International Journal of Fatigue 82 (2016) 474–486

Contents lists available at ScienceDirect

International Journal of Fatigue

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

Relative contributions of aleatory and epistemic uncertainty sources in time series prediction

Chenzhao Li, Sankaran Mahadevan *

Department of Civil and Environmental Engineering, Vanderbilt University, Nashville, TN 37235, USA

ARTICLE INFO

Article history:
Received 9 February 2015
Received in revised form 31 August 2015
Accepted 1 September 2015
Available online 21 September 2015

Keywords:
Epistemic uncertainty
Sobol indices
Time series
Autoregressive moving average
Fatigue

ABSTRACT

This paper develops a novel computational framework to compute the Sobol indices that quantify the relative contributions of various uncertainty sources towards the system response prediction uncertainty. In the presence of both aleatory and epistemic uncertainty, two challenges are addressed in this paper for the model-based computation of the Sobol indices: due to data uncertainty, input distributions are not precisely known; and due to model uncertainty, the model output is uncertain even for a fixed realization of the input. An auxiliary variable method based on the probability integral transform is introduced to distinguish and represent each uncertainty source explicitly, whether aleatory or epistemic. The auxiliary variables facilitate building a deterministic relationship between the uncertainty sources and the output, which is needed in the Sobol indices computation. The proposed framework is developed for two types of model inputs: random variable input and time series input. A Bayesian autoregressive moving average (ARMA) approach is chosen to model the time series input due to its capability to represent both natural variability and epistemic uncertainty due to limited data. A novel controlled-seed computational technique based on pseudo-random number generation is proposed to efficiently represent the natural variability in the time series input. This controlled-seed method significantly accelerates the Sobol indices computation under time series input, and makes it computationally affordable.

1. Introduction

In many practical engineering systems, direct measurement of the system response under actual usage conditions is often not available; instead a model is used to predict the response, in order to facilitate decisions related to design, risk management etc. In this case, the uncertainty in system response prediction is affected by various uncertainty sources. The importance of each uncertainty source can be measured by its contribution to the uncertainty in the system response prediction. Such information is useful in several ways, especially in problem size reduction (by ignoring the insignificant uncertainty sources) and in resource allocation for uncertainty reduction (by focusing additional data collection or model refinement efforts on significant uncertainty sources).

Global sensitivity analysis (GSA) [1] provides a quantitative assessment of the relative contribution of model inputs towards the uncertainty in the model output. GSA methods can be either data-driven (e.g., based on analysis of variance ANOVA), or model-based, such as the computation of Sobol indices [2]. In model-based prediction as shown in Fig. 1, the computation of Sobol indices is well-established for aleatory inputs [3–5], but their computation considering both aleatory and epistemic uncertainty sources (in model inputs and in model prediction) is not well-established [6–8]. Thus this paper focuses on developing a framework for computing the Sobol indices considering both aleatory and epistemic uncertainty sources, when considering uncertainty propagation through a computational model.

Related to Fig. 1, there is uncertainty in the model inputs, and in the model output even for a fixed input. A model input may be deterministic or random, and epistemic uncertainty can be present in both, due to inadequate data. In case of a deterministic input, its value may be unknown; in the case of a random input, its distribution type and/or distribution parameters may be unknown. The latter case is a mixture of aleatory and epistemic uncertainty. When the input is propagated through the computational model to compute the output, epistemic uncertainty sources in the model (uncertain model parameters, numerical approximations in the model, and model form assumptions) contribute to additional uncertainty in the model prediction. The objective of this paper is quantify the contributions of various aleatory and epistemic uncertainty sources in the input and the model to the uncertainty in the model output.

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

http://dx.doi.org/10.1016/j.ijfatigue.2015.09.002
0142-1123/

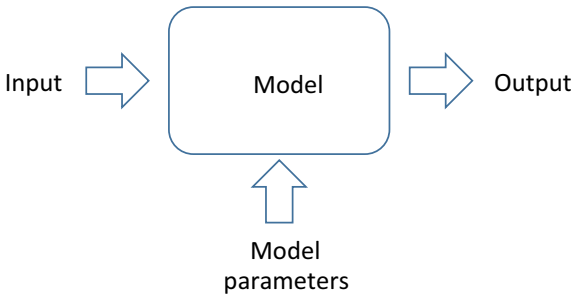

Fig. 1. Model-based prediction.

The proposed framework for realizing this objective is shown in Fig. 2. Due to inadequate data, the aleatory model inputs (either random variables or random processes) are mixed with epistemic uncertainty. Due to model uncertainty sources, the model output is uncertain even for a fixed input. When using an input–output model to compute the Sobol indices, a deterministic input–output relationship, i.e., a one-to-one mapping, is needed. Therefore, a methodology is proposed in this paper by introducing auxiliary variables based on the probability integral transform (explained in Section 2) to establish such a deterministic input–output relationship, and to separate the aleatory and epistemic uncertainty sources when the two are mixed. This strategy helps to calculate the Sobol indices separately for both aleatory and epistemic uncertainty sources.

A particular problem of interest in this paper is when the input to the computational model is a time series, such as the loading history on a mechanical component causing fatigue damage. Several options are available for modeling the time series input; this paper uses the autoregressive moving average (ARMA) approach, which is able to explicitly quantify the aleatory and epistemic uncertainty components in the time series input through the use of Bayesian calibration. The ARMA model and Bayesian calibration are described in Section 2. Sensitivity computation in the presence of time series input brings a significant challenge regarding computational effort, especially due to the introduction of a large number of noise terms (one in each time step). Therefore this paper proposes a novel technique, based on the concept of pseudo-random number generation, to significantly improve the computational efficiency in calculating the Sobol indices in the presence of time series input that has both aleatory and epistemic uncertainty.

In summary, this paper makes three new important contributions to model-based sensitivity analysis: (1) computation of Sobol indices in the presence of both input uncertainty (aleatory and epistemic) and model uncertainty (epistemic); (2) a novel technique to separate the aleatory and epistemic uncertainty sources in time series input; and (3) a novel computational technique (based on pseudo-random number generation) for efficient computation of Sobol indices in the presence of time series input.

The rest of the paper is organized as follows. Section 2 provides background information on related methods. Section 3 explains the proposed methodology. Section 4 illustrates the proposed methodology with a numerical example of fatigue crack growth in a cantilever beam subjected to time series loading, and Section 5 provides concluding remarks.

2. Background

The background concepts used in developing the proposed methodology in this paper can be summarized as: (1) computation of Sobol indices to quantify the contributions of different uncertainty sources to the uncertainty in the output of the prediction model; (2) an auxiliary variable approach to decouple the dependent uncertainty sources and establish the deterministic function required by Sobol indices computation, (3) ARMA approach to model the time series input in the prediction model and Bayesian calibration of the ARMA parameters; and (4) use of sampling to calculate the Sobol indices. These concepts are integrated for the objective of this paper, which is to quantify the contribution of each uncertainty source towards the system response prediction. This section provides brief introduction to these four key concepts: uncertainty sources, ARMA model, Sobol indices, and auxiliary variable.

2.1. Sobol indices

Sobol indices assess the contribution of each input uncertainty to the output uncertainty, and are based on the variance decomposition theorem [2]. A deterministic function [23,24] $Y = F(\boldsymbol{X})$ where $\boldsymbol{X} = \{X_1, \ldots, X_k\}$ is a vector containing all the model inputs is required for the computation in the context of model-based prediction (Note that other GSA methods may not require a deterministic function, for example, the GSA method [25] based on the classical ANOVA using factorial design of experiments can handle both deterministic and stochastic models). Here the model inputs are random variables. The function is deterministic if a single realization of $\boldsymbol{X}$ gives a corresponding single realization of Y. This paper emphasize the term "deterministic function" to contrast from a "stochastic" prediction $Y = S(\boldsymbol{\theta}_m; \boldsymbol{X}) + \epsilon_h(\boldsymbol{X}) + \delta(\boldsymbol{X})$ in Section 1, where the function output is uncertain (i.e., it has many possible realizations) even if all the inputs are fixed. The first-order sensitivity index is defined as:

$$S_j = \frac{V(E(Y|X_j))}{V(Y)} \tag{1}$$

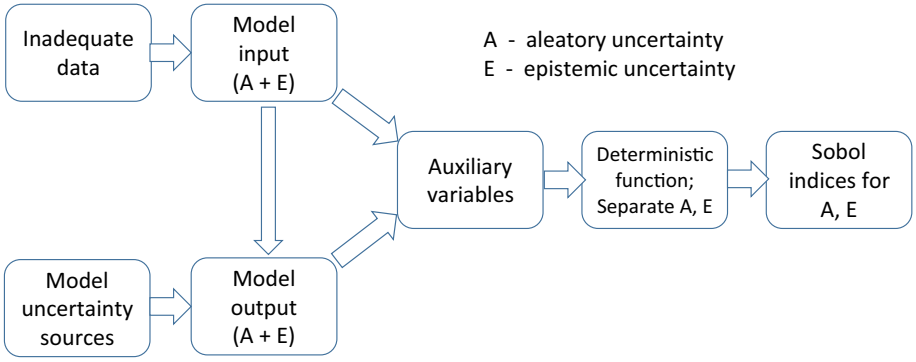

Fig. 2. Proposed framework for Sobol indices computation under aleatory and epistemic uncertainty.

S_j is a number between 0 and 1 since $V(Y) \geqslant V(E(Y|X_j)) > 0$, and a higher value of S_j indicates greater importance of X_j. The first-order index measures the individual contribution of X_j by itself. The overall contribution of X_j by itself plus interactions with other inputs is measured by the total effects index S_j^T:

$$S_j^T = 1 - \frac{V(E(Y|X_{-j}))}{V(Y)} \tag{2}$$

where X_{-j} denote all the model inputs other than X_j.

Sobol's decomposition requires both a deterministic function and uncorrelated model inputs, so the derived total effects index S_j^T is only meaningful for uncorrelated model inputs [26]. In contrast, the first-order index S_j in Eq. (1) only requires a deterministic function, so that S_j can be calculated whether the model inputs are correlated or uncorrelated [26].

