Applied Mathematical Modelling 137 (2025) 115746

Contents lists available at ScienceDirect

Applied Mathematical Modelling

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

Multivariate sensitivity-adaptive polynomial chaos expansion for high-dimensional surrogate modeling and uncertainty quantification

Dimitrios Loukrezis [a,b,*], Eric Diehl [a], Herbert De Gersem [b]

[a] Siemens AG, Technology, Munich, Germany
[b] Technische Universität Darmstadt, Institute for Accelerator Science and Electromagnetic Fields (TEMF), Darmstadt, Germany

ARTICLE INFO

Keywords:
Polynomial chaos expansion
Multivariate sensitivity analysis
Machine learning regression
Surrogate modeling
Uncertainty quantification
Curse of dimensionality
Adaptive approximation
Electric machine
Power grid

ABSTRACT

This work develops a novel basis-adaptive method for constructing anisotropic polynomial chaos expansions of multidimensional (vector-valued, multi-output) model responses. The adaptive basis selection is based on multivariate sensitivity analysis metrics that can be estimated by post-processing the polynomial chaos expansion and results in a common anisotropic polynomial basis for the vector-valued response. This allows the application of the method to problems with up to moderately high-dimensional model inputs (in the order of tens) and up to very high-dimensional model responses (in the order of thousands). The method is applied to different engineering test cases for surrogate modeling and uncertainty quantification, including use cases related to electric machine and power grid modeling and simulation, and is found to produce highly accurate results with comparatively low data and computational demand.

1. Introduction

The polynomial chaos expansion (PCE) method is commonly employed in various forms [1] for surrogate modeling and uncertainty quantification (UQ) in scientific and engineering applications. Therein, the PCE offers inexpensive polynomial approximations, commonly referred to as reduced order models or surrogate models, to be used in computationally demanding design tasks such as optimization and reliability analysis [2–4], or for online model-based estimations in real time, e.g., in the context of digital twins [5,6]. Additionally, uncertainty and sensitivity metrics regarding the model response can be estimated by simply post-processing the expansion's terms [7–9] very efficiently, for example compared to Monte Carlo sampling (MCS) methods [10]. Particularly popular are the regression-based variants of the PCE, where the expansion's coefficients are computed by means of least squares regression [11,12]. In that case, the PCE is essentially a supervised machine learning regression model, with the additional benefit of typically requiring few training data and little computation time compared to other data-driven modeling methods [13].

One of the bottlenecks of the PCE method is the so-called "curse of dimensionality", which in this case concerns the rapid growth of the polynomial basis, caused by large input dimensionality and high polynomial degrees. For regression-based PCE methods, this can lead to an ill-conditioned, ill-posed, or simply intractable least squares problem, since the training data demand rises along with the size of the polynomial basis [12,14]. This issue is particularly relevant for numerous engineering applications where data acquisition is complex and/or expensive. Remedies have been sought in sparse and/or basis-adaptive PCE methods (in the following

* Corresponding author at: Technische Universität Darmstadt, Institute for Accelerator Science and Electromagnetic Fields (TEMF), Darmstadt, Germany.
E-mail address: dimitrios.loukrezis@tu-darmstadt.de (D. Loukrezis).

https://doi.org/10.1016/j.apm.2024.115746
Received 16 October 2023; Received in revised form 26 September 2024; Accepted 1 October 2024

Available online 9 October 2024
0307-904X/© 2024 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

referred to as sparse/adaptive). These methods construct the polynomial basis by exploiting possible anisotropies in the way the input parameters affect the model response. That is, depending on the model at hand, changes in certain parameters or parameter combinations can have a significant or negligible impact on the response. Taking into account the differences in parameter impact, an anisotropic polynomial basis can be constructed, which improves the performance of the PCE in terms of accuracy versus data demand. Extensive reviews on sparse/adaptive, regression-based PCE methods can be found in the works of Lüthen et al. [15,16].

On the one hand, sparse PCEs are computed by solving a regularized least squares problem that induces sparsity in its solution and then omitting the zero coefficients. Common methods for computing sparse PCEs are least angle regression (LAR) [17], orthogonal matching pursuit (OMP) [18], and compressive sensing [19–22]. On the other hand, basis-adaptive PCEs are adaptive approximation methods where the polynomial basis is progressively enriched with terms of high impact, while low-impact terms are neglected. Various approaches and criteria have been suggested in the literature for adaptive basis selection, for example, based on residual correlation [17,18], cross-validation error [23], coefficient confidence intervals [24], entropy principles [25], or variance contribution [26,27]. Considering non-trivial applications of practical value, sparse/adaptive PCEs are generally applicable for up to moderately high-dimensional inputs, typically in the order of tens or, less commonly, low hundreds. In the case of very high-dimensional inputs, linear dimension reduction methods such as the Karhunen-Loève expansion are most commonly employed to enable the use of PCEs [28]. More recently, nonlinear dimension reduction methods have also been used for that purpose [29,30].

