A Comparison of Surrogate Modeling Techniques for Global Sensitivity Analysis in Hybrid Simulation

Journal Article

Author(s):
Tsokanas, Nikolaos; Pastorino, Roland; Stojadinovic, Bozidar

Publication date:
2021-12-24

Permanent link:
https://doi.org/10.3929/ethz-b-000522359

Rights / license:
Creative Commons Attribution 4.0 International

Originally published in:

Funding acknowledgement:
764547 - Dynamic virtualisation: modelling performance of engineering structures (EC)
A Comparison of Surrogate Modeling Techniques for Global Sensitivity Analysis in Hybrid Simulation

Nikolaos Tsokanas 1,*, Roland Pastorino 2 and Božidar Stojadinović 1

1 ETH Zurich, IBK, D-BAUG, Stefano-Franscini Platz 5, 8093 Zurich, Switzerland; stojadinovic@ibk.baug.ethz.ch
2 Siemens Industry Software NV, Interleuvenlaan 68, 3001 Leuven, Belgium; roland.pastorino@siemens.com
* Correspondence: tsokanas@ibk.baug.ethz.ch

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 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 is extensively used across various engineering disciplines, it is often the case that the hybrid model and related excitation are conceived as being deterministic. However, associated uncertainties are present, whilst simulation deviation, due to their presence, could be significant. In 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 an unaffordable amount of hybrid simulation evaluations. Therefore, surrogate modeling techniques using machine learning data-driven regression are utilized to alleviate this burden. This study extends the current global sensitivity analysis practices in hybrid simulation by employing various different surrogate modeling methodologies as well as providing comparative results. In particular, polynomial chaos expansion, Kriging and polynomial chaos Kriging are used. A case study encompassing a virtual hybrid model is employed, and hybrid model response quantities of interest are selected. Their respective surrogates are developed, using all three aforementioned techniques. The Sobol’ indices obtained utilizing each examined surrogate are compared with each other, and the results highlight potential deviations when different surrogates are used.

Keywords: 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), 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 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. Advanced control techniques [1–4] and model order
reduction methods [5,6] 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 [7,8].

HS is widely used across different engineering disciplines [9–15]. However, it is often the case that the hybrid model and related excitation are conceived as being deterministic. Nonetheless, this assumption does not reflect 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 is 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 [16,17]. GSA aims to quantitatively determine the degree each input parameter affects the selected QoI of the hybrid model response. Sobol’ indices are 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 [18]. Monte Carlo simulations can be used to estimate Sobol’ indices. Nonetheless, for each index estimation, $O(10^3)$ model evaluations would be required [16]. To alleviate this burden, surrogate models of the QoI response can be developed. Surrogates (i.e., metamodels) are used to substitute an expensive-to-compute model with an inexpensive-to-compute surrogate, and are developed using machine learning data-driven regression techniques. Therefore, Monte Carlo simulations using the developed surrogates can be performed at affordable cost, and Sobol’ indices’ estimation becomes easier to compute. In particular, for the case that a polynomial chaos expansion (PCE) is constructed, then the Sobol’ indices can be computed analytically as a byproduct of the polynomial coefficients at no additional cost [19,20].

As explained in [16], PCE is used to surrogate the hybrid model response and to then compute the Sobol’ indices. This study extends the GSA framework proposed in [16] by utilizing multiple and different surrogate modeling techniques to conduct GSA via Sobol’ indices in HS. In particular, apart from PCE, Kriging (i.e., Gaussian process) and polynomial chaos Kriging (PCK) are used to surrogate the selected hybrid model QoI responses and to perform GSA. In addition, this study serves also as a demonstration of how the obtained Sobol’ indices could possibly deviate from each other when different surrogate modeling techniques are used. To validate the performance of the aforementioned methodologies, a case study encompassing a virtual hybrid model is employed, and specific QoI responses are selected. Surrogates for the selected QoI are constructed, and their performance is evaluated. Sobol’ indices are computed using each developed surrogate, and a comparison with each other is made to highlight potential deviations. The results demonstrate the effectiveness of each examined surrogate modeling method to perform GSA in HS. It should be noted that the utilized methodologies for conducting GSA handle the hybrid model as a black-box, and thus, they can be applied to any dynamic system investigated via HS.