2.2. Uncertainty sources in system response prediction

Consider a computational model $Y = F(\boldsymbol{\theta}_m; \boldsymbol{X})$, where $\boldsymbol{\theta}_m$ is a set of fixed but unknown model parameters, and $\boldsymbol{X}$ is the vector of model inputs. Often there is discrepancy between the model prediction $F(\boldsymbol{\theta}_m; \boldsymbol{X})$ and the true value of the output, due to two types of errors [9,10]: (1) the numerical errors in solving the mathematical model (such as discretization, truncation and round-off errors); and (2) model form error. Often the estimates of these errors are also uncertain (epistemic); therefore the model output is uncertain. For example, if the mathematical model is a differential equation and $F(\boldsymbol{\theta}_m; \boldsymbol{X})$ solves it using numerical discretization (e.g., finite element, finite difference), then the discretization error $\epsilon_h(\boldsymbol{X})$ at a given model input is deterministic for a given value of the input [11]; however some implementations of computing ϵ_h use Gaussian process (GP) models [12,13] to capture the uncertainty in estimating ϵ_h. The model discrepancy $\delta(\boldsymbol{X})$ is the difference between the computational model and the real system. Kennedy and O'Hagan [14] represent it by a GP model, so the correction of the model discrepancy is also stochastic at a given model input. In some studies the discrepancy term $\delta(\boldsymbol{X})$ includes the numerical errors, and in some studies it refers only to model form error, after accounting for numerical errors that are estimated separately.

In addition, to promote computational efficiency, often the computational model is replaced by a surrogate model $S(\boldsymbol{\theta}_m; \boldsymbol{X})$. This surrogate model brings additional uncertainty in the prediction, due to limited training points. This paper uses the GP surrogate model [15]. The output of the GP model is a Gaussian distribution, which represents the surrogate model prediction uncertainty.

Considering the discretization error, model form error, and surrogate model uncertainty, a general expression of the corrected system response prediction may be written as $Y = S(\boldsymbol{\theta}_m; \boldsymbol{X}) + \epsilon_h(\boldsymbol{X}) + \delta(\boldsymbol{X})$. The prediction Y is stochastic due to the uncertainty in the three terms on the right hand side, even at a fixed value of $\boldsymbol{\theta}_m$ and $\boldsymbol{X}$.

Extra uncertainty sources arise in characterizing $\boldsymbol{\theta}_m$ and $\boldsymbol{X}$. The model parameters $\boldsymbol{\theta}_m$ have fixed but unknown values. Usually, limited data are available to calibrate $\boldsymbol{\theta}_m$, thus there is epistemic uncertainty (lack of knowledge) regarding $\boldsymbol{\theta}_m$.

If a model input X is a random variable, its natural variability can be represented by a probability distribution with distribution parameters $\boldsymbol{\theta}_X$. If only limited observations of X are available, there is uncertainty in the distribution type and distribution parameters. This uncertainty is also referred as statistical uncertainty [16] or second-order uncertainty [17]. Therefore, the uncertainty in model input X has two components: aleatory natural variability and epistemic uncertainty regarding distribution type and distribution parameters.

If a model input X is not a random variable but a time series, the prediction of system response Y requires the values of X over all time steps. Two types of time domain methods have been developed to model the time series input using observed data: (1) cycle counting methods, including the rainflow counting method [18] and the Markov chain method [19]; and (2) random process methods, such as the autoregressive moving average (ARMA) model [20]. This paper uses the ARMA model and therefore a brief introduction to ARMA is given in Section 2.4.

The uncertainty sources discussed above are listed in Table 1. The natural variability in the model input is aleatory; all other sources are epistemic.

This paper develops a methodology to calculate the individual contribution of each uncertainty source in Table 1 to the uncertainty in the system response prediction. The auxiliary variable method is used to incorporate all the uncertainty sources into a deterministic function, thus to assess the contribution of each uncertainty source based on the Sobol indices.

2.3. Auxiliary variable method

The auxiliary variable method was developed by Sankararaman and Mahadevan [27] to distinguish the contributions of aleatory natural variability and epistemic distribution parameter uncertainty in a random variable X. The distribution of X is conditioned on the value of its distribution parameter $\boldsymbol{\theta}_X$, which has uncertainty represented by a probability density $f(\boldsymbol{\theta}_X)$. This parameters distribution $f(\boldsymbol{\theta}_X)$ is also referred as second-order probability. The conditional distribution of X is denoted as $f_X(x|\boldsymbol{\theta}_X = \boldsymbol{\theta}_X^*)$. This conditional distribution $f_X(x|\boldsymbol{\theta}_X = \boldsymbol{\theta}_X^*)$ and the second-order probability $f(\boldsymbol{\theta}_X)$ actually constitutes a hierarchical Bayesian model. With different realizations of $\boldsymbol{\theta}_X, f_X(x|\boldsymbol{\theta}_X = \boldsymbol{\theta}_X^*)$ constitutes a family of distributions, as shown in Fig. 3. Each single distribution represents the natural variability of X at a particular realization of $\boldsymbol{\theta}_X$, and the spread of the distributions indicates the contribution of uncertainty in $\boldsymbol{\theta}_X$. This family of distribution only gives a qualitative representation of aleatory and epistemic

Table 1
Uncertainty sources in system response prediction.

Uncertainty type	Symbol	Uncertainty source	Category
Solution approximation	$S(\boldsymbol{\theta}_m; \boldsymbol{X})$	Surrogate model	Epistemic
Solution approximation	$\epsilon_h(\boldsymbol{X})$	Discretization error	Epistemic
Model form error	$\delta(\boldsymbol{X})$	Model discrepancy	Epistemic
Model parameter	$\boldsymbol{\theta}_m$	Model parameter uncertainty	Epistemic
Random variable input	$\boldsymbol{\theta}_X$	Distribution parameter uncertainty	Epistemic
	X given $\boldsymbol{\theta}_X$	Input natural variability	Aleatory
Time series input by ARMA model	$\bar{X}$, $\boldsymbol{\phi}$, $\boldsymbol{\theta}$, σ_ϵ	Model parameter uncertainty	Epistemic
	ϵ_t	Input natural variability	Aleatory

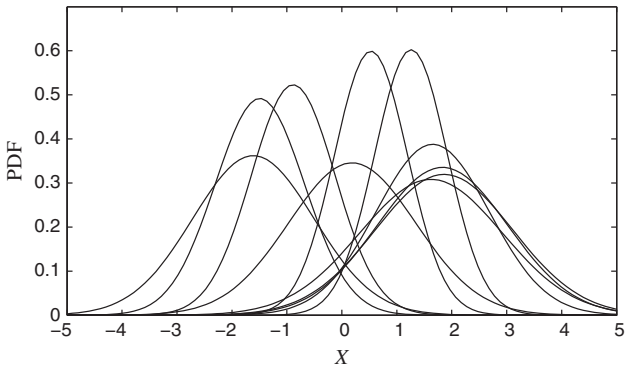

Fig. 3. Family of PDFs.

uncertainties; a method of quantitative contribution assessment is still required.

Based on the probability integral transform theorem [28], random sampling from the conditional distribution $f_X(x|\boldsymbol{\theta}_X = \boldsymbol{\theta}_X^*)$ is realized in two steps: (1) define a variable U_X of standard uniform distribution $U(0,1)$ and generate its sample u_X, which is taken as the CDF value of X, and (2) obtain a sample of X by inverting the conditional CDF $\mathcal{F}_X(x|\boldsymbol{\theta}_X = \boldsymbol{\theta}_X^*)$, i.e.,

$$x = \mathcal{F}_X^{-1}(u_X|\boldsymbol{\theta}_X = \boldsymbol{\theta}_X^*) \tag{3}$$

The same procedure is repeated for other realizations of $\boldsymbol{\theta}_X$. Note that the distribution of U_X is independent of the realization of $\boldsymbol{\theta}_X$. At a given value of $\boldsymbol{\theta}_X$, the sample of U_X and the sample of X have a one-to-one mapping, i.e., a single value of X is determined once the value of U_X is decided. Thus the natural variability in X is represented by U_X.

This standard uniform random variable U_X, which is the CDF value of $f_X(x|\boldsymbol{\theta}_X = \boldsymbol{\theta}_X^*)$, is named as the auxiliary variable. With U_X, Eq. (3) helps to build a deterministic input–output function $X = F(U_X, \boldsymbol{\theta}_X)$ for computing Sobol indices, since a sample of $\boldsymbol{\theta}_X$ and a sample of U_X lead to a deterministic value of X. Then the resultant Sobol index of $\boldsymbol{\theta}_X$ assesses the contribution of epistemic distribution parameter uncertainty, and the Sobol index of U_X assesses the contribution of the natural variability of X.

Although auxiliary variable is a standard procedure in sampling random variables, generally it is used implicitly and only the resultant samples of the random variables are recorded and utilized. However, as explained above, Ref. [27] found that if we use this auxiliary variable explicitly, it brings the benefit of separating the aleatory and epistemic uncertainty in a single random variable X. Ref. [27] only considered the aleatory and epistemic in the random variable model input, whereas this paper extends this idea to two situations: (1) time series input, and (2) model uncertainty. Thus we expand the use of the auxiliary variable to represent either aleatory or epistemic uncertainty depending on the situation and establish the deterministic function required to compute the Sobol indices. The proposed new methodology is developed in Section 3 below.

2.4. Autoregressive moving average (ARMA) model

This paper focuses on the model-based GSA with time series input. The ARMA model is selected to model the time series input due to its ability to capture both natural variability and epistemic uncertainty in the time series input. An ARMA(p,q) model assumes that the input at time step t is a linear combination of (1) earlier input values from step $t-p$ to step $t-1$; (2) earlier values of noise from step $t-q$ to step $t-1$; and (3) the current value of noise at step t, i.e.,

$$X_t = \bar{X} + \sum_{i_a=1}^{p} \phi_{i_a} X_{t-i_a} + \epsilon_t + \sum_{i_m=1}^{q} \theta_{i_m} \epsilon_{t-i_m} \tag{4}$$

where X_t and X_{t-i_a} are the inputs at time step t and time step $t-i_a$; $\boldsymbol{\phi} = \{\phi_1, \ldots, \phi_p\}$ are the coefficients of the AR model; $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_q\}$ are the coefficients of the MA model; $\overline{X}$ is a constant; and ϵ_t and ϵ_{t-i_m} are the random noise terms at time step t and time step $t-i_m$. All the random noise terms are generally assumed to be independent and identically distributed Gaussian variables $N(0, \sigma_\epsilon^2)$, i.e., Gaussian white noise [21]. And these noise terms represent the natural variability of the time series input.

To build an ARMA model, the values of its orders p and q are first identified by matching the theoretical autocorrelation function to the sample autocorrelation function computed from the observed time series data. The Ljung–Box Q statistic [22] can be used to measure the adequacy of the matching.

