A survey of surrogate modeling techniques for global sensitivity analysis in hybrid simulation

N. Tsokanas[1], R. Pastorino[2], and B. Stojadinović[1]

[1]*ETH Zurich, IBK, D-BAUG, Stefano-Franscini Platz 5, 8093 Zurich, Switzerland.*
[2]*Siemens Industry Software NV, Interleuvenlaan 68, 3001 Leuven, Belgium.*

Draft submitted to *engrXiv*

Abstract

Hybrid simulation is a method used to investigate the dynamic response of a system subjected to a realistic loading scenario. The system under consideration is divided into multiple individual loading-rate-sensitive substructures, out of which one or more are tested physically, whereas the remaining are simulated numerically. The coupling of all substructures forms the so-called hybrid model. Although hybrid simulation has been extensively used across various engineering disciplines, it is often the case that the hybrid model and related excitation is conceived as deterministic. However, associated uncertainties are present whilst simulation deviation due to their presence could be significant. To this regard, global sensitivity analysis based on Sobol' indices can be used to determine the sensitivity of the hybrid model response due to the presence of the associated uncertainties. Nonetheless, estimation of the Sobol' sensitivity indices requires unaffordable amount of hybrid simulation evaluations. Therefore, surrogate modeling techniques are used to alleviate this burden. In this paper, three different surrogate modeling methods are examined, namely polynomial chaos expansion, Kriging and polynomial chaos Kriging. A case study encompassing a virtual hybrid model is employed and hybrid model response quantities of interest are selected, and their respective surrogates are developed using all three aforementioned techniques. The Sobol' indices obtained utilizing each examined surrogate are compared with each other and results highlight potential deviations. The accuracy of the developed surrogates is also compared with each other.
Keywords: real-time hybrid simulation, surrogate modeling, polynomial chaos expansion, Kriging, polynomial chaos Kriging, global sensitivity analysis, Sobol' indices, uncertainty quantification.

1 Introduction

Hybrid simulation (HS), also known as Hardware-in-the-Loop (HiL) or model-based system testing, is a method used to investigate the dynamic response of a system subjected to a realistic loading scenario. The system under consideration is divided into multiple individual loading-rate-sensitive substructures, out of which one or more are tested physically, and thus correspond to the physical substructures (PS), whereas the remaining substructures are simulated numerically, namely the numerical substructures (NS). The coupling of PS and NS forms the so-called hybrid model. The response of the latter is obtained from a step-by-step numerical solution of the equations governing the motion of the underlying hybrid model, combined with measurements acquired from the PS. Therefore, the calculation of the next HS time step is computed online. An arrangement of transfer systems, e.g. servo-controlled motors or actuators, is used to deliver the motion between NS and PS. The employed transfer system accounts for the substructure coupling and hence for synchronizing their dynamic boundary conditions in every time step of the HS. From the coupling of substructures, several challenges arise, e.g. time delays due to inherent transfer system dynamics or due to computational power needed to compute the NS response. Advance control techniques [1, 2] and model order reduction methods [3, 4] have been used to tackle such issues. Albeit the challenges, the HS approach is beneficial since it can be used to experimentally study the inner workings of specific substructures over their linear regime and hence acquire realistic results but without constructing the entire considered system nor risking damaging it. A detailed overview of HS can be found in the report by Schellenberg and Mahin [5].

HS has been widely used across different engineering disciplines [6, 7, 8, 9, 10]. However, it is often the case that the hybrid model and related excitation are conceived as deterministic. Nonetheless, this assumption does not reflect to real-world structures and systems as loading is stochastic whereas hybrid model parameters may be highly uncertain. A thorough investigation of all possible parameter and excitation variations is not feasible considering the cost related to an individual HS evaluation. Still, uncertainties are present whilst simulation deviation due to their presence could be significant.

In this regard, surrogate modeling has been proposed to perform global sensitivity analysis (GSA) of a quantity of interest (QoI) hybrid model response with respect to a set of input parameters originating from both substructures and excitation [11, 12]. GSA aims to quantitatively determine the degree each input parameter affects the selected QoI of the hybrid model response. Sobol' indices is a popular methodology to perform variance-based GSA, in which the variance of the response QoI is decomposed as a sum of contributions related to each input variable or combination with each other [13]. Monte Carlo simulations can be used to estimate the Sobol' indices. Nonetheless, for each index estimation, $\mathcal{O}(10^3)$ model evaluations would be required [11]. To alleviate this burden, surrogate models of the QoI response can be developed. Surrogates (a.k.a. metamodels) are used to substitute an expensive-to-compute model with an inexpensive-to-compute surrogate. Therefore, Monte Carlo simulations using the developed surrogates can be performed at affordable cost and Sobol' indices estimation becomes easier to compute. Especially, for the case that a polynomial chaos expansion

(PCE) has been constructed, then the Sobol' indices can be computed analytically at no additional cost [14, 15].

In this study, three different surrogate modeling methods, namely PCE, Kriging and polynomial chaos Kriging (PCK), are used to perform GSA via Sobol' indices in HS. A case study encompassing a virtual hybrid model is employed and specific QoI of the hybrid model response are selected. Surrogates for the selected QoI are constructed and their performance is evaluated. Sobol' indices are computed using each developed surrogate and comparison with each other is made to highlight potential deviations. Results demonstrate the effectiveness of each examined surrogate modeling method.

This paper is organized as follows. Sections 2 and 3 present the background of the examined surrogate modeling methods as well as the GSA with Sobol' indices. Section 4 presents the case study with the virtual hybrid model and Section 5 discusses the respective results. Finally, Section 6 presents the overall conclusions of this study.

2 Surrogate modeling methods

Considering an input vector $\boldsymbol{X} \in \mathcal{D}_{\boldsymbol{X}} \subset \mathbb{R}^N$ and a computational model $\boldsymbol{Y} = \mathcal{M}(\boldsymbol{X})$ with $\boldsymbol{Y} \in \mathbb{R}^N$, machine-learning data-driven regression algorithms formulate a map $\mathcal{M}^s : \boldsymbol{X} \mapsto \boldsymbol{Y}$ based on an obtained sample set of input points $\boldsymbol{X} = \left\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right\}^T$ and of the respective output values, i.e. model evaluations or experimental measurements, $\boldsymbol{Y} = \left\{\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(N)}\right\}^T$. The set of $\boldsymbol{X}, \boldsymbol{Y}$ realizations corresponds to the so-called experimental design (ED).

2.1 Polynomial chaos expansion

PCE is a well-known uncertainty quantification spectral method used to substitute the dynamics of an expensive-to-compute numerical model with an inexpensive-to-compute surrogate (a.k.a. metamodel), representing the outputs of the model by a polynomial function of its inputs [16, 17]. It is proven to be a powerful surrogate technique used in a wide variety of engineering contexts, to replicate the dynamic response of complex high-dimensional models [11, 18, 19].

In more detail, given a random input vector $\boldsymbol{X}$ with independent components expressed by the joint PDF $f_{\boldsymbol{X}}$ and a finite variance computational model $Y = \mathcal{M}(\boldsymbol{X})$, such that $\mathbb{E}[Y^2] = \int_{\mathcal{D}_{\boldsymbol{X}}} \mathcal{M}^2(\boldsymbol{x}) f_{\boldsymbol{X}}(\boldsymbol{x})\, d\boldsymbol{x} < \infty$, the PCE of $\mathcal{M}(\boldsymbol{X})$ follows:

$$Y = \mathcal{M}(\boldsymbol{X}) \approx \mathcal{M}^{PCE}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^N} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}). \tag{1}$$

The PCE function is built on the $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ multivariate orthonormal polynomial basis with respect to the input vector $f_{\boldsymbol{X}}$. The degree of the $\Psi_{\boldsymbol{\alpha}}$ polynomials components is identified by the $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N), \boldsymbol{\alpha} \in \mathbb{N}^N$ multi-index for each of the input variables, while $y_{\boldsymbol{\alpha}}$ corresponds to the polynomial coefficients.

2.1.1 Polynomial basis

The multivariate polynomials are constructed as a tensor product of their univariate orthonormal polynomials $\phi_k^{(i)}(x^{(i)})$, such that:

$$\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) = \prod_{i=1}^{N} \phi_{\alpha_i}^{(i)}(x^{(i)}). \tag{2}$$

The latter meets the orthonormality criteria:

$$\left\langle \phi_j^{(i)}(x^{(i)}), \phi_k^{(i)}(x^{(i)}) \right\rangle = \int_{\mathcal{D}_{X_i}} \phi_j^{(i)}(x^{(i)}) \phi_k^{(i)}(x^{(i)}) f_{X_i}(x^{(i)})\, dx_i = \delta_{jk}, \tag{3}$$

where i corresponds to the input variable, j and k to the polynomial degree, $f_{X_i}(x^{(i)})$ to the i^{th}-input marginal PDF and δ_{jk} to the Kronecker symbol. The selection of the univariate orthonormal polynomial families depends on the marginal PDF of each input variable to which they are orthogonal, e.g. if an input variable follows the uniform/ Gaussian distribution then the Legendre/ Hermite orthogonal polynomial family is used respectively for this specific input variable [17, 20].