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 $X \in D_X \subset \mathbb{R}^N$ and a computational model $Y = M(X)$ with $Y \in \mathbb{R}^N$, machine learning data-driven regression algorithms formulate a map $M^*: X \mapsto Y$ based on an obtained sample set of input points $X = \{x^{(1)}, \ldots, x^{(N)}\}$ and of the respective output values, i.e., model evaluations or experimental measurements,
$Y = \{ y^{(1)}, \ldots, y^{(N)} \}^T$. The set of $X,Y$ 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 (i.e., metamodel), representing the outputs of the model by a polynomial function of its inputs [21,22]. 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 [16,23,24].

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

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

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

#### 2.1.1. Polynomial Basis

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

$$\Psi_\alpha(X) = \prod_{i=1}^N \phi^{(i)}_{\alpha_i}(x^{(i)}). \quad (2)$$

The latter meets the orthonormality criteria:

$$\left\langle \phi^{(i)}_j(x^{(i)}), \phi^{(i)}_k(x^{(i)}) \right\rangle = \int_{D_{X_i}} \phi^{(i)}_j(x^{(i)}) \phi^{(i)}_k(x^{(i)}) f_{X_i}(x^{(i)}) dx_i = \delta_{jk}, \quad (3)$$

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 $\delta_{jk}$ 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 [22,25].

#### 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 Equation (1). However, because the sum of Equation (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 $A \subset \mathbb{N}^N$, and the PCE of $M(X)$ admits

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

The performance of PCE is closely connected with the truncation scheme used. Underfitting or overfitting is possible when too many terms are discarded or introduced, respectively [24]. In the standard basis truncation scheme [22] 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 $(N+p)^N_\pi = \frac{(N+p)!}{N!\pi}$ elements and follows
where $N$ corresponds to the number of input variables of $\mathcal{M}(X)$ and $|\alpha| = \sum_{i=1}^{N} a_i$ to the total degree of all polynomials in $\Psi$. To further reduce the polynomial basis size, additional truncation schemes were developed, namely, the maximum interaction and hyperbolic truncation scheme.

In the maximum interaction truncation scheme [25], the basis of Equation (5) is reduced such that the $\alpha$ indices to include at most are $r$ non-zero components, and thus the rank of $\mathcal{A}$ is decreased [25]. Accordingly, Equation (5) is written as

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

where $||\alpha||_0 = \sum_{i=1}^{N} I_{\{a_i > 0\}}$ and $I_{\{a_i > 0\}}$ is the indicator function.

The hyperbolic (i.e., $q$-norm) truncation scheme [26] reforms the basis of Equation (5) such that

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

where $||\alpha||_q = \left( \sum_{i=1}^{N} a_i^q \right)^{\frac{1}{q}}$ 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 $y = \{ y_\alpha, \alpha \in \mathcal{A} \}$ of Equation (4). Various methods were developed to do this [25]. One of these techniques is the least-squares minimization [27]. This is an non-intrusive method, meaning that the coefficients are obtained after post-processing the ED points. More specifically, Equation (4) is reformed as

$$\mathcal{M}^{PCE}(X) = \sum_{j=0}^{p-1} y_j \Psi_j(X) + \epsilon_P = y^T \Psi(X) + \epsilon_P,$$

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

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

The solution of Equation (9) is attained by ordinary least-squares (OLS) and follows:

$$\hat{y} = (A^T A)^{-1} A^T Y,$$

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

However, for $N < P - 1$, the $A^T A$ matrix is non-invertible, and thus the OLS problem cannot be solved [24,28]. 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 Equation (9) and the latter is modified as

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

where $||y||_1 = \sum_{\alpha \in \mathcal{A}} |y_\alpha|$ and $\lambda$ is 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 Equation (11), is the least angle regression (LAR) algorithm [26]. The latter is also employed in the case study in Section 4 for the respective PCE surrogate. Apart from the LAR algorithm, in the literature, there exist also other meth-
ods for solving the minimization problem of Equation (11), such as orthogonal matching pursuit (OMP) and greedy coordinate descent (GCD). For a more comprehensive review of such algorithms, the reader is encouraged to consult [25,29] and the references therein.