The values of the ARMA parameters $\{\bar{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon\}$ have epistemic uncertain due to limited history data. The ARMA model can capture this epistemic uncertainty by assigning probability distributions to the ARMA parameters $\{\overline{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon\}$. Bayesian inference may be used to calibrate the distributions of the ARMA parameters using the observed data [21]. In contrast, the counting matrix in the cycle counting methods is deterministic so that the epistemic uncertainty due to limited time series data is difficult to quantify.

3. Proposed methodology: Sobol indices under both aleatory and epistemic uncertainty

Theoretically GSA based on Sobol indices can be used to assess the contribution of any uncertainty source, no matter whether it is aleatory or epistemic. However, the existence of both aleatory and epistemic uncertainties in Table 1 brings two challenges to compute the Sobol indices using an input–output prediction model. First, the model prediction $Y = S(\boldsymbol{\theta}_m; \boldsymbol{X}) + \epsilon_h(\boldsymbol{X}) + \delta(\boldsymbol{X})$ is not deterministic, i.e., Y does not have a single deterministic value even if $\boldsymbol{\theta}_m$ and $\boldsymbol{X}$ are fixed. The reason is that $S(\boldsymbol{\theta}_m; \boldsymbol{X})$, $\epsilon_h(\boldsymbol{X})$ and $\delta(\boldsymbol{X})$ can each be uncertain even for fixed values of $\boldsymbol{X}$ and $\boldsymbol{\theta}_m$. In this paper, since the GP surrogate model is used, the surrogate model prediction $S(\boldsymbol{\theta}_m; \boldsymbol{X})$ is a Gaussian random variable for fixed values of $\boldsymbol{X}$ and $\boldsymbol{\theta}_m$; the discretization error $\epsilon_h(\boldsymbol{X})$ and model form error $\delta(\boldsymbol{X})$ are also estimated by GP models, thus they are both Gaussian random variables for a fixed value of $\boldsymbol{X}$. Therefore Y is the sum of three Gaussian random variables.

Second, each uncertainty source in Table 1 should be represented by a single random variable of known or fixed probabilistic distribution if we want to compute the Sobol indices. However, this required single random variable is not available for some uncertainty sources. The main reason is that one uncertainty source may depend on another one. For example, the uncertainty in the discretization error $\epsilon_h(\boldsymbol{X})$ depends on the value of $\boldsymbol{X}$. In this case, the distribution of $\epsilon_h(\boldsymbol{X})$ is not fixed but changes with the value of $\boldsymbol{X}$. The first contribution of this paper is to use the auxiliary variable to decouple the dependent uncertainty sources, so that the uncertainty term that depends on other uncertainty sources can be separately represented by a single auxiliary variable of fixed uniform distribution $U(0,1)$, and the deterministic function required for the Sobol indices computation can be established.

Identifying the single variable to represent the natural variability in the ARMA model is even more difficult. At given values of ARMA parameters, if we run the ARMA model N times, N different time series histories can be obtained. The variation among these histories represents the natural variability in the ARMA model (last

row in Table 1), which is caused by the noise terms $\{\epsilon_1, \epsilon_2, \ldots, \epsilon_N\}$ in the ARMA model at each time step. Although we can consider all the noise terms in the GSA, this will make the GSA extremely high-dimensional. Thus the second contribution of this paper is a new method defining a single auxiliary variable that captures all the noise terms, i.e., the natural variability in the ARMA model; this method is described in Section 3.2.2, and referred to as uncontrolled-seed method.

Although the proposed uncontrolled-seed method reduces the dimension of the GSA, its computational efficiency is still not satisfying. Therefore the third contribution of this paper is a new controlled-seed method proposed in Section 3.3, which uses the seed as a single random variable capturing the natural variability in the ARMA model. This method obtains the same result as the uncontrolled-seed method, and reduces computational cost significantly.

3.1. GSA with random variable input

The auxiliary variable method can be extended to any variable whose distribution is conditioned on other variables. Assume that the distribution of a random variable A depends on the value of another random value B by a conditional distribution $f(A|B)$. Then the uncertainty in $f(A|B)$ can be captured by a single auxiliary variable U_A, which is the CDF value of $f(A|B)$. In other words, the auxiliary variable can be used to represent any uncertainty term whose distribution depends on other uncertainty sources. The represented uncertainty term can be either aleatory or epistemic.

Assume that the model inputs $\boldsymbol{X} = \{X_1, \ldots, X_k\}$ are random variables. For the random variables $S(\boldsymbol{\theta}_m; \boldsymbol{X}), \epsilon_h(\boldsymbol{X})$ and $\delta(\boldsymbol{X})$ in Table 1 whose distribution is conditioned on the value of $\boldsymbol{X}$ and $\boldsymbol{\theta}_m$, auxiliary variables U_S, U_{ε_h} and U_δ can be introduced to represent the uncertainties due to surrogate model, discretization error, and model discrepancy respectively at fixed values of $\boldsymbol{X}$ and $\boldsymbol{\theta}_m$. In addition, auxiliary variables $\boldsymbol{U_X} = \{U_{X_1}, U_{X_2}, \ldots, U_{X_k}\}$ are also introduced for each model input $X_j (j = 1 \text{ to } k)$ that has both aleatory and epistemic uncertainty. Then a deterministic function suitable for Sobol indices computation can be built as:

$$Y = F(\boldsymbol{\theta}_m, \boldsymbol{\theta}_X, \boldsymbol{U_X}, U_S, U_{\varepsilon_h}, U_\delta) \tag{5}$$

Note that no auxiliary variable is needed for $\boldsymbol{\theta}_X$ or $\boldsymbol{\theta}_m$ since their distributions are not conditioned on any other variables. Another observation is that either aleatory or epistemic uncertainty can be represented by the auxiliary variables depending on the situation. For example, $\boldsymbol{U_X}$ represents the aleatory uncertainty in model inputs; whereas U_S, U_{ε_h}, and U_δ represent the epistemic uncertainties caused by surrogate model uncertainty, discretization error, and model form error respectively.

The flowchart in Fig. 4 illustrates the application of Eq. (5). A sample of the distribution parameters $\boldsymbol{\theta}_X$ gives the marginal distribution for each model input X, and auxiliary variables $\boldsymbol{U_X} = \{U_{X_1}, U_{X_2}, \ldots, U_{X_k}\}$ helps to generate a deterministic sample of $\boldsymbol{X}$ by CDF inversion on the joint distribution of model inputs $\boldsymbol{X}$.

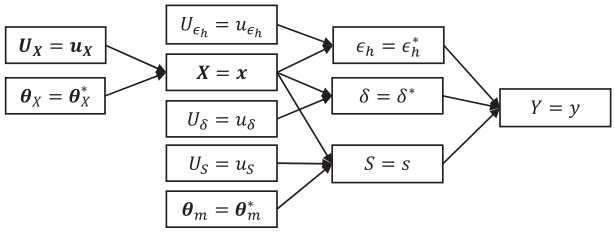

Fig. 4. Deterministic function for random variable input.

Note that the model inputs $\boldsymbol{X}$ discussed in this section is a set of scalar random variables. The case that $\boldsymbol{X}$ represents a time series input will be discussed in Section 3.2.

The sample of $\boldsymbol{X}$ decides the distribution of input-dependent discretization error ϵ_h and model form error δ, and the corresponding auxiliary variables U_{ϵ_h} and U_δ generate deterministic values of ϵ_h and δ, respectively by inverting the corresponding CDFs. Similarly, the value of S is determined by the value of $\boldsymbol{X}, \boldsymbol{\theta}_m$, and auxiliary variable U_S. Finally a deterministic prediction is computed as $y = s + \epsilon_h + \delta$. The deterministic function in Eq. (5) is now ready for Sobol indices computation. The resultant sensitivity indices of $\boldsymbol{\theta}_X$ assess the contributions of input distribution parameter uncertainty towards the uncertainty in model prediction Y; the indices of $\boldsymbol{\theta}_m$ assess the contributions of model parameter uncertainty; and the indices of auxiliary variables assess the contributions of the corresponding uncertainty sources, as shown in Table 1.

Note that Eq. (5) proposes a framework to assess the contribution of each uncertainty source with random variable inputs. If any uncertainty source is ignored in practice, this framework is still applicable by removing the corresponding variable in Eq. (5). For instance, if Richardson extrapolation is used to compute a deterministic discretization error and ignore the uncertainty in it, the auxiliary variable U_{ϵ_h} is not needed in Eq. (5). Similarly, if an input random variable X_j has only aleatory uncertainty and no epistemic uncertainty (i.e., its probability distribution is precisely known), then the corresponding auxiliary variable U_{X_j} is not needed; in this case, the probability density $f_{X_j}(x_j)$ represents the uncertainty (variability) in X_j.

3.2. GSA with time series input

3.2.1. Challenges with time series input

As discussed earlier, the epistemic uncertainty in the ARMA model of the times series input can be represented by assigning probability distributions to its parameters $\{\overline{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon\}$ and updating these distributions using Bayesian inference. Like other random process representations, the ARMA model takes the input at each time step as a random variable X_t and the observed value at this time step is a realization of this random variable. Theoretically, this time series can be considered as a N_t-dimensional vector of random variables $\boldsymbol{X} = \{X_1, \ldots, X_{N_t}\}$ where N_t is the number of time steps, so the flowchart in Fig. 4 is still applicable. However, since N_t is usually very large, several studies have tried to reduce this N_t-dimensional time series input to a low-dimensional representation.

Ben-Haim [29] employed a deterministic convex model of Fourier series rather than probabilistic models to represent the uncertainty in a load history. However, this deterministic model ignores the aleatory uncertainty in the time series input, even if the epistemic uncertainty can be introduced into this model by allowing the Fourier coefficients to vary. Echard et al. [30] used nine displacement histories to represent the uncertainty of in-service loads. This method needs adequate observations of time series input, which may be impossible.