2.1.2 Truncation schemes

Once the univariate polynomials families are selected for each input variable, the next step is the construction of the PCE following Eq. (1). However, because the sum of Eq. (1) consists of infinite terms, it is often truncated to a finite number of terms for practical reasons. Hence, the truncated basis is defined as $\mathcal{A} \subset \mathbb{N}^N$ and the PCE of $\mathcal{M}(\boldsymbol{X})$ admits:

$$\mathcal{M}^{PCE}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}). \tag{4}$$

The performance of PCE is closely connected with the truncation scheme used. Underfitting or overfitting is possible to occur when too many terms are discarded or introduced respectively [19]. In the standard basis truncation scheme [17] the maximum degree $p \in \mathbb{N}^+$ is defined, such that the degree of each polynomial is capped by this value. Therefore, the standard basis truncation scheme consists of $\binom{N+p}{p} = \frac{(N+p)!}{N!\,p!}$ elements and follows:

$$\mathcal{A}^{N,p} = \left\{ \boldsymbol{\alpha} \in \mathbb{N}^N : |\boldsymbol{\alpha}| \leq p \right\}, \tag{5}$$

where N corresponds to the number of input variables of $\mathcal{M}(\boldsymbol{X})$ and $|\boldsymbol{\alpha}| = \sum_{i=1}^{N} \alpha_i$ to the total degree of all polynomials in $\boldsymbol{\Psi}$. To further reduce the polynomial basis size, additional truncation schemes have been developed, namely the maximum interaction and hyperbolic truncation scheme.

In the maximum interaction truncation scheme [20], the basis of Eq. (5) is reduced such that the $\boldsymbol{\alpha}$ indices to include at most r non-zero components and thus the rank of $\boldsymbol{\alpha}$ is decreased [20]. Accordingly, Eq. (5) is written as:

$$\mathcal{A}^{N,p,r} = \{\boldsymbol{\alpha} \in \mathcal{A}^{N,p} : ||\boldsymbol{\alpha}||_0 \leq r\}, \tag{6}$$

where $||\boldsymbol{\alpha}||_0 = \sum_{i=1}^{N} \mathbb{1}_{\{\alpha_i > 0\}}$ and $\mathbb{1}_{\{\alpha_i > 0\}}$ is the indicator function.

The hyperbolic (a.k.a. q-norm) truncation scheme [21] reforms the basis of Eq. (5) such that:

$$\mathcal{A}^{N,p,q} = \{\boldsymbol{\alpha} \in \mathcal{A}^{N,p} : ||\boldsymbol{\alpha}||_q \leq p\}, \tag{7}$$

where $||\boldsymbol{\alpha}||_q = \left(\sum_{i=1}^{N} a_i^q\right)^{\frac{1}{q}}$ and $q \in (0, 1]$.

2.1.3 PCE coefficient calculation

Once the truncation scheme is determined, the next step is the calculation of coefficients $\boldsymbol{y} = \{\boldsymbol{y}_{\boldsymbol{\alpha}}, \boldsymbol{\alpha} \in \mathcal{A}\}$ of Eq. (4). Various methods have been developed to do this [20]. One of these techniques is the least-squares minimization [22]. This is an non-intrusive method, meaning that the coefficients are obtained after post-processing the ED points. More specifically, Eq. (4) is reformed as:

$$\mathcal{M}^{PCE}(\boldsymbol{X}) = \sum_{j=0}^{P-1} y_j \Psi_j(\boldsymbol{X}) + \varepsilon_P = \boldsymbol{y}^T \boldsymbol{\Psi}(\boldsymbol{X}) + \varepsilon_P, \tag{8}$$

where $\boldsymbol{y} = (y_0, \ldots, y_{P-1})^T$ denote the PCE coefficients, ε_P the truncation error, $\boldsymbol{\Psi}(\boldsymbol{X}) = \{\Psi_0(\boldsymbol{X}), \ldots, \Psi_{P-1}(\boldsymbol{X})\}^T$ the multivariate orthonormal polynomials, and $P = \binom{N+p}{p}$. Then, the PCE coefficients are obtained by solving:

$$\hat{\boldsymbol{y}} = \arg \min_{\boldsymbol{y} \in \mathbb{R}^{|\mathcal{A}|}} \mathbb{E}\Big[\big(\boldsymbol{y}^T \boldsymbol{\Psi}(\boldsymbol{X}) - \mathcal{M}(\boldsymbol{X})\big)^2\Big]. \tag{9}$$

The solution of Eq. (9) is attained by Ordinary Least-Squares (OLS) and follows:

$$\hat{\boldsymbol{y}} = \left(\boldsymbol{A}^T \boldsymbol{A}\right)^{-1} \boldsymbol{A}^T \boldsymbol{Y}, \tag{10}$$

where the experimental matrix is $A_{ij} = \Psi_j(\boldsymbol{x}^{(i)})$ with $i = 1, \ldots, N$ and $j = 0, \ldots, P-1$.

However, for $N < P - 1$ the $\boldsymbol{A}^T \boldsymbol{A}$ matrix is non-invertible and thus the OLS problem cannot be solved [19, 23]. Nonetheless, through sparse regression accurate surrogates can be constructed with fewer coefficients. A way to achieve sparse regression in high-dimensions is by a modification of the least-squares minimization method. In more detail, a penalty term is introduced in Eq. (9) and the latter is modified as:

$$\hat{\boldsymbol{y}} = \arg \min_{\boldsymbol{y} \in \mathbb{R}^{|\mathcal{A}|}} \mathbb{E}\Big[\big(\boldsymbol{y}^T \boldsymbol{\Psi}(\boldsymbol{X}) - \mathcal{M}(\boldsymbol{X})\big)^2\Big] + \lambda ||\boldsymbol{y}||_1, \tag{11}$$

where $||\boldsymbol{y}||_1 = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} |y_{\boldsymbol{\alpha}}|$ and λ a parameter of the penalty term. This penalty term causes the minimization to facilitate sparse solutions. One of the widely utilized techniques, used to solve the minimization problem of Eq. (11) is the Least Angle Regression (LAR) algorithm [21]. The latter is also employed in the case study in Section 4 for the respective PCE surrogate.

2.2 Kriging

Kriging (a.k.a. Gaussian process) is a widely used surrogate modeling method. It considers the output of a model $\boldsymbol{Y} = \mathcal{M}(\boldsymbol{X})$ as a realization of a Gaussian process, with $\boldsymbol{x} \in \mathcal{D}_{\boldsymbol{X}} \subset \mathbb{R}^N$ and follows [24, 25]:

$$\mathcal{M}^{Krig}(\boldsymbol{x}) = \boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x}) + \sigma^2 Z(\boldsymbol{x}, \omega), \tag{12}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_P)$ and $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_P(\boldsymbol{x}))$ are coefficients and regression functions, respectively. The product of the latter two corresponds to the mean values of the Gaussian process, i.e. the Kriging trend, while σ^2 to its variance. $Z(\boldsymbol{x}, \omega)$ is a stationary Gaussian process with zero mean and unit variance, defined by an autocorrelation function $R\big(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}\big)$ and its hyperparameters $\boldsymbol{\theta}$. The coefficients $\boldsymbol{\beta}$ are computed from the generalized least-square solution following:

$$\beta = \left(\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F}\right)^{-1} \boldsymbol{F}^T \boldsymbol{R}^{-1} \mathbf{Y}, \tag{13}$$

where $\boldsymbol{F}_{ij} = f_j(\boldsymbol{x}^{(i)})$ and $\boldsymbol{R}_{ik} = R\big(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(k)}; \boldsymbol{\theta}\big)$ the correlation matrix with $i, k = 1, \ldots, N$ and $j = 1, \ldots, P$.

In Kriging, the model Y and surrogate $\widehat{Y}(\boldsymbol{x}) = \mathcal{M}^{Krig}(\boldsymbol{x})$ response is assumed to have a joint Gaussian distribution and therefore the mean value and variance of the prediction $\widehat{Y}(\boldsymbol{x})$ read [24, 26]:

$$\mu_{\widehat{Y}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})^T \boldsymbol{\beta} + \boldsymbol{r}(\boldsymbol{x})^T \boldsymbol{R}^{-1} \big(\boldsymbol{Y} - \boldsymbol{F}\boldsymbol{\beta}\big), \tag{14}$$

$$\sigma^2_{\widehat{Y}}(\boldsymbol{x}) = \sigma^2 \Big(1 - \boldsymbol{r}(\boldsymbol{x})^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) + \boldsymbol{u}(\boldsymbol{x})^T \left(\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F}\right)^{-1} \boldsymbol{u}(\boldsymbol{x})\Big), \tag{15}$$

where $\boldsymbol{r}_i(\boldsymbol{x}) = R\big(\boldsymbol{x}, \boldsymbol{x}^{(i)}; \boldsymbol{\theta}\big)$ consists of the cross-correlations between predictions $\boldsymbol{x}$ and sample points $\boldsymbol{x}^{(i)}$ with $i = 1, \ldots, N$ and $\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})$.

2.2.1 Trend families

The first step in constructing a Kriging surrogate is to determine its trend family. Three common family types can be found in the literature [26, 27], namely the simple Kriging expressed by $\boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x}) = \sum_{j=1}^{P} f_j(\boldsymbol{x})$, the ordinary with $\boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x}) = \beta_0$ and the universal which follows $\boldsymbol{\beta}^T \boldsymbol{f}(\boldsymbol{x}) = \sum_{j=1}^{P} \beta_j f_j(\boldsymbol{x})$. The latter is a generic type admitting a variety of formations, e.g. multivariate polynomials (see Section 2.3).

2.2.2 Autocorrelation functions

The autocorrelation functions used in Kriging metamodeling represent the relative position of two input sample points $\boldsymbol{x}$ and $\boldsymbol{x}'$. Some common autocorrelation function families [26, 27] are the linear:

$$R\big(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}\big) = \prod_{i=1}^{N} \max\left(0, 1 - \frac{|x^{(i)} - x'^{(i)}|}{\theta_i}\right), \tag{16}$$

the exponential:

$$R\big(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}\big) = \exp\left(-\sum_{i=1}^{N} \frac{|x^{(i)} - x'^{(i)}|}{\theta_i} \right), \tag{17}$$

the Gaussian:

$$R\big(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}\big) = \exp\left(-\sum_{i=1}^{N} \left(\frac{|x^{(i)} - x'^{(i)}|}{\theta_i} \right)^2 \right) \tag{18}$$

and the Matérn:

$$R\big(\boldsymbol{x}, \boldsymbol{x}'; \boldsymbol{\theta}, v\big) = \prod_{i=1}^{N} \frac{1}{2^{v-1}\Gamma(v)} \left(2\sqrt{v} \frac{|x^{(i)} - x'^{(i)}|}{\theta_i} \right)^{v} \mathcal{K}_v \left(2\sqrt{v} \frac{|x^{(i)} - x'^{(i)}|}{\theta_i} \right), \tag{19}$$

where $v \geq 1/2$ corresponds to the shape parameter, Γ is the Euler Gamma function and $\mathcal{K}_v$ is the modified Bessel function of the second kind.