The dimensionality of the model's response is also a cause for concern regarding the computational cost of regression-based PCEs, as the least squares problem must be solved M times, where M denotes the dimension of the response. Particularly considering high-dimensional responses, e.g., time-series or frequency responses [31,32], the computational cost for computing the PCE may become unfeasible. This is particularly troublesome if sparse/adaptive PCEs are used, since most, if not all, of the available methods have been developed considering scalar model responses and rely on iterative algorithms that cannot be efficiently vectorized. For a multidimensional model response, these methods must be applied element-wise, often resulting in unacceptable computation times. To render sparse/adaptive PCE methods tractable, dimension reduction techniques are used to reduce the dimensionality of the response to its minimum necessary components. Most commonly, linear dimension reduction is used, e.g., based on principal component analysis (PCA) or proper orthogonal decomposition (POD) [33–37]. Nonlinear dimension reduction has also been considered recently [38,39]. Note that UQ based on PCEs combined with dimension reduction necessitates the transformation of the PCE coefficients, such that they refer to the original (non-reduced) output. While this is straightforward for linear dimension reduction methods, it is not trivial for nonlinear methods, in which case UQ is typically performed by sampling the PCE instead of post-processing it. In any case, if dimension reduction does not result in a sufficiently small response dimension, the computational cost may remain unacceptable. Parallel computing can also be used to accelerate computation, assuming however that suitable hardware is available.

This paper suggests a new basis-adaptive PCE method that specifically targets multidimensional model responses. Its main novelty lies in the basis selection criterion employed for the adaptive approximation, which utilizes the metrics of a multivariate global sensitivity analysis (GSA) method [40]. The latter can be seen as a generalization of the well-known, variance-based Sobol sensitivity analysis [41] from scalar to functional and vector-valued model responses. Similar to the Sobol indices, the multivariate sensitivity indices can be estimated by post-processing a vector-valued PCE [42,43] and correspond to contributions to the aggregated variance over the multidimensional model response. Accordingly, the vector-valued PCE coefficients can be interpreted as multivariate sensitivity indicators. Basis enrichment is then performed according to the multivariate sensitivity indicators, such that the terms included in the expansion are the ones with the maximum contribution to the aggregated variance of the vector-valued model response. With respect to the candidate terms for basis enrichment, a forward neighbor approach is used [16], such that the polynomial basis remains downward-closed [14]. We call this basis-adaptive PCE method *multivariate sensitivity-adaptive (MVSA)*.

The MVSA PCE method is designed specifically for vector-valued model responses, contrary to sparse/adaptive PCE methods available in the literature. Using the multivariate GSA-based basis selection criterion, the MVSA algorithm constructs a single anisotropic polynomial basis for the model response. The common polynomial basis allows the use of vectorized ordinary least squares (OLS) solvers, leading to significantly reduced computation times. This can be an important advantage in several use cases, for example, surrogate models that must be computed online and in real time, e.g., in the context of digital twin applications, UQ-informed decision-making in short time windows, or if multiple models must be computed using different data sets, as in bootstrapping or ensemble modeling. The common basis and the vectorized OLS solver also allow the MVSA PCE method to be applied to very high-dimensional model responses, i.e., with sizes up to the order of thousands. On possible downsides, the common basis obtained with the MVSA algorithm can be sub-optimal considering all elements of the model response. However, the trade-off is found to be more than acceptable, as illustrated by the numerical results presented in this work.

The remaining of this paper is structured as follows. Section 2 describes the general problem setting of parameter-dependent models with random inputs. Section 3 recalls variance-based GSA methods for scalar and multidimensional (vector-valued) model responses. Section 4 discusses the PCE method, its computation by means of regression, and its connection to GSA. Section 5 presents the MVSA PCE method for the construction of anisotropic vector-valued PCEs. In Section 6, the MVSA PCE method is applied for surrogate modeling and UQ in different engineering test cases. Our conclusions are discussed in Section 7.