2.2. Kriging

Kriging (i.e., Gaussian process) is a widely used surrogate modeling method. It considers the output of a model \( Y = M(X) \) as a realization of a Gaussian process, with \( x \in D_X \subset \mathbb{R}^N \) and follows [30,31]

\[
M^\text{Krig}(x) = \beta^T f(x) + \sigma^2 Z(x, \omega),
\]

where \( \beta = (\beta_1, \ldots, \beta_p) \) and \( f(x) = (f_1(x), \ldots, f_p(x)) \) 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 \( \sigma^2 \) to its variance. \( Z(x, \omega) \) is a stationary Gaussian process with zero mean and unit variance, defined by an autocorrelation function \( R(x, x'; \theta) \) and its hyperparameters \( \theta \). The coefficients \( \beta \) are computed from the generalized least-square solution, following

\[
\beta = (F^T R^{-1} F)^{-1} F^T R^{-1} Y,
\]

where \( F_{ij} = f_j(x^{(i)}) \) and \( R_{kl} = R(x^{(i)}, x^{(k)}; \theta) \) are the correlation matrix with \( i, k = 1, \ldots, N \) and \( j = 1, \ldots, P \).

In Kriging, the model \( Y \) and surrogate \( \hat{Y}(x) = M^\text{Krig}(x) \) response is assumed to have a joint Gaussian distribution and therefore the mean value and variance of the prediction \( \hat{Y}(x) \) read as [30,32]

\[
\mu_{\hat{Y}}(x) = f(x)^T \beta + r(x)^T R^{-1} (Y - F\beta),
\]

\[
\sigma^2_{\hat{Y}}(x) = \sigma^2 \left( 1 - r(x)^T R^{-1} r(x) + u(x)^T (F^T R^{-1} F)^{-1} u(x) \right),
\]

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

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 [32,33], namely, the simple Kriging expressed by \( \beta^T f(x) = \sum_{j=1}^p f_j(x) \), the ordinary with \( \beta^T f(x) = \beta_0 \) and the universal which follows \( \beta^T f(x) = \sum_{j=1}^p \beta_j f_j(x) \). 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 \( x \) and \( x' \). Some common autocorrelation function families [32,33] are the linear,

\[
R(x, x'; \theta) = \prod_{i=1}^N \max \left( 0, 1 - \frac{|x^{(i)} - x'^{(i)}|}{\theta_i} \right),
\]

the exponential,

\[
R(x, x'; \theta) = \exp \left( - \sum_{i=1}^N \frac{|x^{(i)} - x'^{(i)}|}{\theta_i} \right),
\]

the Gaussian,

\[
R(x, x'; \theta) = \exp \left( - \sum_{i=1}^N \left( \frac{|x^{(i)} - x'^{(i)}|}{\theta_i} \right)^2 \right)
\]
and the Matérn,
\[
R(x, x'; \theta, v) = \prod_{i=1}^{N} \frac{1}{2^{v-1} \Gamma(v)} \left( 2 \sqrt{v} |x^{(i)} - x'^{(i)}| / \theta_i \right)^v K_v \left( 2 \sqrt{v} |x^{(i)} - x'^{(i)}| / \theta_i \right),
\]
(19)
where \( v \geq 1/2 \) corresponds to the shape parameter, \( \Gamma \) is the Euler gamma function, and \( K_v \) is the modified Bessel function of the second kind.