2.2.3 Hyperparameters estimation

Following the selection of trend and autocorrelation family, the next step in building the Kriging surrogate consists of estimating the unknown hyperparameters $\boldsymbol{\theta}$. One of the existing methodologies is the maximum-likelihood estimation. The goal of the latter is to maximize the likelihood of model evaluations $\boldsymbol{Y}$ by determining the appropriate set of the Kriging parameters β, σ^2 and $\boldsymbol{\theta}$. The likelihood function of $\boldsymbol{Y}$ admits:

$$\mathcal{L}(\beta, \sigma^2, \boldsymbol{\theta}; \boldsymbol{Y}) = \frac{\big(\det \boldsymbol{R}\big)^{-1/2}}{\big(2\pi\sigma^2\big)^{N/2}} \exp\left(-\frac{1}{2\sigma^2} \big(\boldsymbol{Y} - \boldsymbol{F}\beta\big)^T \boldsymbol{R}^{-1} \big(\boldsymbol{Y} - \boldsymbol{F}\beta\big) \right). \tag{20}$$

The hyperparameters $\boldsymbol{\theta}$ are computed by solving the optimization problem:

$$\begin{aligned} \boldsymbol{\theta} &= \arg \min_{\boldsymbol{\theta} \in \mathcal{D}_{\boldsymbol{\theta}}} \big(-\log \mathcal{L}(\boldsymbol{\theta}; \boldsymbol{Y}) \big) \\ &= \arg \min_{\boldsymbol{\theta} \in \mathcal{D}_{\boldsymbol{\theta}}} \frac{1}{2} \big(\log(\det \boldsymbol{R}) + N \log(2\pi\sigma^2) + N \big). \end{aligned} \tag{21}$$

For a more detailed description of the derivation of Eq. (21) along with additional methods for hyperparameter estimation, e.g. cross-validation estimation, the reader should refer to [24, 27].

2.2.4 Optimization methods

The solution of Eq. (21) is obtained by employing an optimization algorithm. Two main categories of such algorithms are the local, e.g. gradient-based, and global, e.g. evolutionary algorithms, methods. Local methods need less objective functions evaluations and can converge faster. However, their performance may be comprised due to multiple local minima. Integrating characteristics from both local and global methods, results in a third category, the hybrid methods. Such a method is for example the hybrid genetic algorithm [26, 28].

The Kriging surrogate used later in the case studies of this chapter is of the ordinary trend type using the Matérn autocorrelation function family, while the hyperparameters are estimated with the maximum-likelihood method and optimized using the hybrid genetic algorithm method.

2.3 Polynomial chaos Kriging

PCK is another surrogate modeling method that combines features from PCE and Kriging into one. In more detail, PCK is a Kriging model with a universal trend type. Its trend though is formulated from sparse orthogonal polynomials. The output of PCK combines Eq. (4) and Eq. (12) and follows [29, 30, 31]:

$$\mathcal{M}^{PCK}(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha}\in\mathcal{A}} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) + \sigma^2 Z(\boldsymbol{x}, \omega). \tag{22}$$

The first term, $\sum_{\boldsymbol{\alpha}\in\mathcal{A}} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$, is the trend of the Kriging model as described in Section 2.2 and is computed from sparse PCE as described in Section 2.1. The second term of Eq. (22) corresponds to the variance of the Gaussian process σ^2 and $Z(\boldsymbol{x}, \omega)$ is a stationary Gaussian process with zero mean and unit variance as of Eq. (12).

Therefore, building a PCK involves a step of constructing the PCE and an additional one to determine the Kriging model, i.e. to estimate its parameters. Two ways of combining the above two steps are the so-called sequential and optimal approaches [29, 30, 32].

The PCK surrogate used later in the case studies of this chapter is developed using the optimal approach and the Matérn autocorrelation function family is used along with the maximum-likelihood estimation and the hybrid genetic algorithm optimization method.

2.4 Leave-one-out cross-validation error

To assess the performance of the developed surrogates the leave-one-out (LOO) cross-validation error is used, defined by:

$$\varepsilon_{LOO} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathcal{M}\big(\boldsymbol{x}^{(i)}\big) - \mathcal{M}^{s\setminus i}\big(\boldsymbol{x}^{(i)}\big)\right)^2, \tag{23}$$

where $\mathcal{M}^{s\setminus i}\big(\boldsymbol{x}^{(i)}\big)$ corresponds to the surrogates developed from the subset of the ED, acquired by removing the $\boldsymbol{x}^{(i)}$ point.

3 Global sensitivity analysis with Sobol' indices

Broadly speaking, GSA aims to quantitatively identify the level each input variable in $\boldsymbol{X} = \left\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right\}^T$, $\boldsymbol{X} \in \mathcal{D}_X \subset \mathbb{R}^N$ affects the response of $Y = \mathcal{M}(\boldsymbol{X})$, $Y \in \mathbb{R}$. As mentioned previously, in this dissertation, the Sobol' indices are used for GSA. It should be noted that this method is only valid for independent input variables.

3.1 Sobol'-Hoeffding decomposition

For $\boldsymbol{X} \in \mathcal{D}_X \subset \mathbb{R}^N$ with independent components, $Y = \mathcal{M}(\boldsymbol{X})$ with $\mathrm{Var}\left[Y\right] < +\infty$ can be rewritten as [33, 11]:

$$\begin{aligned}\mathcal{M}^s(\boldsymbol{X}) &= \mathcal{M}_0 + \sum_{i=1}^{N} \mathcal{M}_i(x^{(i)}) + \sum_{1 \le i \le j \le N} \mathcal{M}_{ij}(x^{(i)}, x^{(j)}) + \cdots + \mathcal{M}_{1,2,\ldots,N}(\boldsymbol{X}) \\ &= \mathcal{M}_0 + \sum_{\substack{\mathbf{u} \subset \{1,\ldots,N\} \\ \mathbf{u} \neq \emptyset}} \mathcal{M}_{\boldsymbol{u}}(\boldsymbol{X}_{\boldsymbol{u}}),\end{aligned} \tag{24}$$

where $\mathcal{M}_0$ is constant and denotes the mean value of Y, $\mathbf{u} \neq \emptyset$, $\mathbf{u} \subset \{1, \ldots, N\}$ and $\boldsymbol{X}_{\boldsymbol{u}}$ is a sub-vector of $\boldsymbol{X}$, including the $\boldsymbol{u}$-indexed elements. The summands in the latter equation corresponds to 2^{N-1} terms. The elementary functions are defined by conditional expectations:

$$\mathcal{M}_{\mathbf{u}}(\boldsymbol{X}_{\mathbf{u}}) = \sum_{\mathbf{v} \subset \mathbf{u}} (-1)^{|\mathbf{u}|-|\mathbf{v}|} \mathbb{E}\left[Y \mid \boldsymbol{X}_{\mathbf{v}} = \boldsymbol{x}_{\mathbf{v}}\right], \tag{25}$$

where $|\mathbf{u}|$ gives the cardinality of $\mathbf{u}$. When the following holds:

$$\int_{\mathcal{D}_{X_m}} \mathcal{M}_{\boldsymbol{u}}(\boldsymbol{X}_{\boldsymbol{u}}) f_{X_m}(x_m)\, dx_m = 0, \text{ if } m \in \boldsymbol{u}, \tag{26}$$

the Sobol'-Hoeffding decomposition of Eq. (24) is unique. Due to this uniqueness, the orthogonality property:

$$\mathbb{E}\big[\mathcal{M}_{\boldsymbol{u}}(\boldsymbol{X}_{\boldsymbol{u}}) \cdot \mathcal{M}_{\boldsymbol{v}}(\boldsymbol{X}_{\boldsymbol{v}})\big] = 0, \text{ if } \boldsymbol{u} \neq \boldsymbol{v}, \tag{27}$$

also holds. Because of Eq. (26) and (27) the total variance of $\mathcal{M}(\boldsymbol{X})$ decomposes as:

$$D = \mathrm{Var}\left[\mathcal{M}(\boldsymbol{X})\right] = \sum_{\substack{\mathbf{u} \subset \{1,\ldots,N\} \\ \mathbf{u} \neq \emptyset}} D_{\boldsymbol{u}}, \tag{28}$$

where

$$D_{\boldsymbol{u}} = \mathrm{Var}\left[\mathcal{M}_{\boldsymbol{u}}(\boldsymbol{X}_{\boldsymbol{u}})\right] = \mathbb{E}\big[\mathcal{M}_{\boldsymbol{u}}^2(\boldsymbol{X}_{\boldsymbol{u}})\big], \tag{29}$$

corresponds to the partial variance.