Another option to reduce the dimension of a random process is the Karhunen–Loeve expansion [31,32], which represents a random process by the eigenvalues and eigenfunctions of the covariance function. The first l largest eigenvalues and the corresponding eigenfunctions are retained if the explained variance of the random process reaches a threshold such as 95% or 99%. The explained variance is given by $\sum_{i_e=1}^{l} \lambda_{i_e} / \sum_{i_e=1}^{\infty} \lambda_{i_e}$, where λ_{i_e} is the i_eth largest eigenvalue [32]. However, the value of l highly depends on the autocorrelation function of the random process: more eigenvalues and eigenfunctions are needed to explain the same

variance if the autocorrelation function decays faster. Consider a random process represented by an ARMA(1, 1) model $X_t = -2 + 0.2X_{t-1} + \epsilon_t + 0.2\epsilon_{t-1}$ where the noise terms have a Gaussian distribution $N(0, 0.1^2)$. Fig. 5 shows the autocorrelation function and the first 50 eigenvalues. The autocorrelation function decays to almost zero after 3 lags. No dominant eigenvalue is observed, therefore most eigenvalues should be retained to explain the variance. Thus the dimension of the random process cannot be significantly reduced in some cases.

3.2.2. Auxiliary variable and GSA with time series input

The objective of this paper is not only to make the Sobol indices computation affordable, but also to distinguish the contributions of aleatory and epistemic uncertainties towards the uncertainty in the prediction. Here the auxiliary variable method is extended to assess the individual contribution of each uncertainty source. The deterministic function required for the Sobol indices computation is:

$$Y = F(\bar{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon, \boldsymbol{\theta}_m, U_S, U_{\epsilon_h}, U_\delta, U_\epsilon) \tag{6}$$

An evaluation of Eq. (6) is shown Fig. 6, which can be realized in 7 steps:

1. Generate a sample of the ARMA model parameters $\bar{X}$, $\boldsymbol{\phi}$, $\boldsymbol{\theta}$, σ_ϵ from their joint distribution. This joint distribution represents the epistemic uncertainty regarding the ARMA model parameters, and can be obtained by Bayesian inference using observed time series data.
2. Generate a sample $\boldsymbol{\theta}_m^*$ of the physics model parameters $\boldsymbol{\theta}_m$.
3. Generate N time histories $\{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N\}$ based on the samples of $\bar{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon$ from Step 1. Here the model input $\boldsymbol{X} = \{X_1, \ldots, X_{N_t}\}$ is a time series input of N_t time steps. A generated history $\boldsymbol{x}^i (i = 1, \ldots, N)$ is a realization of $\boldsymbol{X}$, thus $\boldsymbol{x}^i = \left\{x_1^i, \ldots, x_{N_t}^i\right\}$ is a vector of N_t elements. The difference between these time histories represents the natural variability in the ARMA model caused by the noise terms. By propagating each time history with the sample of $\boldsymbol{\theta}_m^*$ through the stochastic surrogate model $S(\boldsymbol{\theta}_m; \boldsymbol{X})$, a family of N distributions can be constructed. Each distribution $S_i(\boldsymbol{\theta}_m^*, \boldsymbol{x}^i)(i = 1, \ldots, N)$ represents the effect of epistemic surrogate model uncertainty at a given time history, thus an auxiliary variable U_s is introduced to represent it.
4. Generate a sample of U_s to conduct CDF inversion of each distribution $S_i(\boldsymbol{\theta}_m^*, \boldsymbol{x}^i)(i = 1, \ldots, N)$ in Step 3. The resultant N samples $\{s_1, \ldots, s_N\}$ from the N distributions constitute a new random variable S whose uncertainty is caused by the ARMA model natural variability.
5. If the discretization error $\epsilon_h(\boldsymbol{X})$ is stochastic (e.g., due to the use of a GP model) at a given time series input, each time history from Step 3 gives a distribution of discretization error, thus a family of N distributions $\epsilon_{h_i}(\boldsymbol{x}^i)(i = 1 \text{ to } N)$ can be constructed. An auxiliary variable U_{ϵ_h} representing the discretization error uncertainty is introduced to obtain a sample from each distribution, and the resultant N samples construct a random variable ϵ_h whose uncertainty is caused by the ARMA model natural variability.

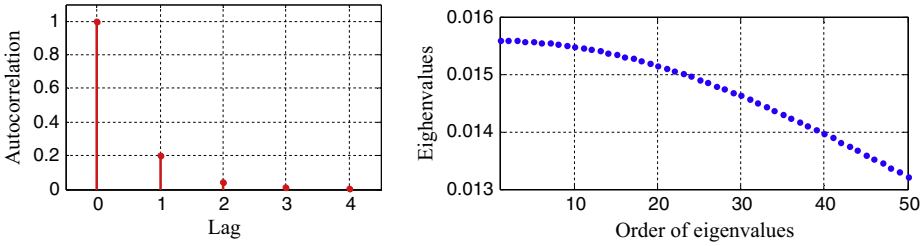

Fig. 5. Autocorrelation and eigenvalues for ARMA model.

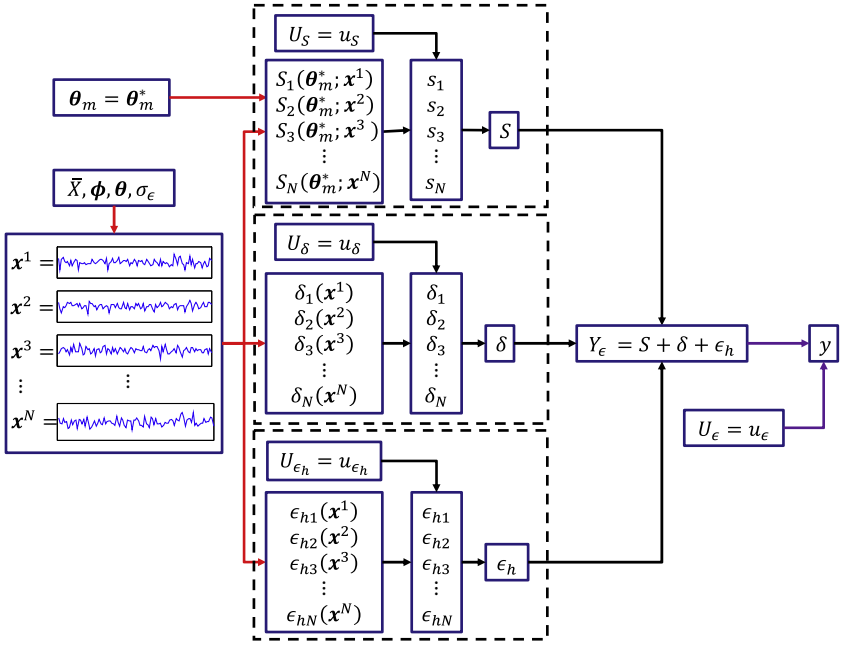

Fig. 6. An evaluation of Eq. (6).

6. Use the same procedure as Step 5 for $\delta(\boldsymbol{X})$: each time history from Step 3 gives a distribution of model discrepancy, thus a family of N distributions $\delta_i(\boldsymbol{x}^i)(i = 1 \text{ to } N)$ can be constructed. An auxiliary variable U_δ is introduced to obtain a sample from each distribution, and the resultant N samples construct a random variable δ whose uncertainty is caused by the ARMA model natural variability.
7. Define a new variable Y_ϵ as the sum of S, δ, and ϵ_h from steps 3 to 6. The uncertainty in S, δ and ϵ_h is caused by the natural variability in the ARMA model, thus the uncertainty in Y_ϵ is also caused by natural variability in the ARMA model. Another auxiliary variable U_ϵ is introduced to represent the uncertainty in Y_ϵ. (Note that S, δ, and ϵ_h can be correlated, and the calculation in Fig. 6 correctly accounts for this correlation by generating correlated samples of S, δ, and ϵ_h). With N samples of S from step 4, N samples of δ from step 5, and N samples of ϵ_h from step 6, we can obtain N samples of Y_ϵ to represent its distribution. A sample of U_ϵ is generated to conduct CDF inversion on Y_ϵ to obtain a deterministic value y so that a deterministic function can be established.

Note that Eq. (6) is as flexible as Eq. (5). The corresponding variable in Eq. (6) can be removed if any uncertainty source is ignored.

3.3. Controlled-seed method for GSA with time series input

3.3.1. Computational cost for the GSA with time series input

An important challenge in the application of Eq. (6) is the computational cost. Here we define "one evaluation of the deterministic function such as Eq. (6)" as a function evaluation. Computation of the Sobol indices based on Eqs. (1) and (2) is computationally intensive since it requires repeated function evaluations at different values of the inputs. Take the first-order index of X_j in Eq. (1) as an example. The numerator $V(E(Y|X_j))$ indicates a double-loop Monte Carlo simulation. The inner loop $E(Y|X_j)$ fixes X_j and computes the mean value of Y over n_1 different samples of X_{-j}. Thus the inner loop conducts n_1 function evaluations. The outer loop $V(E(Y|X_j))$ repeats the inner loop at n_2 different samples of X_j and computes the variance of mean values, thus the number of function evaluations for $V(E(Y|X_j))$ is $n_1 n_2$. As the deterministic function for GSA has k inputs $X_j(j = 1 \text{ to } k)$ and another n_3 function evaluations are required to compute the denominator $V(Y)$ in Eq. (1), the number of function evaluations to compute all the first-order indices is $N_f = kn_1n_2 + n_3$. If we assume $n_1 = n_2 = n_3 = n$:

$$N_f = kn^2 + n \tag{7}$$

The number of function evaluations for the total effects indices is the same as the first order indices. If $X_j(j = 1 \text{ to } k)$ are uncorrelated with each other, a single loop method [3] has been developed to reduce the cost in Eq. (7) to $kn + n$. But when the model inputs are correlated, there is no alternative to the double loop method [3]. This paper only applies the double loop method, considering the general case of correlated inputs.

Regarding Eq. (6) for GSA with time series input, one function evaluation shown in Fig. 6 requires computations over N time histories. If the time cost for one time history is t_0, the time cost for one function evaluation of Eq. (6) is Nt_0. Thus the overall time cost for the first-order indices is:

$$T_1 = Nt_0 \times N_f = Nt_0\left(kn^2 + n\right) \tag{8}$$

The time cost given by Eq. (8) is sometimes unaffordable. Consider a simple example where (1) the time series input is generated by a ARMA(1, 1) model of four model parameters, i.e., $\bar{X}, \phi_1, \theta_1$ and σ_ϵ; (2) the discretization error and surrogate model uncertainty are ignored; and (3) the values of the model parameters $\boldsymbol{\theta}_m$ are precisely known. Then the deterministic function of Eq. (6) reduces to $Y = F(\bar{X}, \phi_1, \boldsymbol{\theta}_1, \sigma_\epsilon, U_\delta, U_\epsilon)$, which requires a six-dimensional GSA ($k = 6$). Assume $t_0 = 0.01$ s, which is quite fast and implies the use of a surrogate or a simplified reduced-order model for a realistic structure. Suppose $N = 100$ and $n = 500$, the overall time cost by Eq. (8) is about 417 h, which is rarely affordable. Of course, parallel computing can be used to reduce this time cost, but that requires more computational resources.