2.2.3. Hyperparameter Estimation

Following the selection of trend and autocorrelation family, the next step in building the Kriging surrogate consists of estimating the unknown hyperparameters \( \theta \). One of the existing methodologies is the maximum-likelihood estimation. The goal of the latter is to maximize the likelihood of model evaluations \( Y \) by determining the appropriate set of the Kriging parameters \( \beta, \sigma^2 \) and \( \theta \). The likelihood function of \( Y \) admits
\[
L(\beta, \sigma^2, \theta; Y) = \left( \det R \right)^{-1/2} \left( \frac{1}{2 \pi \sigma^2} \right)^{N/2} \exp \left( - \frac{1}{2 \sigma^2} (Y - F \beta)^T R^{-1} (Y - F \beta) \right).
\]
(20)
The hyperparameters \( \theta \) are computed by solving the optimization problem,
\[
\theta = \arg \min_{\theta \in D_\theta} \left( - \log L(\theta; Y) \right)
= \arg \min_{\theta \in D_\theta} \frac{1}{2} \left( \log(\det R) + N \log(2 \pi \sigma^2) + N \right).
\]
(21)
For a more detailed description of the derivation of Equation (21) along with additional methods for hyperparameter estimation, e.g., cross-validation estimation, the reader should refer to [30,33].

2.2.4. Optimization Methods

The solution of Equation (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 [32,34].

The Kriging surrogate used later in the case study of this paper 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 Equations (4) and (12) and follows [35–37]:
\[
\mathcal{M}_{\text{PCK}}(x) = \sum_{a \in A} y_a \Psi_a(X) + \sigma^2 Z(x, \omega).
\]
(22)
The first term, \( \sum_{a \in A} y_a \Psi_a(X) \), 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 Equation (22) corresponds to the variance of the Gaussian process \( \sigma^2 \) and \( Z(x, \omega) \) is a stationary Gaussian process with zero mean and unit variance as of Equation (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 \[35,36,38\].

The PCK surrogate used later in the case study of this paper 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

\[
\varepsilon_{\text{LOO}} = \frac{1}{N} \sum_{i=1}^{N} \left( M(x^{(i)}) - M_{\text{\bar{i}}}(x^{(i)}) \right)^2,
\]  

(23)

where \( M_{\text{\bar{i}}}(x^{(i)}) \) corresponds to the surrogates developed from the subset of the ED, acquired by removing the \( x^{(i)} \) point.

3. Global Sensitivity Analysis with Sobol’ Indices

Broadly speaking, GSA aims to quantitatively identify the level to which each input variable in \( X = \{x^{(1)}, \ldots, x^{(N)}\}^T \), \( X \in D_X \subset \mathbb{R}^N \) affects the response of \( Y = M(X) \), \( Y \in \mathbb{R} \). As mentioned previously, 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 \( X \in D_X \subset \mathbb{R}^N \) with independent components, \( Y = M(X) \) with \( \text{Var}[Y] < +\infty \) can be rewritten as \[16,39\]

\[
M(X) = M_0 + \sum_{i=1}^{N} M_i(x^{(i)}) + \sum_{1 \leq i \leq j \leq N} M_{ij}(x^{(i)}, x^{(j)}) + \cdots + M_{1,2,\ldots,N}(X)
\]  

(24)

where \( M_0 \) is constant and denotes the mean value of \( Y \), \( u \neq \emptyset \), \( u \subseteq \{1, \ldots, N\} \) and \( X_u \) is a sub-vector of \( X \), including the \( u \)-indexed elements. The summands in the latter equation correspond to \( 2^{N-1} \) terms. The elementary functions are defined by conditional expectations:

\[
M_u(X_u) = \sum_{v \subseteq u} (-1)^{|u|-|v|} E[Y | X_v = x_v],
\]  

(25)

where \(|u|\) gives the cardinality of \( u \). When the following holds,

\[
\int_{D_X \setminus \{m\}} M_u(X_u) f_{X_m}(x_m) \, dx_m = 0, \text{ if } m \in u,
\]  

(26)

the Sobol’–Hoeffding decomposition of Equation (24) is unique. Due to this uniqueness, the orthogonality property

\[
E[M_u(X_u) \cdot M_v(X_v)] = 0, \text{ if } u \neq v,
\]  

(27)

also holds. Because of Equations (26) and (27), the total variance of \( M(X) \) decomposes as

\[
D = \text{Var}[M(X)] = \sum_{u \subseteq \{1, \ldots, N\}} D_u,
\]  

(28)
where

\[ D_u = \text{Var} [\mathcal{M}_u(X_u)] = \mathbb{E} [\mathcal{M}_u^2(X_u)], \]  

(29)

corresponds to the partial variance.