2. Parameter-dependent model with random input data

In the following, we shall consider a parameter-dependent model $f(\mathbf{x})$, such that

$$f : \mathbf{x} \to \mathbf{y}, \text{ equivalently, } \mathbf{y} = f(\mathbf{x}), \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^N$ is an input parameter vector and $\mathbf{y} \in \mathbb{R}^M$ the corresponding model response. The model, which is here abstractly given as the function f, is considered to be deterministic, meaning that its response cannot vary given the same input. As a minimal requirement, we will assume that f is continuous and relatively smooth, such that the PCE method developed in this work can be applied as described in section 5. For non-smooth functions, a multi-element method would be necessary [44].

Throughout this work, the model's input parameters $\mathbf{x}$ are considered to be realizations of a random vector $\mathbf{X} = (X_1, X_2, \dots, X_N)^\top$, where X_n, $n = 1, 2, \dots, N$, are assumed to be independent random variables. The multivariate random variable $\mathbf{X}$ is defined on the probability space (Θ, Σ, P), where Θ is the sample space, Σ the sigma algebra of events, and $P : \Sigma \to [0, 1]$ a probability measure. Accordingly, $\mathbf{x} = \mathbf{X}(\theta) \in \Xi$ is a random variable realization for $\theta \in \Theta$, where $\Xi \subset \mathbb{R}^N$ is called the image space. The random vector is additionally characterized by the probability density function (PDF) $\varrho_{\mathbf{X}}(\mathbf{x}) : \Xi \to \mathbb{R}_{\geq 0}$. Under the independence assumption for the random variables X_n, it holds that $\varrho_{\mathbf{X}}(\mathbf{x}) = \prod_{n=1}^{N} \varrho_{X_n}(x_n)$ and $\Xi = \Xi_1 \times \cdots \times \Xi_N$, where $\varrho_{X_n}(x_n)$ and Ξ_n are the marginal (univariate) PDFs and image spaces, respectively. Due to the propagation of uncertainty through the model, its response is now a dependent random vector $\mathbf{Y} = f(\mathbf{X})$.

3. Global sensitivity analysis

GSA is broadly defined as the study of how uncertainty in the response of a model can be allocated to the considered input uncertainty sources [45]. We are interested in variance-based GSA, also referred to as the Sobol method [41,46]. The corresponding sensitivity metrics are known as Sobol indices. Note that the Sobol method concerns scalar model responses only. Extensions were developed later to address multidimensional model responses [40,47].

3.1. Global sensitivity analysis for scalar model response

We consider a parametric model similar to the one described in Section 2, however, with a scalar response $y = f(\mathbf{x}) \in \mathbb{R}$. The Sobol method begins with a decomposition of the scalar random response $Y = f(\mathbf{X})$, such that

$$Y = f(\mathbf{X}) = f_0 + \sum_{n=1}^{N} f_n\left(X_n\right) + \sum_{n<k}^{N} f_{nk}\left(X_n, X_k\right) + \cdots + f_{1,\dots,N}\left(X_1, \dots, X_N\right), \tag{2}$$

where f_0 is a constant function, f_n is a function of X_n only, f_{nk} is a function of X_n and X_k, and so forth. The terms of (2) are proven to be mutually orthogonal [46], therefore, the expected value of the response is $\mathbb{E}[Y] = f_0$. Then, the variance of the response, here denoted as $\mathbb{V}[Y]$, can be decomposed as

$$\mathbb{V}[Y] = \sum_{n=1}^{N} \mathbb{V}\left[f_n\right] + \sum_{n<k}^{N} \mathbb{V}\left[f_{nk}\right] + \cdots + \mathbb{V}\left[f_{1,\dots,N}\right] = \sum_{n=1}^{N} \mathbb{V}_n + \sum_{n<k}^{N} \mathbb{V}_{nk} + \cdots + \mathbb{V}_{1,\dots,N}, \tag{3}$$

where $\mathbb{V}_n$ is a partial variance attributed only to X_n, $\mathbb{V}_{nk}$ is a partial variance attributed to the combination of X_n and X_k, and so forth. First-order Sobol indices quantify the impact of the random variable X_n alone, i.e., with all other parameters regarded as constant, and are given as

$$S_n^{\mathrm{F}} = \frac{\mathbb{V}_n}{\mathbb{V}[Y]}. \tag{4}$$

Total-effect Sobol indices quantify the impact of the random variable X_n when interacting with all remaining random variables X_k, $k \neq n$, and are given as

$$S_n^{\mathrm{T}} = \frac{1}{\mathbb{V}[Y]}\left(\mathbb{V}_n + \sum_{n<k}^{N} \mathbb{V}_{nk} + \cdots + \mathbb{V}_{1,\dots,N}\right). \tag{5}$$

Sobol indices quantifying other interactions among the random variables, e.g., second- or third-order, can be computed in a similar manner [45].