The reason for the unaffordable time cost by Eq. (8) is as follows: in Eq. (6) the natural variability of time series input is represented by N sampled time histories (Fig. 6), so the auxiliary variable U_ϵ can be introduced only after computing all the sampled time histories to predict the system response. In other words, one function evaluation of Eq. (6) requires computing N time histories. In contrast, in Eq. (5) for random variable inputs, the natural variability in random variable input X is represented by a single PDF (a PDF in Fig. 3), and the auxiliary variable U_X generates a deterministic value of X from this distribution. Thus in a function evaluation of Eq. (5), only a single value of X is propagated into the model of $S(\boldsymbol{\theta}_m; \boldsymbol{X}), \epsilon_h(X)$ and $\delta(\boldsymbol{X})$ to predict the system response. Therefore, the function evaluation of Eq. (6) can be accelerated significantly if the natural variability in time series input can be captured by a single PDF before propagating the time histories through the prediction model. The next subsection proposes a controlled-seed method to achieve this outcome.

3.3.2. Proposed method

As explained earlier, the natural variability of time series input is represented by generating multiple time histories, which is basically a process of generating random numbers. Random numbers in computers are always generated by deterministic algorithms such as Mersenne Twister generator [33], Combined Multiplicative Recursive generator [34] or Wichmann–Hill generator [35]. These pseudo-random number generators use a positive integer known as seed to generate a random number of various distribution types, and a new seed is deterministically computed before generating the next random number. A fixed initial seed value will give a fixed set of random numbers. Nevertheless, the deterministic generators are sufficiently complicated so that the generated pseudo-random samples can pass various statistical tests of randomness.

Therefore, if a code is used to generate time series input using a mathematical model such as the ARMA model, the sample at each time step is determined once the initial seed value for sampling the first time step is given. For example, Fig. 7 shows that the same initial seed K leads to the same load history at different runs of the ARMA model in MATLAB.

This initial seed K is considered as a random variable controlling the generation of the time series input. For the random variable input, the auxiliary variable U_X captures the natural variability in the random variable input X due to the one-to-one mapping between each value of U_X and the value of X; similarly, the initial seed K captures the natural variability in the time series input due to one-to-one mapping between the value of K and the realization of the time series input. It is equally possible for any positive integer to be used as a seed, so theoretically K has a discrete uniform distribution $U_d(1, n_c)$ where the upper bound n_c is a very large positive integer decided by the specific programing language and computer. But in practice we can define the bounds of this discrete uniform distribution, depending on how many different possible histories are adequate to represent the natural variability in time series input. The numerical example in Section 4.2.1 assigns a discrete uniform distribution $U_d(1, 100)$ to the initial seed K by implying that 100 possible histories are adequate to represent the natural variability in the ARMA model.

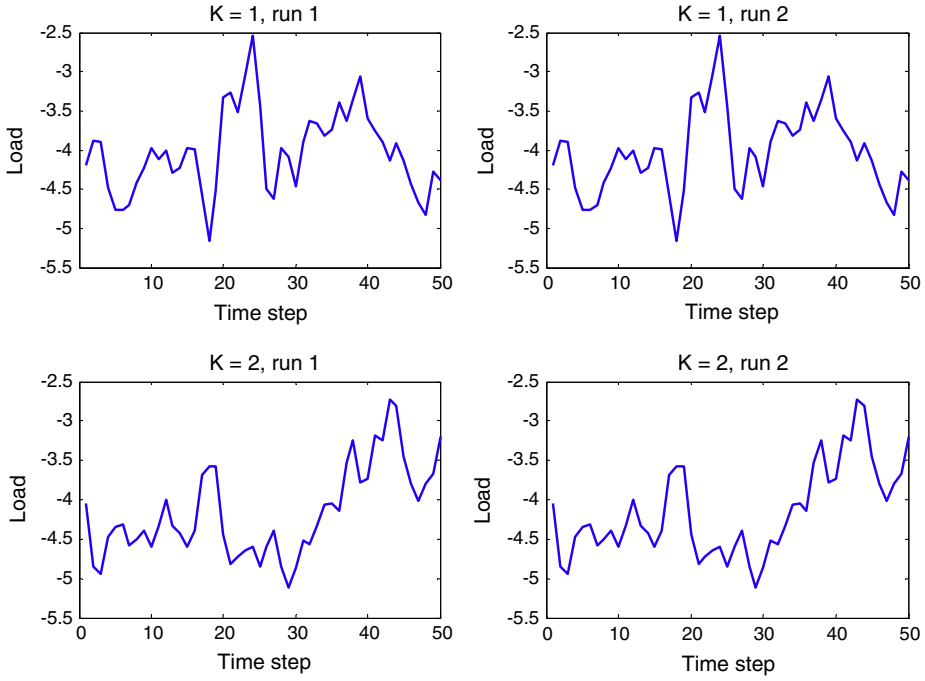

Fig. 7. Seed and ARMA model simulation in MATLAB.

An additional step is needed to apply the initial seed K to global sensitivity analysis. Although the initial seed K captures the natural variability in the time series input, its distribution is discrete but Sobol indices requires continuous random variables. Therefore another auxiliary variable U_K, which is the CDF value of K, is introduced to represent K. The mapping between the value of U_K and the value of K is:

$$K = a + \lfloor U_K(b - a + 1) \rfloor \tag{9}$$

where a and b are the positive integers of lower and upper bounds respectively, and $\lfloor \cdot \rfloor$ is the floor function. The first constraint for a and b is that $a < b \leqslant n_c$. In addition, the difference between a and b should be large enough to guarantee the diversity of resultant seed values, so that adequate different time histories can be generated to represent the natural variability in the time series input.

In Fig. 6, the auxiliary variable U_ϵ is to pick one sample of Y from N samples. In other words, U_ϵ actually picks one time series time history. Now the auxiliary variable U_K reaches the same objective, thus it equivalently captures the natural variability in the time series input via K. Then a new deterministic function for GSA is proposed as:

$$Y = F(\bar{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon, \boldsymbol{\theta}_m, U_S, U_{\epsilon_h}, U_\delta, U_K) \tag{10}$$

where U_K plays the same role as U_ϵ in Eq. (6).

Similar to Eq. (6), an evaluation of Eq. (10) is shown in Fig. 8, which can be realized in five steps:

1. Generate a sample of $\bar{X}$, $\boldsymbol{\phi}$, $\boldsymbol{\theta}$, σ_ϵ from their joint distribution;
2. Sample U_K and compute the corresponding value of K using Eq. (9). Then generate a time history $\boldsymbol{x}$ by taking the value of K as the initial seed;
3. Generate a sample $\boldsymbol{\theta}_m^*$ of model parameters $\boldsymbol{\theta}_m$;
4. Compute $S(\boldsymbol{\theta}_m^*; \boldsymbol{x}), \epsilon_h(\boldsymbol{x})$ and $\delta(\boldsymbol{x})$, where each one is a distribution;
5. Sample the auxiliary variables U_S, U_ϵ and U_δ to obtain deterministic values of s, ϵ_h and δ by CDF inversion on the distributions in Step 4, respectively. Then the deterministic value of response prediction is $y = s + \epsilon_h + \delta$.

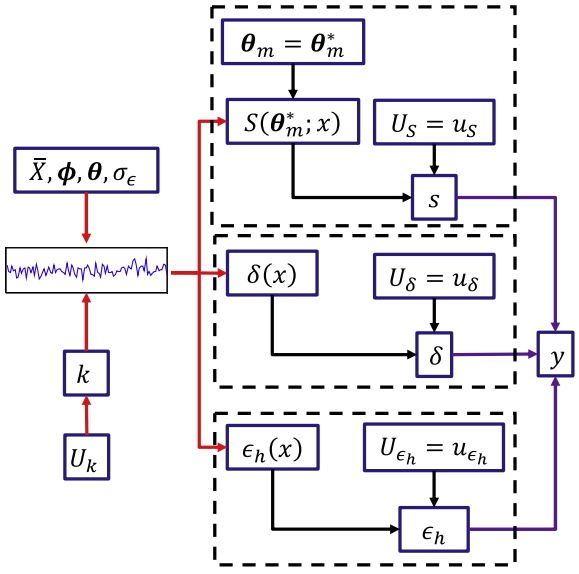

Fig. 8. An evaluation of Eq. (10).

Note that Eq. (10) is as flexible as Eqs. (5) and (6). The corresponding variable can be removed if any uncertainty source is ignored.

As the initial seed K is introduced, the proposed method by Eq. (10) is named as "controlled-seed method"; in contrast, the method by Eq. (6) is named as "uncontrolled-seed method". By developing the controlled-seed method, the natural variability in time series input is captured by U_K. As shown in Fig. 8, only one time history requires computation in each function evaluation of Eq. (10). Thus the overall time cost for the first-order indices is:

$$T_2 = t_0(kn^2 + n) \tag{11}$$

Compared with Eq. (8), the computational effort is significantly reduced by the factor N. For the earlier example in Section 3.3.1

where $t_0 = 0.01$ s, $n = 500$, and $N = 100$, the time cost reduces to 4.17 h, instead of 417 h.

3.4. Summary

This section first considers random variable inputs and model uncertainty terms. The auxiliary variable method facilitates separating each uncertainty source in the system response prediction, and helps to establish a deterministic function in Eq. (5) in order to compute the Sobol indices for each uncertainty source (either aleatory or epistemic).

Next this method is extended to handle time series input. An uncontrolled-seed method is first proposed where an auxiliary variable U_ϵ is introduced to capture the natural variability after propagating all the time histories through the prediction model. The uncontrolled-seed method proves to be time-consuming since one function evaluation requires computations for N time histories. Thus a controlled-seed method is proposed to promote computational efficiency. The controlled-seed method uses the initial seed to represent the natural variability in the time series input. By comparing Figs. 6 and 8, the advantage of the seed-controlled method is that one function evaluation requires computation for only one time history, instead N histories. From Eqs. (8) and (11) we observe that the controlled-seed method accelerates the GSA by reducing the cost of function evaluations but not the algorithm (double-loop method here) to compute the Sobol indices. Thus the controlled-seed method is equivalently useful if other algorithms such the single-loop method are utilized to compute the Sobol indices.