3.2. Sobol’ Indices

The Sobol’ index \( S_u \) is determined as the ratio between the partial and total variance that represents the \( u \)-indexed set of input variables, and follows

\[ S_u = \frac{D_u}{D}. \]  

(30)

For \(|u| = 1\), the separate impact of each input variable \( x^{(i)} \) to the response \( Y = \mathcal{M}(X) \) 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 \( S_{Ti} \) describe the overall impact of each input variable, considering its first-order index and all interactions with the other input variables, and admit

\[ S_{Ti} = 1 - S_{-i}, \]  

(31)

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

3.3. Sobol’ Indices Estimation

3.3.1. Monte Carlo-Based Estimation

The variances of Equations (28) and (29) can be obtained by estimators from Monte Carlo simulations. These read as [40]

\[ \hat{\mathcal{M}}_0 = \frac{1}{N} \sum_{i=1}^{N} \mathcal{M}^e(x^{(i)}) \]

\[ \hat{D} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{M}^e(x^{(i)})^2 - \hat{\mathcal{M}}_0^2 \]  

(32)

\[ \hat{D}_u = \frac{1}{N} \sum_{i=1}^{N} \mathcal{M}^e(x_u^{(i)}, x_{-u}^{(i)}) \mathcal{M}^e(x_u^{(i)}, x_{-u}^{(i)}) - \hat{\mathcal{M}}_0^2 \]

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

3.3.2. Sobol’ Indices from Polynomial Chaos Expansion

Recall from Equation (4) that the PCE of \( \mathcal{M}(X) \) admits

\[ \mathcal{M}^{\text{PCE}}(X) = \sum_{\alpha \in \mathcal{A}} y_{\alpha} \psi_{\alpha}(X). \]  

(33)
Due to the orthonormality of the PCE basis, the mean and variance of $M_{PCE}(X)$ can be computed analytically from the $y$ coefficients at no extra cost, and follow [20]

$$\hat{\mu} = \mathbb{E}[M_{PCE}(X)] = \mathbb{E}\left[ \sum_{\alpha \in A} y_\alpha \Psi_\alpha(X) \right] = y_0$$ (34)

$$\hat{\sigma}^2 = \text{Var}[M_{PCE}(X)] = \mathbb{E}\left[ \left( M_{PCE}(X) - y_0 \right)^2 \right] = \sum_{\alpha \in A} y_\alpha^2.$$ (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 Equation (24) can be rewritten as

$$M_{PCE}(X) = y_0 + \sum_{u \subset \{1, \ldots, N\}} \sum_{u \neq \emptyset} M_{PCE}^u(X_u),$$ (36)

where $A_u = \{ \alpha \in A : \alpha_m \neq 0 \text{ if and only if } m \in u \}$ corresponds to the set of multi-indices including only $u$ and $M_{PCE}^u(X_u) = \sum_{\alpha \in A_u} y_\alpha \Psi_\alpha(X_u)$, as in Equation (33). Thus, because the Sobol’–Hoeffding decomposition is unique, there exists an analytical representation of $M_{PCE}^u$ [19]. Accordingly to Equations (34) and (35), the total and partial variances of $M_{PCE}(X)$ admit

$$D = \text{Var}[M_{PCE}(X)] = \sum_{\alpha \in A} y_\alpha^2$$

$$D_u = \text{Var}[M_{PCE}^u(X_u)] = \sum_{\alpha \in A_u} y_\alpha^2.$$ (37)

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

$$S_i^{(1)} = \frac{\sum_{\alpha \in A_i} y_\alpha^2}{\sum_{\alpha \in A} y_\alpha^2}, \quad A_i = \{\alpha \in A : \alpha_i > 0, \alpha_{i\neq j} = 0\}$$

$$S_i^T = \frac{\sum_{\alpha \in A_i^T} y_\alpha^2}{\sum_{\alpha \in A} y_\alpha^2}, \quad A_i^T = \{\alpha \in A : \alpha_i > 0\},$$ (38)

respectively. Therefore, the Sobol’ indices can be obtained as a byproduct of the PCE coefficients at no extra cost [19].