3.2 Sobol' indices

The Sobol' index $S_{\boldsymbol{u}}$ is determined as the ratio between the partial and total variance that represents the $\boldsymbol{u}$-indexed set of input variables and follows:

$$S_{\boldsymbol{u}} = \frac{D_{\boldsymbol{u}}}{D}. \tag{30}$$

For $|\mathbf{u}| = 1$, the separate impact of each input variable $x^{(i)}$ to the response $Y = \mathcal{M}(\boldsymbol{X})$ is described by the indices respective to this specific variable, namely the first-order indices $S_i^{(1)} = D_i/D$. The impact originating from the interaction of pairs of the input variables $(x^{(i)}, x^{(j)})$ and not already accounted in the S_i, S_j is described by the second-order indices $S_{ij}^{(2)} = D_{ij}/D$. The impact from interaction of a larger set of input variables is described from the higher-order indices. Total indices S_{T_i} describe the overall impact of each input variable, considering its first-order index and all interactions with the other input variables and admit:

$$S_{T_i} = 1 - S_{-i}, \tag{31}$$

where S_{-i} denotes the sum of all $S_{\boldsymbol{u}}$ of $\boldsymbol{u}$ but i. It follows that $\sum_{\boldsymbol{u} \neq \emptyset} S_{\boldsymbol{u}} = 1$.

3.3 Sobol' indices estimation

3.3.1 Monte Carlo-based estimation

The variances of Eq (28) and (29) can be obtained by estimators from Monte Carlo simulations. These read [34]:

$$\begin{aligned}
\hat{\mathcal{M}}_0 &= \frac{1}{N}\sum_{i=1}^{N} \mathcal{M}^s\big(x^{(i)}\big) \\
\hat{D} &= \frac{1}{N}\sum_{i=1}^{N} \mathcal{M}^s\big(x^{(i)}\big)^2 - \hat{\mathcal{M}}_0^2 \\
\hat{D}_{\boldsymbol{u}} &= \frac{1}{N}\sum_{i=1}^{N} \mathcal{M}^s\big(x_n^{(i)}, x_{\sim n}^{(i)}\big)\mathcal{M}^s\big(x_n^{(i)}, x_{\sim n}^{\prime(i)}\big) - \hat{\mathcal{M}}_0^2,
\end{aligned} \tag{32}$$

where $x_{\sim n}^{(i)}$ denotes the i-th realization of x that does not involve the n input variable and x' corresponds to a realization of $\boldsymbol{X}$ which is independent of $x = \big(x_n^{(i)}, x_{\sim n}^{(i)}\big)^T$. Once the total and partial variance estimators are computed, the Sobol' indices can be derived from Eq. (30) and (31).

3.3.2 Sobol' indices from polynomial chaos expansion

Recall from Eq. (4), that the PCE of $\mathcal{M}(\boldsymbol{X})$ admits:

$$\mathcal{M}^{PCE}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}). \tag{33}$$

Due to the orthonormality of the PCE basis, the mean and variance of $\mathcal{M}^{PCE}(\boldsymbol{X})$ can be computed analytically from the $\boldsymbol{y}$ coefficients at no extra cost and follow [15]:

$$\hat{\mu} = \mathbb{E}\big[\mathcal{M}^{PCE}(\boldsymbol{X})\big] = \mathbb{E}\left[\sum_{\boldsymbol{\alpha}\in\mathcal{A}} y_{\boldsymbol{\alpha}}\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right] = y_0 \tag{34}$$

$$\hat{\sigma}^2 = \mathrm{Var}\left[\mathcal{M}^{PCE}(\boldsymbol{X})\right] = \mathbb{E}\left[\Big(\mathcal{M}^{PCE}(\boldsymbol{X}) - y_0\Big)^2\right] = \sum_{\substack{\boldsymbol{\alpha}\in\mathcal{A}\\ \boldsymbol{\alpha}\neq\boldsymbol{0}}} y_{\boldsymbol{\alpha}}^2. \tag{35}$$

The above equations hold since $\Psi_0 \equiv 1$. Therefore, for the case that a PCE of Y is already constructed, the Sobol'-Hoeffding decomposition of Eq. (24) can be rewritten as:

$$\mathcal{M}^{PCE}(\boldsymbol{X}) = y_0 + \sum_{\substack{\mathbf{u}\subset\{1,\ldots,N\}\\ \mathbf{u}\neq\emptyset}} \mathcal{M}_{\boldsymbol{u}}^{PCE}(\boldsymbol{X}_{\boldsymbol{u}}), \tag{36}$$

where $\mathcal{A}_{\boldsymbol{u}} = \{\boldsymbol{\alpha} \in \mathcal{A} : \alpha_m \neq 0$ if and only if $m \in \boldsymbol{u}\}$ corresponds to the set of multi-indices including only $\boldsymbol{u}$ and $\mathcal{M}_{\boldsymbol{u}}^{PCE}(\boldsymbol{X}_{\boldsymbol{u}}) = \sum_{\boldsymbol{\alpha}\in\mathcal{A}_{\boldsymbol{u}}} y_{\boldsymbol{\alpha}}\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}_{\boldsymbol{u}})$, as of Eq. (33). Thus, because the Sobol'-Hoeffding decomposition is unique, there exists an analytical representation of $\mathcal{M}_{\boldsymbol{u}}^{PCE}$ [14]. Accordingly to Eqs. (34), (35), the total and partial variances of $\mathcal{M}^{PCE}(\boldsymbol{X})$ admit:

$$\begin{aligned} D &= \mathrm{Var}\left[\mathcal{M}^{PCE}(\boldsymbol{X})\right] = \sum_{\boldsymbol{\alpha}\in\mathcal{A}} y_{\boldsymbol{\alpha}}^2 \\ D_{\boldsymbol{u}} &= \mathrm{Var}\left[\mathcal{M}_{\boldsymbol{u}}^{PCE}(\boldsymbol{X})\right] = \sum_{\boldsymbol{\alpha}\in\mathcal{A}_{\boldsymbol{u}}} y_{\boldsymbol{\alpha}}^2. \end{aligned} \tag{37}$$

Hence, the first-order and total Sobol' indices are obtained from:

$$\begin{aligned} S_i^{(1)} &= \frac{\sum_{\boldsymbol{\alpha}\in\mathcal{A}_i} y_{\boldsymbol{\alpha}}^2}{\sum_{\boldsymbol{\alpha}\in\mathcal{A}} y_{\boldsymbol{\alpha}}^2}, \quad \mathcal{A}_i = \big\{\boldsymbol{\alpha}\in\mathcal{A} : \alpha_i > 0,\ \alpha_{i\neq j} = 0\big\} \\ S_{T_i} &= \frac{\sum_{\boldsymbol{\alpha}\in\mathcal{A}_i^T} y_{\boldsymbol{\alpha}}^2}{\sum_{\boldsymbol{\alpha}\in\mathcal{A}} y_{\boldsymbol{\alpha}}^2}, \quad \mathcal{A}_i^T = \big\{\boldsymbol{\alpha}\in\mathcal{A} : \alpha_i > 0\big\}, \end{aligned} \tag{38}$$

respectively. Therefore, the Sobol' indices can be obtained as a by-product of the PCE coefficients at no extra cost [14].

4 Case study

This case study employs a virtual hybrid model to conduct the respective RTHS. Therefore, a virtual PS (vPS) is utilized instead of physical specimen and the overall RTHS is performed numerically, resulting thus in a virtual RTHS (vRTHS).

The hybrid model represents a prototype motorcycle. It consists of one vPS, the electronically-controlled continuously variable transmission (eCVT) of the motorcycle, and four NS, namely i) the engine, ii) the motorcycle body dynamics, iii) the rear wheel braking system and iv) the front wheel braking system. Figure 1 depicts how the NS and vPS are interconnected, forming the hybrid model.

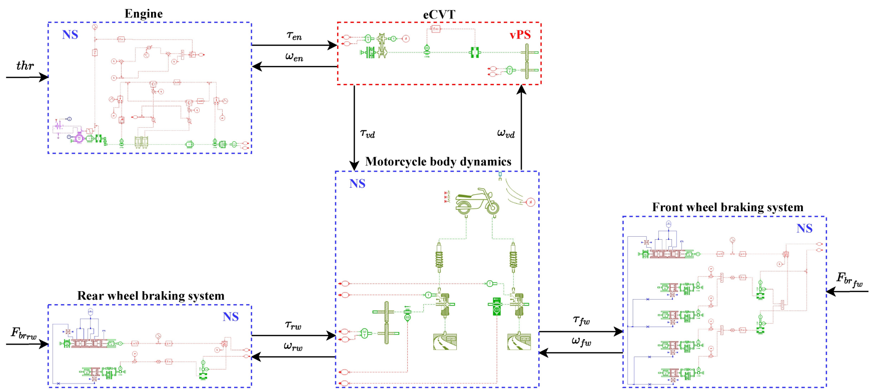


Figure 1. Motorcycle hybrid model block diagram.

The eCVT vPS is a multiple-input-multiple-output (MIMO) model including two sets of one input and one output each. The first set is interconnected to the motorcycle engine NS, which simulates the dynamics of the combustion engine, while the second set to the motorcycle body NS. The latter connection represents the motorcycle's transmission output shaft. The motorcycle body NS includes the motorcycle's body/chassis dynamics along with the dynamics of the wheels, tires and suspensions. The profile of the road as well as the environmental driving conditions are also included in this NS. The engine NS is expressed by a multiple-input-single-output (MISO) model. Its inputs are the throttle percentage thr and the angular velocity of the engine ω_{en}, whereas its output is the torque of the engine τ_{en}. The motorcycle body NS is expressed by a MIMO model, which includes three sets of one input and one output each. The first set of input/output is the torque τ_{vd} and angular velocity ω_{vd} of the transmission output shaft, respectively. The last two sets are connected to the rear and front wheel braking systems NS, respectively. The latter NS are both MISO models. In detail, the inputs of the rear wheel braking system NS are the angular velocity of the rear wheel ω_{rw} and the applied force on the brake pedal $F_{br_{rw}}$, whereas its output is the braking torque of the rear wheel τ_{rw}. Accordingly, the inputs of the front wheel braking system NS are the angular velocity of the front wheel ω_{fw} and the applied force on the brake lever $F_{br_{fw}}$, whereas its output is the braking torque of the front wheel τ_{fw}. The aforementioned substructures are developed using the multi-physics simulation software *Simcenter Amesim*. The report of [35] provides a detailed description of the development of these substructures, explaining the equations that govern their motion. Figure 2 depicts the real eCVT PS, located at the testing facilities of Siemens Industry Software, that would be utilized in a non-virtual RTHS.