Furthermore, this paper focuses on a scalar prediction Y. However, with time series input, a time-indexed set of outputs may be of interest. A user may require the sensitivity indices at all the time steps or at least at a few key time steps, since many models show sensitivity to some parameters at early time but sensitivity to other parameters at later time. The proposed framework in this section can be easily extended to this case of multiple output time steps by handling each time step separately.

4. Numerical example

Consider a single cantilever beam shown in Fig. 9. An edge crack is assumed to have initiated at the top surface, and this crack grows under the time series loading $\boldsymbol{X}$ of N_t cycles imposed at the other end of the beam. The initial crack size a_0 is assumed to have a normal distribution $N(0.03, 0.0015^2)$, representing the uncertainty in measuring a_0. The objective of this example is to assess the contribution of each uncertainty source to the uncertainty in the final crack length prediction. The uncertainty sources include structure properties (structure geometry, initial crack size, material properties, and crack growth parameters), loading history, and various model errors (surrogate model error, discretization error, and model form error). In this example, for the sake of illustration, we only consider the uncertainty in initial crack size, loading history, and model errors. Properties of the structure are assumed to be fixed and known. However, the proposed methodology can easily include these additional uncertainty sources.

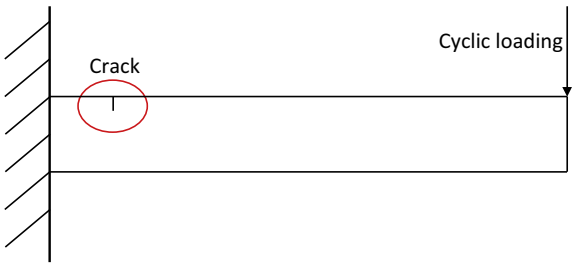

Fig. 9. Cantilever beam.

Section 4.1 illustrates the prediction model to compute the final crack length, i.e., how to compute the crack growth at a given time series history generated by ARMA. Section 4.2 develops the deterministic functions required for global sensitivity analysis and provides two scenarios: Section 4.2.1 assumes known ARMA model parameter distributions, and compares the efficiency of the uncontrolled-seed method proposed in Section 3.2.2 and the controlled-seed method proposed in Section 3.3.2; Section 4.2.2 calibrates ARMA model parameters by Bayesian inference, and the effect of correlation between ARMA parameters is investigated.

4.1. Computational models

As shown in Fig. 10, two finite element (FE) models are established by the commercial software ANSYS to compute the stress intensity factor ΔK_s under load X and crack length A. The first FE model has coarse mesh around the crack tip, while the second FE model has fine mesh around the crack tip.

At given stress intensity factor ΔK_s, an empirical curve of crack growth rate vs. stress intensity factor obtained in material experiment can be used to compute crack growth ΔA in each cycle, as shown in Fig. 11.

Alternatively, the Paris' law can be also used as the crack growth model to compute ΔA:

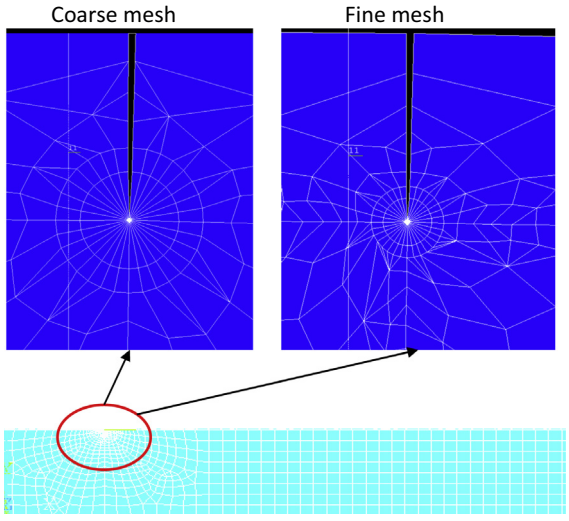

Fig. 10. FEA model.

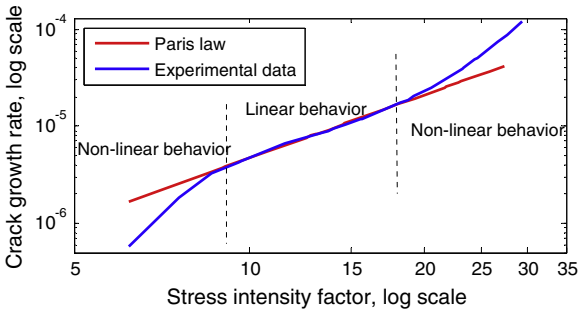

Fig. 11. Paris law vs. empirical crack growth curve.

$$dA/dN = C\Delta K_s^m \tag{12}$$

In Eq. (12), C and m are Paris' law parameters; dA/dN is the crack growth rate, and its magnitude is equal to the predicted crack growth ΔA in one cycle. Since ΔK_s depends on load X and the crack length A, ΔA is a also function of X and A, i.e., $\Delta A(X, A)$.

Paris' law is fits the linear behavior part of the empirical curve well, but diverges from the empirical curve in the non-linear behavior parts and brings errors. Using the linear behavior data of the empirical curve, the Paris's law parameters C and m are obtained by a linear regression model of $\log(dA/dN) = \log C + m\log\Delta K_s$. The values of C and m by the linear regression are $C = 3.2379 \times 10^{-8}$ and $m = 2.1577$. Since this linear regression gives a high R-squared value of 0.997, this paper fixes C and m as constants.

Depending on different mesh resolutions and crack growth models, three models with different levels of fidelity are established, as shown in Table 2. The crack growth predicted by each of the these three models in the tth cycle are denoted as ΔA_t^l, ΔA_t^m, and ΔA_t^h, respectively.

For the sake of illustration, the crack growth prediction by the high fidelity model is assumed to be the true value, and the low fidelity model is assumed to be the computational model. As illustrated in Section 1, the computational model needs two corrections to approximate the true value. At the tth cycle, the low fidelity model prediction ΔA_t^l is corrected as:

$$\begin{aligned}\Delta A_t^c(X_t, A_{i-1}) &= \Delta A_t^l + \left(\Delta A_t^m - \Delta A_t^l\right) + \left(\Delta A_t^h - \Delta A_t^m\right) \\ &= \Delta A_t^l(X_t, A_{t-1}) + \epsilon_h(X_t, A_{t-1}) + \delta(X_t, A_{t-1})\end{aligned} \tag{13}$$

where X_t is the load at the tth cycle, and A_{t-1} is the crack length after the $(t-1)$th cycle. The difference between ΔA_t^m and ΔA_t^l is caused by mesh resolutions (indicating discretization error) and denoted as $\epsilon_h(X_t, A_{t-1})$; and the difference between ΔA_t^h and ΔA_t^m is caused by different crack growth models (model form error or model discrepancy) and denoted as $\delta(X_t, A_{t-1})$.

25 values of $\epsilon_h(X_t, A_{t-1})$ and $\delta(X_t, A_{t-1})$ at different load X and crack length A are computed to train Gaussian process (GP) models for $\epsilon_h(X_t, A_{t-1})$ and $\delta(X_t, A_{t-1})$, which are used to compute the error terms at desired values of load and crack length. The GP model output for $\epsilon_h(X_t, A_{t-1})$ is denoted as $gp_{\epsilon_h}(X_t, A_{t-1})$, and the GP model output for $\delta(X_t, A_{t-1})$ is denoted as $gp_\delta(X_t, A_{t-1})$. In addition, for the sake of computational efficiency during uncertainty propagation (since many Monte Carlo samples will be used), a third GP model denoted as $gp_s(X_t, A_{t-1})$ is built to replace the low fidelity model in Table 2 and used in the prediction. Therefore the crack growth prediction at the tth cycle is:

$$\Delta A_t^c(X_t, A_{t-1}) = gp_s(X_t, A_{t-1}) + gp_{\epsilon_h}(X_t, A_{t-1}) + gp_\delta(X_t, A_{t-1}) \tag{14}$$

Note that all three terms at the right hand side of Eq. (14) are GP models, thus their outputs are Gaussian random variables at given values of X_t and A_{t-1}, so that the crack growth ΔA_t^c is also a Gaussian variable. In our computation, the standard deviation of this Gaussian variable is less than 1% of its mean value, so that the probability that Eq. (14) gives a negative crack growth is almost zero.

Eq. (14) is for one cycle. The final crack length is predicted by applying Eq. (14) at all cycles sequentially and using $A_t = A_{t-1} + \Delta A_t^c$. Since the crack growth $\Delta A_t^c(X_t, A_{t-1})$ is the sum of three Gaussian distributions, the crack growth in each cycle is stochastic so the starting crack length for each cycle is also stochastic. But this stochastic starting crack length will make the application of Eq. (14) tedious due to the nesting of Monte Carlo sampling loops from one cycle to another. Since this numerical example is mainly used to illustrate the proposed framework of contribution assessment, we simply use the mean value of A_{t-1} to compute the crack growth; the uncertainty in A_t is the accumulated uncertainty from the three GP models. Specifically, each of the three Gaussian distributions on the right hand side of Eq. (14) are separated into the sum of its mean value and a zero mean Gaussian distribution:

$$\begin{aligned}gp_s(X_t, A_{t-1}) &= \mu_s(X_t, A_{t-1}) + gp_s^0(X_t, A_{t-1}) \\ gp_{\epsilon_h}(X_t, A_{t-1}) &= \mu_{\epsilon_h}(X_t, A_{t-1}) + gp_{\epsilon_h}^0(X_t, A_{t-1}) \\ gp_\delta(X_t, A_{t-1}) &= \mu_\delta(X_t, A_{t-1}) + gp_\delta^0(X_t, A_{t-1})\end{aligned} \tag{15}$$

The crack length prediction A_t after the tth cycle is assumed to be the sum of a mean value μ_{A_t} and three zero mean Gaussian distributions:

$$A_t = \mu_{A_t} + \sum_1^t gp_s^0(X_t, \mu_{A_{t-1}}) + \sum_1^t gp_{\epsilon_h}^0(X_t, \mu_{A_{t-1}}) + \sum_1^t gp_\delta^0(X_t, \mu_{A_{t-1}}) \tag{16}$$

where:

$$\begin{aligned}&\mu_{A_t} = \mu_{A_{t-1}} + \mu_s\left(X_t, \mu_{A_{t-1}}\right) + \mu_{\epsilon_h}\left(X_t, \mu_{A_{t-1}}\right) + \mu_\delta\left(X_t, \mu_{A_{t-1}}\right) \text{ for } t \geqslant 2 \\ &\mu_{A_t} = a_0 + \mu_s(X_t, a_0) + \mu_{\epsilon_h}(X_t, a_0) + \mu_\delta(X_t, a_0) \text{ for } t = 1\end{aligned} \tag{17}$$

Eq. (16) is the prediction model used in this numerical example. In Eq. (16), $\sum_1^i gp_s^0(X_t, A_{t-1})$ is the variable of accumulated surrogate model uncertainty, denoted as S_t^a; $\sum_1^t gp_{\epsilon_h}^0(X_t, A_{t-1})$ is the variable of accumulated discretization error uncertainty, denoted as $\epsilon_{h_t}^a$; $\sum_1^i gp_\delta^0(X_t, A_{t-1})$ is the variable of accumulated mode discrepancy uncertainty, denoted as δ_t^a. Auxiliary variables will be introduced to assess the contribution of S_t^a, $\epsilon_{h_t}^a$ and δ_t^a to the uncertainty of A_t.