4. Case Study

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

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.
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, and 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 \( \text{thr} \) and the angular velocity of the engine \( \omega_{en} \), whereas its output is the torque of the engine \( \tau_{en} \). 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 \( \tau_{vd} \) and angular velocity \( \omega_{vd} \) 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 \( \omega_{rw} \) and the applied force on the brake pedal \( F_{br\,rw} \), whereas its output is the braking torque of the rear wheel \( \tau_{rw} \). Accordingly, the inputs of the front wheel braking system NS are the angular velocity of the front wheel \( \omega_{fw} \) and the applied force on the brake lever \( F_{br\,fw} \), whereas its output is the braking torque of the front wheel \( \tau_{fw} \). The aforementioned substructures are developed using the multi-physics simulation software Simcenter Amesim. The report of [41] 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 HS.

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 vHS algorithm, the Simcenter real-time platform is utilized. For the time integration scheme of the vHS, the fourth-order Runge–Kutta (RK4) method is used with a fixed time-step of 0.1 milliseconds.
The simulation duration of the case study is 45 s long, on a given driving scenario and wind/road conditions. In particular, the driving scenario is defined by the variations of the throttle $\text{thr}$ 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 vHS, namely $F_{\text{br},\text{rw}} = F_{\text{br},\text{f},\text{w}}$. Equations (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 newtons. Equation (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.

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

\[
F_{\text{br},\text{rw}}(t) = \begin{cases} 
10t & , 20 \leq t < 25 \\
50 & , 25 \leq t \leq 32 \\
-10t & , 32 < t \leq 37 \\
0 & , \text{elsewhere}
\end{cases}
\]

\[
h(x) = \begin{cases} 
0 & , 0 \leq x \leq 2 \\
0.02 \left( \cos \left( \frac{2\pi x}{3} \right) - 1 \right) & , \text{elsewhere}
\end{cases}
\]
quantities from the dynamic response of the motorcycle’s velocity are 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 12 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 are identified from [42–45] 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

\[
X = \{K_{rt}, Z_{rt}, K_{ft}, Z_{ft}, K_{rs}, Z_{rs}, K_{fs}, Z_{fs}, M, J, E\mu, B\mu\}
\]

\[
Y = \{v_{\text{max}}, v_{\text{mean}}\},
\]

From the input parameters of Table 1, 200 points are generated using the Latin hypercube sampling (LHS) method [46], and accordingly, 200 vHS are conducted, collecting the QoI values in the \(Y\) vector of Equation (42). With the obtained data in \(\{X, Y\}\), a PCE, a Kriging and a PCK surrogate are developed for each QoI, as described in Sections 2.1–2.3, respectively. Figure 4 illustrates a sample vHS 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, i.e., nominal, values of Table 1, while the hybrid model is excited with the driving scenario, the road profile and wind conditions that are described by Equations (39)–(41).