To examine the performance of the motorcycle prototype, testing of its virtual hybrid model under predefined driving, road and wind scenarios takes place. To co-simulate the employed substructures and to coordinate the overall RTHS algorithm, the *Simcenter real-time platform* was utilized. For the time integration scheme of the RTHS, the fourth-order Runge–Kutta (RK4) method is used with a fixed time-step of 0.1 msec.

The simulation duration of the case study is 45 sec long, on a given driving scenario and wind/road conditions. In particular, the driving scenario is defined by the variations of the throttle thr and of the forces applied to the brakes. It is assumed that the force applied in the brake lever is equal to the force applied in the brake pedal of the motorcycle in each time-step of the RTHS, namely $F_{br_{rw}} = F_{br_{fw}}$. Eqs. (39) and (40) describe the values that the throttle and braking forces obtain, in each time-step of the simulation, while Figure 3 illustrates their variations in time. It should be noted that the maximum applied throttle is 0.5 (50%), which corresponds to a half-open throttle, and the braking forces are expressed in Newton. Eq. (41) describes the road profile, namely the height of the ground, that is assumed in this case study. It is expressed by the sinusoidal function $h(x)$, where x denotes the current position of the motorcycle in meters. Additionally, the ambient wind velocity is considered to be zero.

$$thr(t) = \begin{cases} 0.125t & , 5 \leq t < 9 \\ 0.5 & , 9 \leq t \leq 13 \\ -0.125t & , 13 < t \leq 17 \\ 0 & \text{, elsewhere} \end{cases} \tag{39}$$

$$F_{br_{rw}}(t) = \begin{cases} 10t & , 20 \leq t < 25 \\ 50 & , 25 \leq t \leq 32 \\ -10t & , 32 < t \leq 37 \\ 0 & \text{, elsewhere} \end{cases} \tag{40}$$

$$h(x) = \begin{cases} 0 & , 0 \leq x \leq 2 \\ 0.02\Big(\cos\left(\frac{2\pi x}{3}\right) - 1\Big) & \text{, elsewhere} \end{cases} \tag{41}$$

For the GSA, two hybrid model response QoI are selected, namely the maximum v_{max} and mean v_{mean} values of the motorcycle velocity. Both QoI are expressed in km/h. Scalar quantities from the dynamic response of the motorcycle's velocity were chosen in particular, since the latter is a global response parameter that can characterize the overall dynamic behavior of the hybrid model of the prototype motorcycle. In addition, the input variables to GSA are the 11 parameters which are listed in Table 1. The probability distributions that are assigned to each parameter along with their respective moments are also presented in Table 1. The bounds of each parameter were identified from [36, 37, 38] to reflect a range of possible parameter variations of the corresponding motorcycle components. According to the notation presented in Sections 2 and 3, the experimental design (ED) vectors for this case study follow:

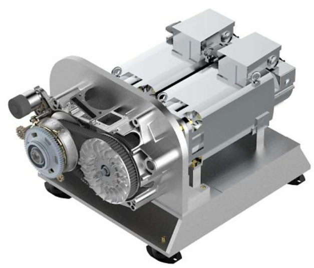


Figure 2. The eCVT test bench at the testing facilities of Siemens Industry Software.

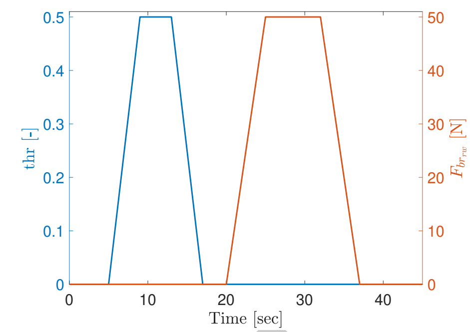


Figure 3. Case study driving scenario.

$$\boldsymbol{X} = \{K_{rt}, Z_{rt}, K_{ft}, Z_{ft}, K_{rs}, Z_{rs}, K_{fs}, Z_{fs}, M, J, \mu\}$$
$$\boldsymbol{Y} = \{v_{\text{max}}, v_{\text{mean}}\}, \tag{42}$$

From the input parameters of Table 1, 50 points were generated using the Latin hypercube sampling (LHS) method [39] and accordingly 50 vRTHS were conducted, collecting the QoI values in the $\boldsymbol{Y}$ vector of Eq. 42. With the obtained data in ED$\{\boldsymbol{X}, \boldsymbol{Y}\}$, a PCE, a Kriging and a PCK surrogate was developed for each QoI, as described in Sections 2.1, 2.2 and 2.3, respectively. Figure 4 illustrates a sample RTHS time history response of the hybrid model, indicating the QoI of the case study, which are highlighted in red color. Additionally, Figure 5 depicts some indicative dynamic responses of the variables that couple the substructures in the virtual hybrid model. The responses in the latter two figures are obtained using the mean values of Table 1, while the hybrid model is excited with the driving scenario and the road profile and wind conditions that are described by Eqs. (39)-(41).

In addition, the time history responses of Figures 4 and 5 can be intuitively interpreted. Figure 5b shows the response of the front wheel braking torque τ_{fw}, generated by the utilized driving scenario. Recall from Figure 3 and Eq. (40), that the brakes of the motorcycle are enabled in the time interval between the 20^{th} and 37^{th} second of the simulation and as a result the braking torques are non-zero only then. According to Figure 3 and Eq. (39), the throttle applied in the motorcycle's engine in the five first seconds of the simulation is zero and therefore its velocity is also zero (Figure 4). In the time interval between the 5^{th} and 17^{th} second of the simulation, the throttle starts to increases and hence the motorcycle starts to accelerate. From Figure 4 it can be seen that the velocity is also increasing. After the 20^{th} second of the simulation the braking is starting (Eq. (40)) and as a result, the velocity of the motorcycle begins to decrease, while it is brought to almost a full stop in the 45^{th} second of the simulation.

Param.	Prob. Distrib.	Mean Value	Stand. Dev.	CV (%)	Parameter Description	Units
K_{rt}	Unif.	58570	11714	20	Vertical stiffness rear tire	$\frac{N}{m}$
Z_{rt}	Unif.	11650	3495	30	Vertical damping rear tire	$\frac{Ns}{m}$
K_{ft}	Unif.	25000	5000	20	Vertical stiffness front tire	$\frac{N}{m}$
Z_{ft}	Unif.	2134	640.2	30	Vertical damping front tire	$\frac{Ns}{m}$
K_{rs}	Unif.	125000	25000	20	Stiffness rear suspension	$\frac{N}{m}$
Z_{rs}	Unif.	10000	3000	30	Damping rear suspension	$\frac{Ns}{m}$
K_{fs}	Unif.	19000	3800	20	Stiffness front suspension	$\frac{N}{m}$
Z_{fs}	Unif.	1250	375	30	Damping front suspension	$\frac{Ns}{m}$
M	Unif.	300	15	5	Motorcycle mass	Kg
J	Unif.	0.0115	0.0023	20	Engine moment of inertia	Kgm^2
μ	Unif.	0.0012	0.00018	15	Engine coefficient of viscous friction	$\frac{\text{Nm}}{\text{rev/min}}$

Table 1. Input variables of the hybrid model for GSA and their probability distributions.

5 Results and discussions

Figure 6 displays the convergence of the LOO errors of the PCE, PCK and Kriging surrogates for both QoI. The LOO errors are computed using EDs of increasing size from 2 to 50 samples. From Figure 6, it can be appreciated that the LOO errors of PCE and PCK are quite close, while the error of Kriging is slightly larger. Yet, all the errors are negligibly small as they converge to values in the magnitude of 10^{-3} or smaller. The satisfying performance of the surrogates can be also confirmed from Figure 7, which illustrates validation scatter plots by comparing the measurements of the two QoI with the respective predictions from the corresponding PCE, PCK and Kriging surrogates. In more detail, for each QoI, the surrogates are trained on 40 samples of the ED and the remaining 10 samples are used to validate them.

In addition, Figures 9 and 10 depict the convergence plots of the PCE-based moments estimates, i.e. mean, standard deviation and CV values, for both QoI. Evidently, each PCE-based moment estimate is converging in a stable manner for EDs larger than 20 samples. Recall that the PCE-based moments estimates can be computed from the PCE coefficients as post-process at no additional computational cost [14]. However, the latter is not a feature of Kriging or PCK. Thus, Figures 9 and 10 do not include moments

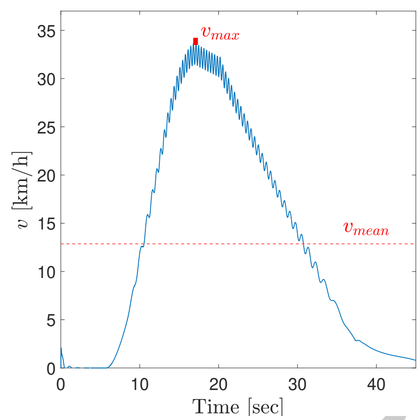


Figure 4. Sample RTHS time history response of the motorcycle prototype hybrid model, using the mean values of Table 1. The QoI are highlighted in red color.

estimates from these two surrogates. However, to further evaluate the performance of the developed surrogates, Figure 8 compares the prediction PDF of each surrogate against the histograms of the measurements for each QoI. The surrogate predictions are evaluated based on 50 random samples, generated from the distributions described in Table 1 using the LHS method. From Figure 8 it can be acknowledged that all three developed surrogates can capture quite well both the mean value and the variance of the corresponding measurements.