4.2. Contribution assessment of each uncertainty source

First, the uncertainty in the final crack length A_{N_t} is from the time series input represented by an ARMA model, including the natural variability in ARMA model and the epistemic uncertainty in the ARMA parameters. Second, for a given time series input, the uncertainty in A_{N_t} is from the three accumulative error terms in Eq. (16). Based on Eqs. (6) and (9), the deterministic functions required in global sensitivity analysis are:

$$A_{N_t} = F\left(a_0, \bar{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon, U_S, U_{\epsilon_h}, U_\delta, U_\epsilon\right) \text{ for uncontrolled-seed method} \tag{18}$$

$$A_{N_t} = F\left(a_0, \bar{X}, \boldsymbol{\phi}, \boldsymbol{\theta}, \sigma_\epsilon, U_S, U_{\epsilon_h}, U_\delta, U_K\right) \text{ for controlled-seed method} \tag{19}$$

The evaluations of Eqs. (18) and (19) follow the steps in Sections 3.2.2 and 3.3.2, respectively.

Table 2
Models of different fidelities.

Models	Predicted crack growth in each cycle	Mesh type	Crack growth model
Low fidelity model	ΔA_t^l	Coarse mesh	Paris's law
Mid fidelity model	ΔA_t^m	Fine mesh	Paris' law
High fidelity model	ΔA_t^h	Fine mesh	Empirical curve

4.2.1. Uncontrolled-seed method vs. controlled-seed method with known ARMA model

This section assumes that the time series input is a ARMA(1, 1) model. The distributions of ARMA parameters are assumed as $\bar{X} \sim N(-2, 0.2^2), \phi_1 \sim N(0.5, 0.1^2), \theta_1 \sim N(0.75, 0.1^2), \sigma_\epsilon \sim U(0.1, 0.5)$ and they are uncorrelated. To make the uncontrolled-seed method computationally affordable, we assume that the time series input only has 10 time steps. Assuming that 100 time series histories are adequate to represent the ARMA model with given parameters, Eq. (18) generates $N = 100$ time series in each function evaluation and Eq. (19) sets the distribution of seed as $K \sim U_d(1, 100)$. This assumption can be verified by checking the consistency of the autocorrelation function $R_s(\tau)$ based on 100 sample histories and the theoretical autocorrelation function $R(\tau)$ of ARMA(1, 1) model with the given ARMA parameters. The comparison of $R_s(\tau)$ and $R(\tau)$ is shown in Fig. 12 for a ARMA model with $\bar{X} = -2$, $\phi_1 = 0.5$, $\theta_1 = 0.75$ and $\sigma_\epsilon = 0.1$, where $R_s(\tau)$ are computed based on 100 sample histories with initial seed values ranging from 1 to 100. In Fig. 12, $R_s(\tau)$ and $R(\tau)$ is consistent, so our assumption of 100 sample histories is reasonable.

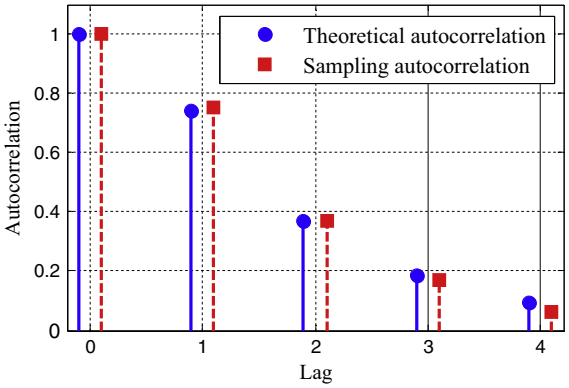

Fig. 12. Theoretical and sampling autocorrelation function for ARMA(1, 1) model.

Table 3
Global sensitivity analysis for example 1.

Methods	First-order effects		Total effects	
	Uncontrolled-seed	Controlled-seed	Uncontrolled-seed	Controlled-seed
n	120	500	120	500
Time (h)	17.4	3.0	17.7	3.1
Indices				
a_0	0.073	0.075	0.086	0.084
$\bar{X}$	0.107	0.100	0.297	0.290
ϕ_1	0.497	0.484	0.735	0.737
θ_1	0.000	0.000	0.001	0.001
σ_ϵ	0.004	0.006	0.032	0.039
U_S	0.000	0.000	0.000	0.000
U_{ϵ_h}	0.000	0.000	0.000	0.000
U_δ	0.000	0.000	0.000	0.000
U_ϵ/U_K	0.051	0.055	0.068	0.072

The result of GSA using both the uncontrolled-seed and controlled-seed method are reported in Table 3. Both the first order and total effects indices can be reported since all the variables in this example are uncorrelated.

Table 3 shows consistent results between the two methods: (1) the indices for the same uncertainty source using different methods are very close; (2) both U_ϵ in the uncontrolled-seed method and U_K in the seed method equivalently capture the natural variability in time series input; (3) ϕ_1 is the most dominant variable in the prediction uncertainty. The slight difference between the indices for the same uncertainty source is mainly caused by the limited number of samples in the uncontrolled-seed method ($n = 120$). But if we also apply 500 samples for the uncontrolled-seed method, its time cost will be over unaffordable 300 h. This also proves the efficiency of the controlled-seed method.

4.2.2. Correlation vs. non-correlation with calibrated ARMA model

Fig. 13 shows a synthetic time history generated as observed data. The loading at each cycle includes a maximum value and a minimum value. Fig. 13 only shows the minimum value at each cycle since the maximum values are assumed to be zero.

An ARMA(2, 2) model is selected to model this time series input. The parameters of the ARMA(2, 2) model are $\bar{X}$, $\boldsymbol{\phi} = \{\phi_1, \phi_2\}$, $\boldsymbol{\theta} = \{\theta_1, \theta_2\}$. and σ_ϵ. Prior distributions are assumed for the ARMA model parameters, and posterior distributions are obtained from Bayesian calibration using Markov Chain Monte Carlo (MCMC) sampling [36].

The marginal PDFs of the priors and the posterior distributions are shown in Fig. 14. The posteriors of some ARMA parameters are highly correlated, as shown in bold in the correlation matrix of Table 4. For example, the correlation between ϕ_1 and ϕ_2 is -0.8. This can also be explained physically: one criterion to guarantee the stationarity of the ARMA(2, 2) model is $\phi_1 + \phi_2 < 1$ [20], i.e., a larger ϕ_1 requires a smaller ϕ_2 thus indicates a negative correlation between them. The correlation of ARMA parameters has a significant influence on assessing the contribution of each uncertainty source, as shown later.

By assuming that 100 samples of the time series are adequate to represent the ARMA model with given parameters, Eq. (18) generates 100 time series in each evaluation and Eq. (19) sets the distribution of seed as $K \sim U_d(1, 100)$.

In addition to the longer time series input (200 verses 10), this example is different from the previous one regarding the correlation of ARMA parameters. To show the impact of this correlation, two results of global sensitivity analysis are shown in Table 5. One result intentionally uses the marginal distributions in Fig. 14 and ignores the correlation in ARMA parameters and another correctly considers the correlation. Only the first-order indices are reported since the total effects indices are not applicable for corre-

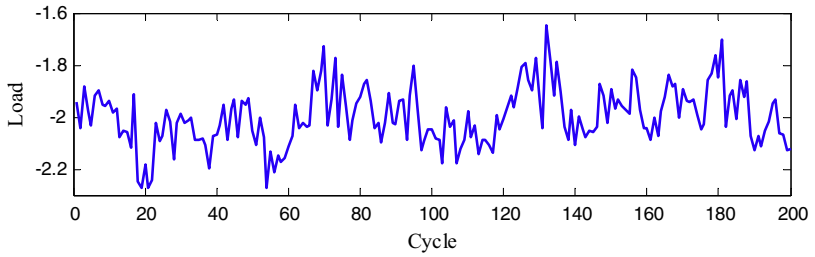

Fig. 13. Synthetic time series data.

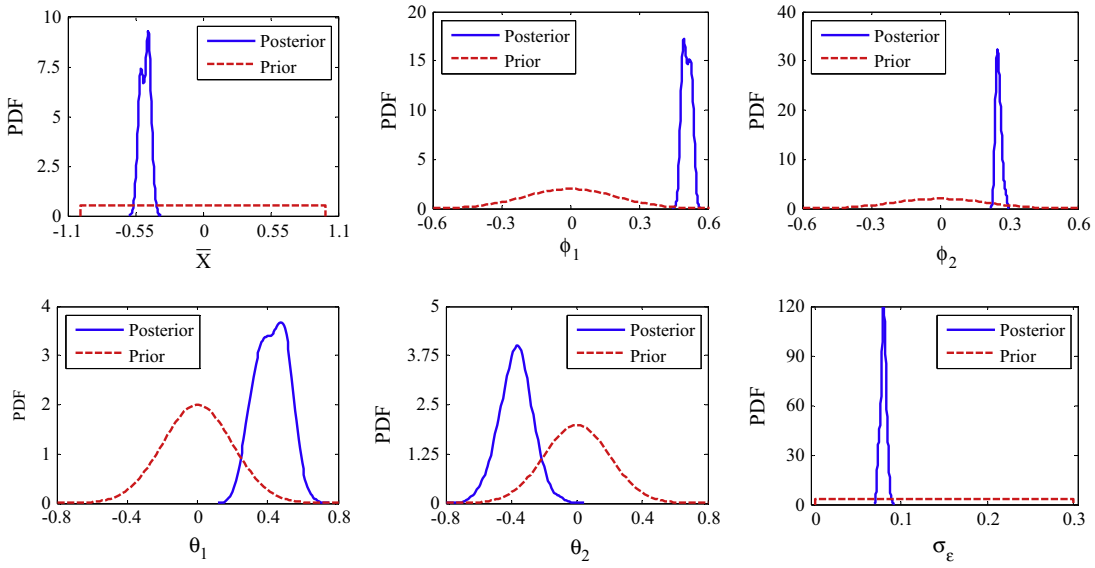

Fig. 14. Prior and posterior distributions of ARMA parameters.