<table>
<thead>
<tr>
<th>Param.</th>
<th>Prob. Distrib.</th>
<th>Mean Value</th>
<th>Stand. Dev.</th>
<th>CV (%)</th>
<th>Parameter Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>(K_{rt})</td>
<td>Unif.</td>
<td>58,570</td>
<td>11,714</td>
<td>20</td>
<td>Vertical stiffness rear tire</td>
<td>N/m</td>
</tr>
<tr>
<td>(Z_{rt})</td>
<td>Unif.</td>
<td>11,650</td>
<td>3495</td>
<td>30</td>
<td>Vertical damping rear tire</td>
<td>Ns/m</td>
</tr>
<tr>
<td>(K_{ft})</td>
<td>Unif.</td>
<td>25,000</td>
<td>5000</td>
<td>20</td>
<td>Vertical stiffness front tire</td>
<td>N/m</td>
</tr>
<tr>
<td>(Z_{ft})</td>
<td>Unif.</td>
<td>2134</td>
<td>640.2</td>
<td>30</td>
<td>Vertical damping front tire</td>
<td>Ns/m</td>
</tr>
<tr>
<td>(K_{rs})</td>
<td>Unif.</td>
<td>125,000</td>
<td>25,000</td>
<td>20</td>
<td>Stiffness rear suspension</td>
<td>N/m</td>
</tr>
<tr>
<td>(Z_{rs})</td>
<td>Unif.</td>
<td>10,000</td>
<td>3000</td>
<td>30</td>
<td>Damping rear suspension</td>
<td>Ns/m</td>
</tr>
<tr>
<td>(K_{fs})</td>
<td>Unif.</td>
<td>19,000</td>
<td>3800</td>
<td>20</td>
<td>Stiffness front suspension</td>
<td>N/m</td>
</tr>
<tr>
<td>(Z_{fs})</td>
<td>Unif.</td>
<td>1250</td>
<td>375</td>
<td>30</td>
<td>Damping front suspension</td>
<td>Ns/m</td>
</tr>
<tr>
<td>(M)</td>
<td>Unif.</td>
<td>300</td>
<td>15</td>
<td>5</td>
<td>Motorcycle mass</td>
<td>Kg</td>
</tr>
<tr>
<td>(J)</td>
<td>Unif.</td>
<td>0.0115</td>
<td>0.0023</td>
<td>20</td>
<td>Engine moment of inertia</td>
<td>Kgm²</td>
</tr>
<tr>
<td>(E\mu)</td>
<td>Unif.</td>
<td>0.0012</td>
<td>0.00018</td>
<td>15</td>
<td>Engine coefficient of viscous friction</td>
<td>Nm/rev/min</td>
</tr>
<tr>
<td>(B\mu)</td>
<td>Unif.</td>
<td>0.13</td>
<td>0.0195</td>
<td>15</td>
<td>Brake coefficient of viscous friction</td>
<td>Nm/rad/s</td>
</tr>
</tbody>
</table>
Figure 4. Sample vHS time history response of the motorcycle prototype hybrid model, using the mean (nominal) values of Table 1. The QoI are highlighted in red color.

Figure 5. Cont.
Figure 5. Indicative hybrid model responses, using the mean (nominal) 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 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 $\tau_{\text{fw}}$, generated by the utilized driving scenario. Recall from Figure 3 and Equation (40) that the brakes of the motorcycle are enabled in the time interval between the 20th and 37th second of the simulation and as a result, the braking torques are non-zero only then. According to Figure 3 and Equation (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 5th and 17th second of the simulation, the throttle starts to increase and hence, the motorcycle starts to accelerate. From Figure 4, it can be seen that the velocity is also increasing. After the 20th second of the simulation, the braking starts (Equation (40)) and as a result, the velocity of the motorcycle begins to decrease, while it is brought to almost a full stop in the 45th second of the simulation.

5. Results and Discussion

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 15 to 200 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 150 samples of the ED, and the remaining 50 samples are used to validate them.

In addition, Figures 8 and 9 depict the convergence plots of the PCE-based moments estimates, i.e., mean, standard deviation and coefficient of variation (CV) values, for both QoI. Recall that the PCE-based moments estimates can be computed from the PCE coefficients as a post-process at no additional computational cost [19]. However, the latter is not a feature of Kriging or PCK. Thus, Figures 8 and 9 do not include moment estimates from these two surrogates. Nevertheless, to further evaluate the performance of the developed surrogates, Figure 10 compares the prediction PDF of each surrogate against the histograms of the measurements for each QoI. The surrogate predictions are evaluated based on 200 random samples, generated from the distributions described in Table 1 using the LHS method. From Figure 10, 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 10, 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 times are approximately 6.1 and 7.5 s, respectively. Nevertheless, for the case of PCK, it is approximately 38 s. 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 for which each surrogate is developed.