From Figures 6, 7 and 8 it can be appreciated that the performance and accuracy prediction of PCE, PCK and Kriging are similar. However, it should be noted that the computational time associated to train each surrogate differs. For the case of PCE and Kriging, the training computational time was approximately 1.3 and 1.6 sec, respectively. Nevertheless, for the case of PCK, it was approximately 12.7 sec. The latter finding can be intuitively interpreted as in the PCK case, the training phase consists of two steps; one constructing the PCE and a second one to determine the Kriging model. Note that these timings depend heavily on the ED sample size as well as on the settings each surrogate is developed.

Once the surrogates of each QoI are developed, the Sobol' indices can be computed. For the case of PCE, the Sobol' indices are computed analytically at no extra cost [14], as described in Section 3.3.2. For the case of PCK and Kriging though, this is not possible and thus the Sobol' indices are computed using Monte Carlo evaluations of the surrogate response QoI, as described in Section 3.3.1. It should be noted that more than 10^4 Monte Carlo evaluations of the Kriging and PCK surrogates are needed to ensure convergence of the Sobol' indices. In this case study, 10^5 evaluations of the response QoI of the Kriging and PCK surrogates were performed to compute the respective Sobol' indices. However, it is worth mentioning that the computational cost of running these evaluations was negligibly small (≈ 1 sec for each surrogate).

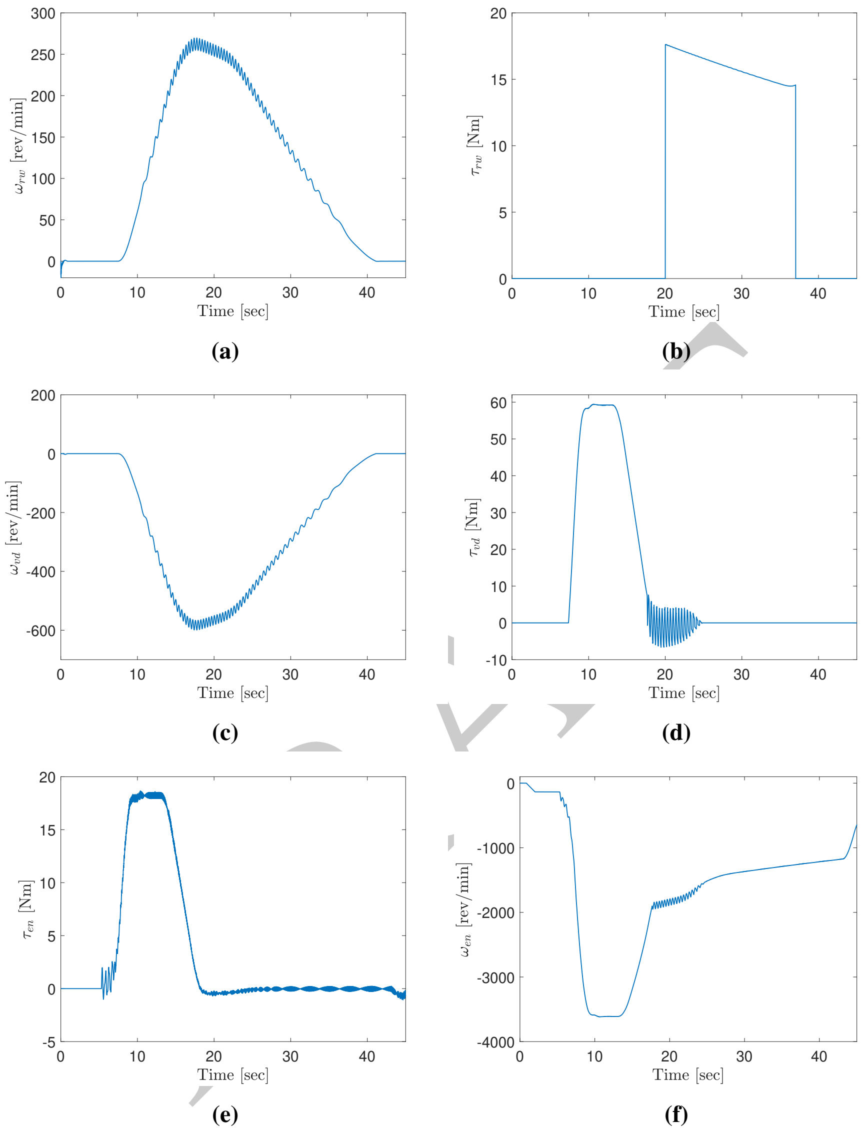
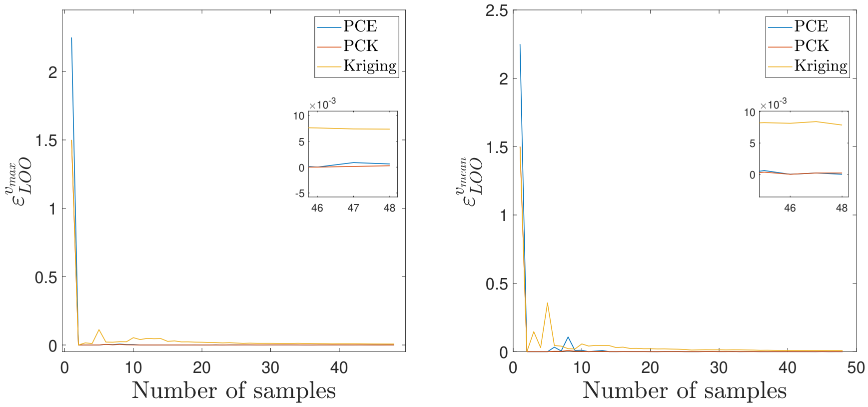


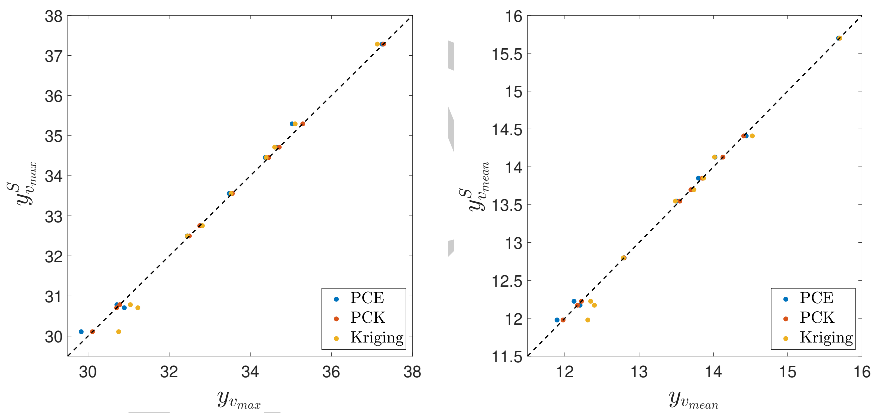
Figure 5. Indicative hybrid model responses, using the mean values of Table 1: **(a)** angular velocity of the rear wheel, **(b)** braking torque of the rear wheel, **(c)** angular velocity of the transmission output shaft, **(d)** torque of the transmission output shaft, **(e)** torque of the engine and **(f)** angular velocity of the engine.

In Figure 11 the first-order and total Sobol' indices for both QoI v_{max} and v_{mean} are reported. Clearly, both first-order and total Sobol' indices computed utilizing the different examined surrogates are admittedly close with each other. This confirms



(a) **(b)**

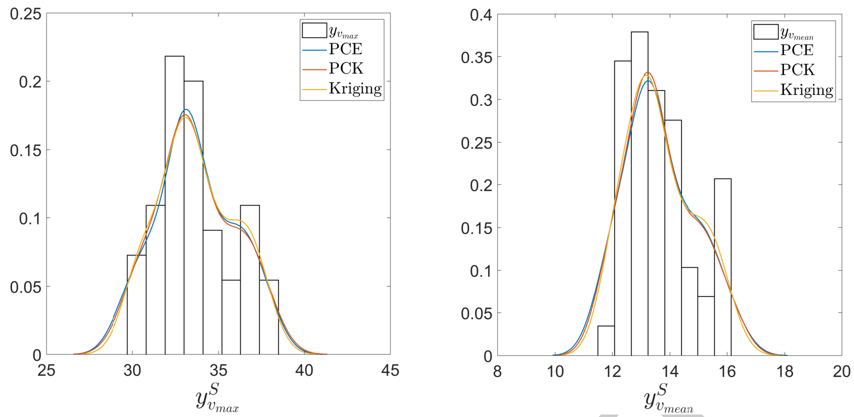
Figure 6. Convergence of LOO error estimates of (a) v_{max} and (b) v_{mean} surrogates.



(a) **(b)**

Figure 7. Comparison between QoI measurements and corresponding surrogate predictions for (a) v_{max} and (b) v_{mean}.

that the indices obtained from the PCK and Kriging converge to the analytical values computed from the PCE coefficients. From Figure 11 it can be stated that the input variables related to the motorcycle mass M as well as to the engine coefficient of viscous friction μ, are the most sensitive variables for both QoI. In particular, for the QoI related to the maximum motorcycle velocity, namely v_{max}, the motorcycle mass has a greater effect than the engine friction coefficient. Whereas, for the QoI related to the mean motorcycle velocity, namely v_{mean}, the engine friction coefficient contributes



(a) **(b)**

Figure 8. Comparison between histogram measurements and corresponding surrogate estimates for (a) v_{max} and (b) v_{mean}.

more than the motorcycle mass. On the contrary, the remaining input variables, namely the stiffness and damping of the front and rear tire and suspension as well as the engine moment of inertia, do not affect the selected QoI. Furthermore, since the first-order and total Sobol' indices for both QoI are quite close, it can be noticed that the effect from high-order interactions between the input variables is negligible.

