI. Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficients. I Theory. J Chem Phys 1973;59:3873. http://dx.doi.org/10.1063/1.1680571.
[33] Koda M, Mcrae GJ, Seinfeld JH. Automatic sensitivity analysis of kinetic mechanisms. Int J Chem Kinet 1979;11:427–44. http://dx.doi.org/10.1002/kin.550110408.
[34] Hu Z, Mahadevan S. Global sensitivity analysis-enhanced surrogate (GSAS) modeling for reliability analysis. Struct Multidiscip Optim 2015. http://dx.doi.org/10.1007/s00158-015-1347-4.
[35] Ginot V, Gaba S, Beaudouin R, Aries F, Monod H. Combined use of local and ANOVA-based global sensitivity analyses for the investigation of a stochastic dynamic model: Application to the case study of an individual-based model of a fish population. Ecol Model 2006;193:479–91. http://dx.doi.org/10.1016/j.ecolmodel.2005.08.025.
[36] Archer G, Saltelli A, Sobol' IM. Sensitivity measures, ANOVA-like techniques and the use of bootstrap. J Stat Comput Simul 1997;58:99–120.
[37] Weiss NA, Holmes PT, Hardy M. A course in probability. Pearson Addison Wesley; 2006.
[38] Keisler HJ. Elementary calculus: an infinitesimal approach. Courier Corp. 2012.
[39] Timoshenko SP, Goodier JN. Theory of Elasticity. 3rd ed. New York: McGraw-Hill; http://dx.doi.org/10.1115/1.3408648.
[40] Mullins J, Li C, Sankararaman S, Mahadevan S, Urbina A. Probabilistic integration of validation and calibration results for prediction level uncertainty quantification: application to structural dynamics. In: Proceedings of the 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Material Conference, Reston (Virginia: American Institute of Aeronautics and Astronautics; 2013. doi: ⟨http://dx.doi.org/10.2514/6.2013-1872⟩.
[41] Mullins J, Li C, Mahadevan S, Urbina A. Optimal selection of calibration and validation test samples under uncertainty. In: Model validation and uncertainity quantification, 3. Springer International Publishing; 2014. p. 391–401. http://dx.doi.org/10.1007/978-3-319-04552-8_39 SE - 39.
[42] Li C, Mahadevan S. Sensitivity analysis for test resource allocation. In: Model validation uncertainity and quantification, 3. Springer International Publishing; 2015. p. 143–50. http://dx.doi.org/10.1007/978-3-319-15224-0_14 SE - 14.