![Figure 6](image1.png)  
**Figure 6.** Convergence of LOO error estimates of (a) $v_{\text{max}}$ and (b) $v_{\text{mean}}$ surrogates.

![Figure 7](image2.png)  
**Figure 7.** Comparison between QoI measurements and corresponding surrogate predictions for (a) $v_{\text{max}}$ and (b) $v_{\text{mean}}$. 
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 [19], as described in Section 3.3.2. For the cases of PCK and Kriging, though, this is not possible; 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 are performed to compute the respective Sobol’ indices. However, it is worth mentioning that the computational cost of running these evaluations is negligibly small ($\approx 1$ s for each surrogate).

**Figure 8.** Convergence of PCE-based moments estimates of (a) mean, (b) standard deviation and (c) CV of QoI $v_{\text{max}}$. 
Figure 9. Convergence of PCE-based moments estimates of (a) mean, (b) standard deviation and (c) CV of QoI $v_{\text{mean}}$.

In Figure 11 the first-order and total Sobol’ indices for both QoI $v_{\text{max}}$ and $v_{\text{mean}}$ are reported. Clearly, both first-order and total Sobol’ indices computed utilizing the different examined surrogates are admittedly close with each other. This confirms 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 $E\mu$ are the most sensitive variables for both QoI. In particular, for the QoI related to the maximum motorcycle velocity, namely, $v_{\text{max}}$, the motorcycle mass has a greater effect than the engine friction coefficient, whereas, for the QoI related to the mean motorcycle velocity, namely $v_{\text{mean}}$, the engine friction coefficient contributes more than the motorcycle mass. On the contrary, the remaining input variables of Table 1 do not affect significantly 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.
Figure 10. Comparison between histogram measurements and corresponding surrogate estimates for (a) $v_{\text{max}}$ and (b) $v_{\text{mean}}$.

The development and implementation of the surrogate modeling, as well as the GSA, are performed with the UQLab software framework developed by the Chair of Risk, Safety and Uncertainty Quantification in ETH Zurich [47].

Figure 11. First-order and total Sobol’ indices for (a,b) $v_{\text{max}}$ and (c,d) $v_{\text{mean}}$. 
6. Conclusions

This paper shows 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, i.e., its hybrid model, 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 many 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 can be constructed and used to estimate the Sobol’ indices at a 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 (i.e., 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 indicate that the respective validation errors were similar. Additionally, negligibly small deviations are observed in the Sobol’ indices 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.

**Author Contributions:** N.T., conceptualization, methodology, software, validation and writing (review and editing); R.P., supervision and writing (review and editing); B.S., supervision, funding acquisition and writing (review and editing). All authors have read and agreed to the published version of the manuscript.

**Funding:** This project has received funding from the European Union’s Horizon 2020 research and innovation program 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.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

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

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**


9. Abbiati, G.; Lanese, I.; Cazzador, E.; Bursi, O.S.; Pavese, A. A computational framework for fast-time hybrid simulation based on partitioned time integration and state-space modeling. **Struct. Control Health Monit.** 2019, 26, e2419. [CrossRef]

10. Abbiati, G.; Covi, P.; Tondini, N.; Bursi, O.S.; Stojadinović, B. A Real-Time Hybrid Fire Simulation Method Based on Dynamic Relaxation and Partitioned Time Integration. **J. Eng. Mech.** 2020, 146, 04020104. [CrossRef]


15. Tsokanas, N.; Stojadinovic, B. Design of an Actuation Controller for Physical Substructures in Stochastic Real-Time Hybrid Simulations. In Model Validation and Uncertainty Quantification; Mao, Z., Ed.; Society for Experimental Mechanics Series; Springer International Publishing: Cham, Switzerland, 2020; Volume 3; pp. 69–82. [CrossRef]