Finally, as mentioned before, in this case study surrogates with negligible small validation error were developed with ED size larger than 20 samples. However, it should be noted that the investigated case study was numerical and hence no measurement noise was present in the results. As reported in [11], for the case that large measurement noise is present in the obtained data, it is possible that ED of larger size would be needed to train surrogates with an accepted validation error.

The development and implementation of the surrogate modelling, as well as the GSA, was performed with the UQLab software framework developed by the Chair of Risk, Safety and Uncertainty Quantification in ETH Zurich [40].

6 Conclusions

This paper serves as a survey of investigating how different surrogate modeling methods can be used to perform global sensitivity analysis via Sobol' indices in hybrid simulation. Global sensitivity analysis using Sobol' indices is established as a powerful tool to unveil the sensitivity of a structural system subjected to parameter and loading variations. However, estimation of the Sobol' indices through Monte Carlo simulations of the original hybrid model is rarely affordable, since several hybrid model evaluations are essential, while each evaluation relies on a single hybrid simulation. To alleviate this burden surrogate models of selected hybrid model response quantities of interest

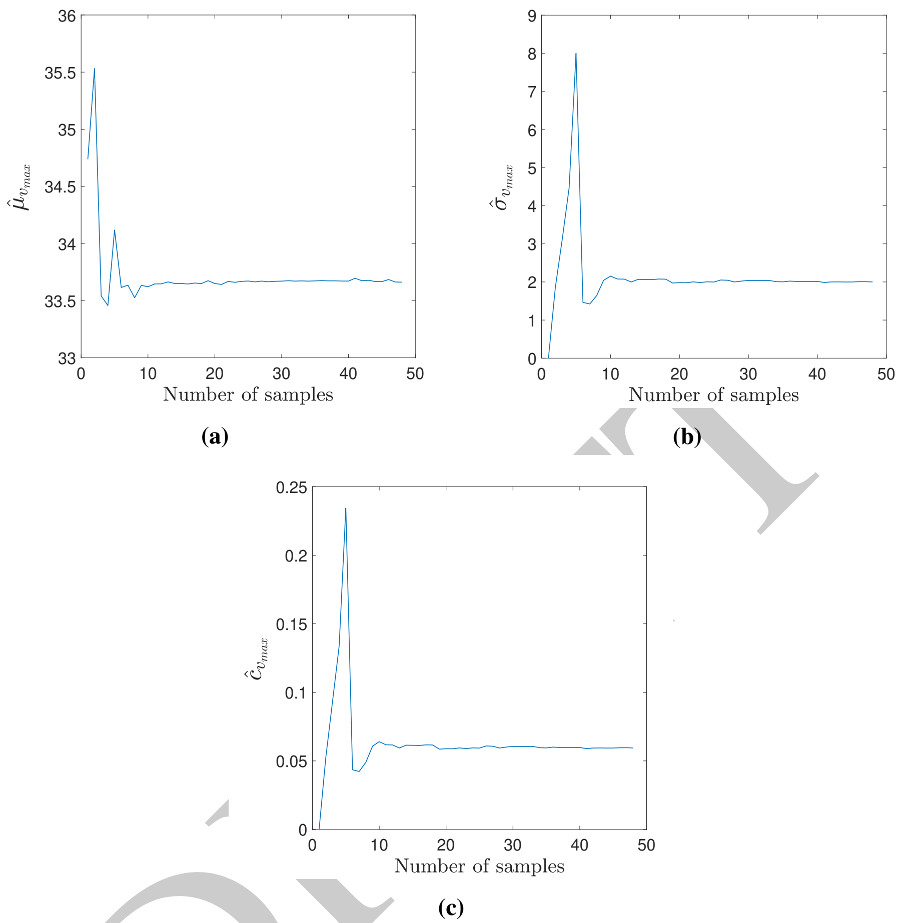


Figure 9. Convergence of PCE-based moments estimates of (a) mean, (b) standard deviation and (c) CV of QoI v_{max}.

can be constructed and used to estimate the Sobol' indices at much lower cost. A case study encompassing a virtual hybrid model is employed to investigate this approach by using a variety of different surrogate modeling techniques. In this regard, polynomial chaos expansion, Kriging (a.k.a Gaussian process) and polynomial chaos Kriging are used to develop surrogates of the selected quantities of interest. For the case of polynomial chaos expansion, the Sobol' indices are computed as a by-product of the polynomial coefficients and hence no additional computational cost is required. However, the latter is not a feature of Kriging and polynomial chaos Kriging. Therefore, Monte Carlo simulations using these surrogates are performed to estimate the Sobol' indices. Performance comparisons between each developed surrogate indicated that the respective validation errors were similar. Additionally, negligibly small deviations

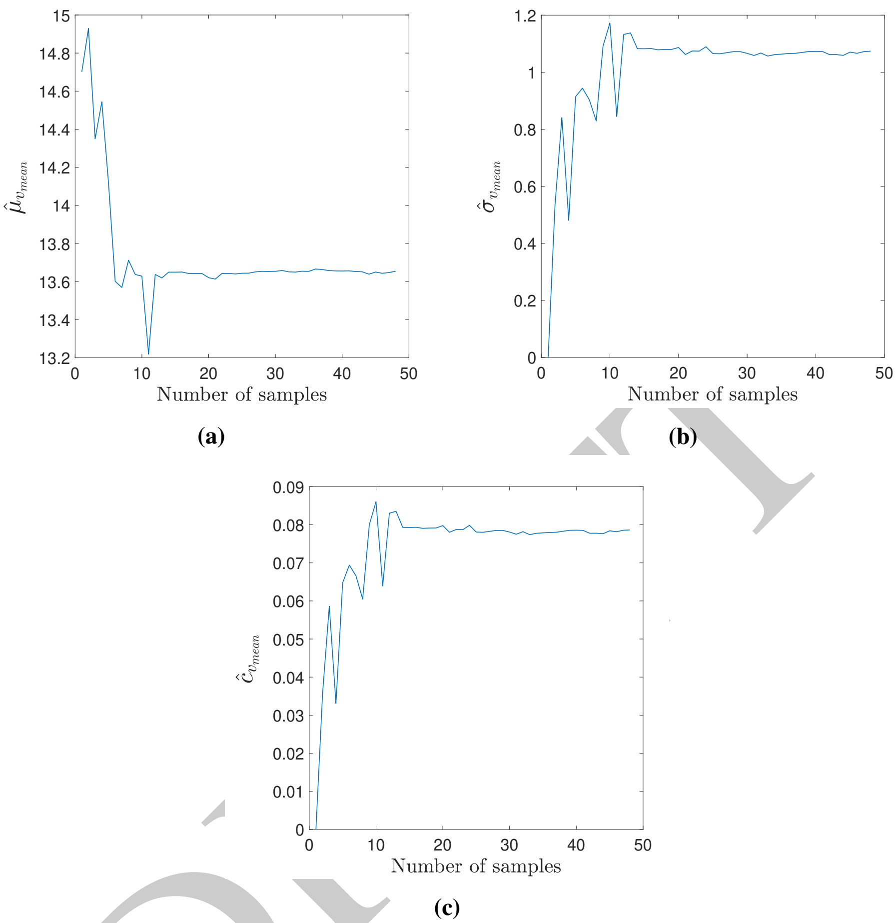


Figure 10. Convergence of PCE-based moments estimates of (a) mean, (b) standard deviation and (c) CV of QoI v_{mean}.

were observed in the Sobol' indices that were obtained using the different surrogate modeling techniques. Future work aims at adaptive sampling of the input parameter space of the hybrid model to minimize the experimentation cost necessary to compute an accurate surrogate model for global sensitivity analysis.

7 Data Availability Statement

All data and code that support the findings of this study are available from the corresponding author upon reasonable request.

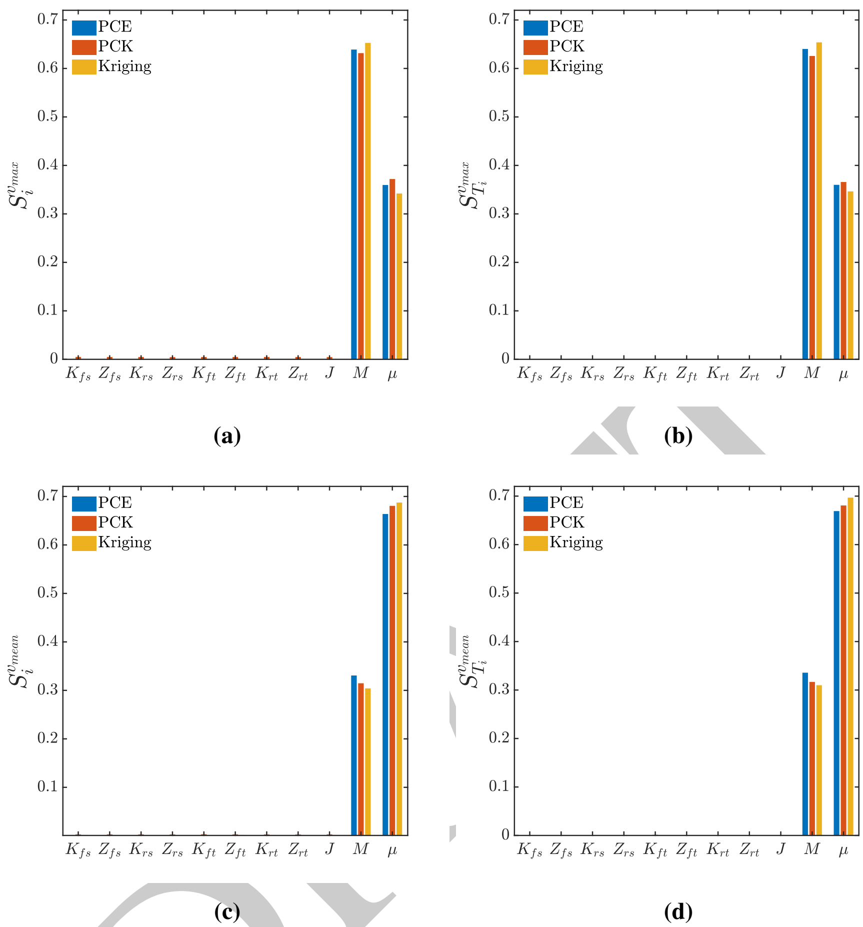


Figure 11. First-order and total Sobol' indices for (a-b) v_{max} and (c-d) v_{mean}.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 764547. The sole responsibility of this publication lies with the author(s). The European Union is not responsible for any use that may be made of the information contained herein.