Table 4
Correlation matrix of ARMA parameters.

	$\bar{X}$	ϕ_1	ϕ_2	θ_1	θ_2	σ_ϵ
$\bar{X}$	1.000	0.174	0.451	0.082	−0.134	0.004
ϕ_1	0.174	1.000	−0.800	0.059	−0.125	−0.082
ϕ_2	0.451	−0.800	1.000	−0.004	0.033	0.080
θ_1	0.082	0.059	−0.004	1.000	−0.105	−0.400
θ_2	−0.134	−0.125	0.033	−0.105	1.000	0.279
σ_ϵ	0.004	−0.082	0.080	−0.400	0.279	1.000

lated variables. This example only uses the controlled-seed method since the uncontrolled-seed method is not affordable for a time series with 200 cycles, given the computational resources available.

The indices in Table 5 indicate the impact of ARMA parameter correlation in assessing the contribution of each uncertainty source. The result ignoring correlation misleads us to take ϕ_1 and ϕ_2 as the dominant factors, while actually their contribution reduces when the correlation is considered. The reason for this overestimation can be revealed by the scatter plots in Fig. 15; the scatter width of ϕ_1 and ϕ_2 is much narrower due to the correlation of −0.8 between them, so the uncertainty caused by them in the prediction is reduced significantly.

In the result considering correlation, the important uncertainty sources are initial crack size a_0 and time series input natural variability U_K. The indices for ARMA model parameters are all small, indicating that collecting more time series data cannot help us reduce uncertainty in the final crack length prediction.

Table 5
Global sensitivity analysis: First-order indices for example 2.

	a_0	$\bar{X}$	ϕ_1	ϕ_2	θ_1	θ_2	σ_ϵ	U_S	U_{ϵ_h}	U_δ	U_K
Corr. ignored	0.001	0.055	0.249	0.314	0.000	0.000	0.000	0.000	0.000	0.000	0.001
Corr. considered	0.291	0.002	0.001	0.000	0.003	0.002	0.002	0.001	0.001	0.005	0.525

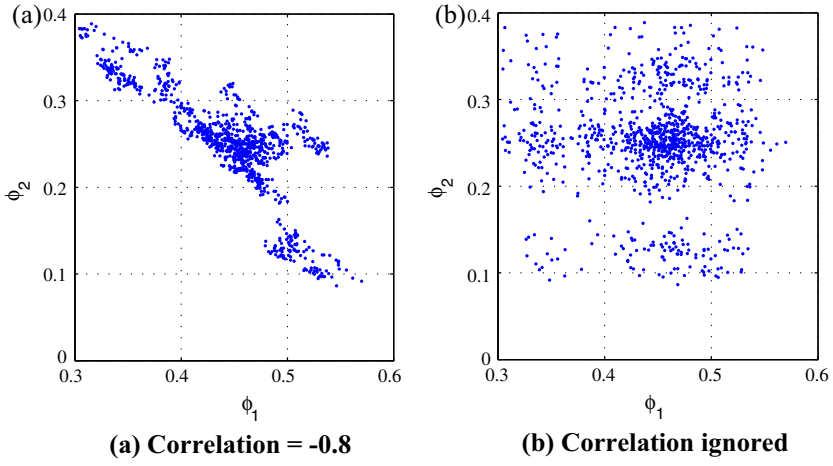

Fig. 15. Scatter plot of ϕ_1 and ϕ_2.

In this example, the sensitivity indices for surrogate model error (captured by U_S), discretization error (captured by U_{ϵ_h}) and model form error (captured by U_δ) are very small because our GP models are quite accurate and the variance of the GP model prediction is very small. Here the GP models reach high accuracy because (1) they have only two inputs (load and crack length); and (2) the crack growth is a smooth function with weak non-linearity. Thus 25 training points were enough to achieve very low prediction variance.

5. Summary

Various uncertainty sources arise at different steps in the computational prediction of the system response, including surrogate model uncertainty, model discrepancy, model input uncertainty, etc. Some uncertainty sources are aleatory and some are epistemic. In this paper, global sensitivity analysis (GSA) based on Sobol indices is used to quantify the contribution of each uncertainty source. One challenge is that under aleatory and epistemic uncertainty the prediction model is stochastic whereas the Sobol indices computation requires a deterministic model. Another challenge is that with time series input the GSA will be extremely high-dimensional since each time step introduces a random noise term in the ARMA model.

To solve the first challenge, this paper uses the auxiliary variable to represent each uncertainty source explicitly and establish the required deterministic function such that the Sobol indices can be computed. Based on the auxiliary variable, this paper proposes an uncontrolled-seed method to solve the second challenge, by defining a single variable to represent the natural variability in the time series input, thus reducing the problem dimension. Furthermore, a novel controlled-seed method is proposed based on the concept of pseudo-random number generation. This method requires computing only one history in each function evaluation thus the computation of the Sobol indices is significantly accelerated. These there contributions help to assess the contributions of each aleatory and epistemic uncertainty source to the uncertainty in the time series prediction, such as fatigue crack growth.

Acknowledgement

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.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.ijfatigue.2015.09.002.

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] Sobol IM. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math Comput Simul 2001;55(1–3):271–80. Feburary.
[3] Saltelli A. Making best use of model evaluations to compute sensitivity indices. Comput Phys Commun 2002;145(2):280–97. May.
[4] Zhang X, Pandey MD. An effective approximation for variance-based global sensitivity analysis. Reliab Eng Syst Saf 2014;121(January):164–74.
[5] Li C, Mahadevan S. Uncertainty quantification and output prediction in multi-level problems. In: 16th AIAA non-deterministic approaches conference; 2014.
[6] Aughenbaugh JM, Paredis CJJ. Probability bounds analysis as a general approach to sensitivity analysis in decision making under uncertainty; 2007.
[7] Ferson S, Troy Tucker W. Sensitivity analysis using probability bounding. Reliab Eng Syst Saf 2006;91(10–11):1435–42. October.
[8] Oberguggenberger M, King J, Schmelzer B. Classical and imprecise probability methods for sensitivity analysis in engineering: a case study. Int J Approx Reason 2009;50(4):680–93. April.
[9] Absi NG, Mahadevan S. Calibration of system parameters under model uncertainty. In: IMAC XXXII; 2014.
[10] Liang B, Mahadevan S. Error and uncertainty quantification and sensitivity analysis in mechanics computational models. Int J Uncertain Quantif 2011;1(2):147–61.
[11] Richards SA. Completed Richardson extrapolation in space and time. Commun Numer Meth Eng 1997;13(7):573–82.
[12] Rangavajhala S, Sura VS, Hombal VK, Mahadevan S. Discretization error estimation in multidisciplinary simulations. AIAA J 2011;49(12):2673–83. December.
[13] 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;June.
[14] Kennedy M, O'Hagan A. Bayesian calibration of computer models. J Roy Stat Soc 2001;63(3):425–64.
[15] Rasmussen CE, Williams CKI. Gaussian processes for machine learning. MIT Press; 2006.
[16] Haldar A, Mahadevan S. Probability, reliability, and statistical methods in engineering design. John Wiley & Sons Incorporated; 2000.
[17] Halpern EF, Weinstein MC, Hunink MGM, Gazelle GS. Representing both first- and second-order uncertainties by monte carlo simulation for groups of patients. Med Decis Mak 2000;20(3):314–22. July.
[18] Dowling NE. Fatigue failure predictions for complicated stress–strain histories; 1971.
[19] Rychlik I. Simulation of load sequences from rainflow matrices: Markov method. Int J Fatigue 1996;18(7):429–38.
[20] Box GEP, Jenkins GM, Reinsel GC. Time series analysis: forecasting and control. Hoboken, NJ: John Wiley & Sons; 1976.
[21] Uturbey W. Identification of ARMA models by Bayesian methods applied to streamflow data. In:2006 Int conf probabilistic methods appl to power syst, June 2006. p. 1–7.
[22] Ljung GM, Box GEP. On a measure of lack of fit in time series models. Biometrika 1978;65(2):297. August.
[23] Marrel A, Iooss B, Laurent B, Roustant O. Calculations of Sobol indices for the Gaussian process metamodel. Reliab Eng Syst Saf 2009;94(3):742–51.
[24] Owen A. Better estimation of small Sobol sensitivity indices. ACM Trans Model Comput Simul 2013;23(2):11.
[25] 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(3–4):479–91.
[26] Saltelli A, Tarantola S. On the relative importance of input factors in mathematical models. J Am Stat Assoc 2002;97(459):702–9. September.
[27] Sankararaman S, Mahadevan S. Separating the contributions of variability and parameter uncertainty in probability distributions. Reliab Eng Syst Saf 2013;112(April):187–99.
[28] Angus JE. The probability integral transform and related results. SIAM Rev 1994;36(4):652–4.
[29] Ben-Haim Y. Fatigue lifetime with load uncertainty represented by convex model. J Eng Mech 1994;120(3):445–62. March.
[30] Echard B, Gayton N, Bignonnet A. A reliability analysis method for fatigue design. Int J Fatigue 2014;59(February):292–300.
[31] Huang S, Mahadevan S, Rebba R. Collocation-based stochastic finite element analysis for random field problems. Probabilist Eng Mech 2007;22(2):194–205. April.
[32] Xi Z, Youn BD, Hu C. Random field characterization considering statistical dependence for probability analysis and design. J Mech Des 2010;132(10):101008.
[33] Matsumoto M, Nishimura T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Trans Model Comput Simul 1998;8(1):3–30. January.
[34] L'ecuyer P. Good parameters and implementations for combined multiple recursive random number generators. Oper Res 1999;47(1):159–64.
[35] Matteis A, Pagnutti S. Long-range correlation analysis of the Wichmann–Hill random number generator. Stat Comput 1993;3(2):67–70.
[36] Tierney L. Markov chains for exploring posterior distributions. Ann Stat 1994;22(4):1701–28.