References

[1] N. Tsokanas, D. Wagg, B. Stojadinović, Robust Model Predictive Control for Dynamics Compensation in Real-Time Hybrid Simulation, Frontiers in Built Environment 6 (2020).

[2] N. Tsokanas, R. Pastorino, B. Stojadinovic, Adaptive model predictive control for actuation dynamics compensation in real-time hybrid simulation, Preprint submitted in engrXiv (Aug. 2021). `doi:10.31224/osf.io/c974v`.
URL `https://engrxiv.org/c974v/`

[3] N. Tsokanas, T. Simpson, R. Pastorino, E. Chatzi, B. Stojadinovic, Model Order Reduction for Real-Time Hybrid Simulation: Comparing Polynomial Chaos Expansion and Neural Network methods, Preprint submitted in engrXiv (Aug. 2021). `doi:10.31224/osf.io/h2bnm`.

[4] G. Miraglia, M. Petrovic, G. Abbiati, N. Mojsilovic, B. Stojadinovic, A model-order reduction framework for hybrid simulation based on component-mode synthesis, Earthquake Engineering & Structural Dynamics 49 (8) (2020) 737–753. `doi:10.1002/eqe.3262`.

[5] A. H. Schellenberg, S. A. Mahin, G. L. Fenves, Advanced Implementation of Hybrid Simulation, Tech. Rep. PEER 2009/104, Pacific Earthquake Engineering Research Center, University of California, Berkeley (2009).

[6] G. Abbiati, I. Lanese, E. Cazzador, O. S. Bursi, A. Pavese, A computational framework for fast-time hybrid simulation based on partitioned time integration and state-space modeling, Structural Control and Health Monitoring 26 (10) (2019) e2419.

[7] G. Abbiati, P. Covi, N. Tondini, O. S. Bursi, B. Stojadinović, A Real-Time Hybrid Fire Simulation Method Based on Dynamic Relaxation and Partitioned Time Integration, Journal of Engineering Mechanics 146 (9) (2020) 04020104.

[8] W. Song, C. Sun, Y. Zuo, V. Jahangiri, Y. Lu, Q. Han, Conceptual Study of a Real-Time Hybrid Simulation Framework for Monopile Offshore Wind Turbines Under Wind and Wave Loads, Frontiers in Built Environment 6 (2020).

[9] S. Idinyang, A. Franza, C. M. Heron, A. M. Marshall, Real-time data coupling for hybrid testing in a geotechnical centrifuge, International Journal of Physical Modelling in Geotechnics 19 (4) (2018) 208–220, publisher: ICE Publishing.

[10] N. Tsokanas, G. Abbiati, K. Kanellopoulos, B. Stojadinovic, Multi-Axial Hybrid Fire Testing based on Dynamic Relaxation, Preprint submitted in engrXiv (Sep. 2021). `doi:10.31224/osf.io/r8xau`.
URL `https://engrxiv.org/r8xau/`

[11] G. Abbiati, S. Marelli, N. Tsokanas, B. Sudret, B. Stojadinović, A global sensitivity analysis framework for hybrid simulation, Mechanical Systems and Signal Processing 146 (2021).

[12] N. Tsokanas, X. Zhu, G. Abbiati, S. Marelli, B. Sudret, B. Stojadinovic, A global sensitivity analysis framework for hybrid simulation with stochastic substructures, Preprint submitted in engrXiv (2021). doi:10.31224/osf.io/kjnec.
URL https://engrxiv.org/kjnec/

[13] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, S. Tarantola, Global Sensitivity Analysis. The Primer, Vol. 76, John Wiley & Sons, Ltd, Chichester, UK, 2007.

[14] B. Sudret, Global sensitivity analysis using polynomial chaos expansions, Reliability Engineering & System Safety 93 (7) (2008) 964–979. doi:10.1016/j.ress.2007.04.002.

[15] L. Le Gratiet, S. Marelli, B. Sudret, Metamodel-Based Sensitivity Analysis: Polynomial Chaos Expansions and Gaussian Processes, in: R. Ghanem, D. Higdon, H. Owhadi (Eds.), Handbook of Uncertainty Quantification, Springer International Publishing, Cham, 2017, pp. 1289–1325.

[16] R. Ghanem, P. D. Spanos, Polynomial Chaos in Stochastic Finite Elements, Journal of Applied Mechanics 57 (1) (1990) 197–202. doi:10.1115/1.2888303.

[17] D. Xiu, G. E. Karniadakis, The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations, SIAM Journal on Scientific Computing 24 (2) (2002) 619–644. doi:10.1137/S1064827501387826.

[18] M. D. Spiridonakos, E. N. Chatzi, Metamodeling of dynamic nonlinear structural systems through polynomial chaos NARX models, Computers & Structures 157 (2015) 99–113. doi:10.1016/j.compstruc.2015.05.002.
URL https://www.sciencedirect.com/science/article/pii/S004579491500139X

[19] E. Torre, S. Marelli, P. Embrechts, B. Sudret, Data-driven polynomial chaos expansion for machine learning regression, Journal of Computational Physics 388 (2019) 601–623. doi:10.1016/j.jcp.2019.03.039.

[20] S. Marelli, B. Sudret, UQLab user manual – Polynomial chaos expansions, Tech. Rep. # UQLab-V1.3-104, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland (2019).

[21] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, Journal of Computational Physics 230 (6) (2011) 2345–2367. doi:10.1016/j.jcp.2010.12.021.

[22] M. Berveiller, B. Sudret, M. Lemaire, Stochastic finite element: a non intrusive approach by regression, European Journal of Computational Mechanics 15 (1-3) (2006) 81–92. doi:10.3166/remn.15.81-92.

[23] M. Eldred, C. Webster, P. Constantine, Evaluation of Non-Intrusive Approaches for Wiener-Askey Generalized Polynomial Chaos., in: 49th AIAA Structures, Structural Dynamics, and Materials Conference, 2008.

[24] T. J. Santner, B. J. Williams, W. I. Notz, The Design and Analysis of Computer Experiments, Springer Series in Statistics, Springer, New York, 2003.

[25] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning), The MIT Press, 2005.

[26] C. Lataniotis, D. Wicaksono, S. Marelli, B. Sudret, UQLab user manual – Kriging (Gaussian process modeling), Tech. Rep. # UQLab-V1.3-105, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland (2019).

[27] V. Dubourg, Adaptive surrogate models for reliability analysis and reliability-based design optimization, Doctoral thesis, Université Blaise Pascal - Clermont-Ferrand II, Clermont-Ferrand, France (2011).

[28] D. E. Goldberg, J. H. Holland, Genetic Algorithms and Machine Learning, Machine Learning 3 (2) (1988) 95–99.

[29] R. Schobi, B. Sudret, J. Wiart, Polynomial-chaos-based Kriging, International Journal for Uncertainty Quantification 5 (2) (2015) 171–193. doi:10.1615/Int.J.UncertaintyQuantification.2015012467.

[30] R. Schöbi, B. Sudret, S. Marelli, Rare Event Estimation Using Polynomial-Chaos Kriging, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 3 (2) (2017) D4016002. doi:10.1061/AJRUA6.0000870.

[31] P. Kersaudy, B. Sudret, N. Varsier, O. Picon, J. Wiart, A new surrogate modeling technique combining Kriging and polynomial chaos expansions – Application to uncertainty analysis in computational dosimetry, Journal of Computational Physics 286 (2015) 103–117. doi:10.1016/j.jcp.2015.01.034.

[32] R. Schöbi, S. Marelli, B. Sudret, UQLab user manual – Polynomial chaos Kriging, Tech. Rep. # UQLab-V1.3-109, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland (2019).

[33] I. M. Sobol', Sensitivity analysis for non-linear mathematical models, Mathematical Modelling and Computational Experiment 1 (1993) 407–414.

[34] S. Marelli, C. Lamas, K. Konakli, C. Mylonas, P. Wiederkehr, B. Sudret, UQLab user manual – Sensitivity analysis, Tech. Rep. # UQLab-V1.3-106, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland (2019).

[35] S. M. Pinheiro, Motorcycle modeling for eCVT-in-the-loop real-time hybrid testing, Master's thesis, University of Porto, Porto, Portugal (2020).

[36] T. Kimishima, T. Nakamura, T. Suzuki, The Effects on Motorcycle Behavior of the Moment of Inertia of the Crankshaft, SAE Transactions 106 (1997) 1993–2003.

[37] M. Tanelli, Modelling, simulation and control of two-wheeled vehicles, John Wiley & Sons, 2014.

[38] R. Sharp, S. Evangelou, D. Limebeer, Advances in the Modelling of Motorcycle Dynamics, Multibody System Dynamics 12 (3) (2004) 251–283.

[39] M. D. Mckay, R. J. Beckman, W. J. Conover, A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code, Technometrics 42 (1) (2000) 55–61.

[40] S. Marelli, B. Sudret, UQLab: A Framework for Uncertainty Quantification in Matlab, in: Proc. 2nd Int. Conf. on Vulnerability, Risk Analysis and Management (ICVRAM2014), Liverpool, United Kingdom, 2014, pp. 2554–2563. doi:10.1061/9780784413609.257.