3.2. Global sensitivity analysis for vector-valued model response

We now consider a multidimensional random response $\mathbf{Y} = \left(Y_1, \dots, Y_M\right)$ and the dependence $\mathbf{Y} = f(\mathbf{X})$, as described in Section 2. In principle, the GSA method presented in Section 3.1 can be applied element-wise to each scalar component Y_m, $m = 1, \dots, M$. However, correlations among the response's components render this approach questionable, as it may lead to redundant and difficult to interpret results [48,49]. To circumvent this impasse, two multivariate GSA methodologies have been developed, based on decomposing either the vector-valued response itself [47] or the corresponding covariance matrix [40]. Both approaches compute generalized sensitivity indices that quantify the impact of the input random variables on the full multivariate response. In fact, the generalized sensitivity indices provided by these two multivariate GSA approaches are equivalent [42].

Following the work of Gamboa et al. [40], we consider a non-empty index set $\mathbf{u} \subset (1, \dots, N)$, its complement $\mathbf{v} = (1, \dots, N) \setminus \mathbf{u}$, and the corresponding input random variable subsets $\mathbf{X}_\mathbf{u} = \left\{X_n\right\}_{n \in \mathbf{u}}$, $\mathbf{X}_\mathbf{v} = \left\{X_n\right\}_{n \in \mathbf{v}}$. By applying the Hoeffding decomposition [50], we obtain the output decomposition

$$\mathbf{Y} = f(\mathbf{X}) = f_0 + f_{\mathbf{u}}(\mathbf{X}_{\mathbf{u}}) + f_{\mathbf{v}}(\mathbf{X}_{\mathbf{v}}) + f_{\mathbf{u},\mathbf{v}}(\mathbf{X}_{\mathbf{u}}, \mathbf{X}_{\mathbf{v}}), \tag{6}$$

where $f_0 \in \mathbb{R}^M$, $f_{\mathbf{u}} : \mathbb{R}^{\#\mathbf{u}} \to \mathbb{R}^M$, $f_{\mathbf{v}} : \mathbb{R}^{\#\mathbf{v}} \to \mathbb{R}^M$, and $f_{\mathbf{u},\mathbf{v}} : \mathbb{R}^N \to \mathbb{R}^M$, where # denotes the cardinality of a set. Taking the covariance matrices in both sides of (6) yields

$$\mathbf{C} = \mathbf{C}_{\mathbf{u}} + \mathbf{C}_{\mathbf{v}} + \mathbf{C}_{\mathbf{u},\mathbf{v}}, \tag{7}$$

where $\mathbf{C}$, $\mathbf{C}_{\mathbf{u}}$, $\mathbf{C}_{\mathbf{v}}$, and $\mathbf{C}_{\mathbf{u},\mathbf{v}}$ are the covariance matrices for f (equivalently, for $\mathbf{Y}$), $f_{\mathbf{u}}$, $f_{\mathbf{v}}$, and $f_{\mathbf{u},\mathbf{v}}$, respectively. By considering index sets comprising single indices, index pairs, triplets, and so forth, the covariance decomposition (7) can be written as

$$\mathbf{C} = \sum_{n=1}^{N} \mathbf{C}_n + \sum_{n<k} \mathbf{C}_{nk} + \cdots + \mathbf{C}_{1,\ldots,N}, \tag{8}$$

which is the multidimensional analog to the variance decomposition (3). In fact, for a scalar response, the variance decomposition (3) is recovered from the covariance matrix decomposition (8). Based on this observation, generalized sensitivity indices for multidimensional responses are motivated as follows. The sum of the variances of all response components Y_m, $m = 1, \ldots, M$, is equal to the trace of the covariance matrix $\mathbf{C}$, here denoted as $\mathrm{tr}(\mathbf{C})$, and corresponds to the aggregated variance of the multidimensional response. Then, the multivariate equivalent to the first-order Sobol index (4) is obtained as

$$G_n^{\mathrm{F}} = \frac{\mathrm{tr}\left(\mathbf{C}_n\right)}{\mathrm{tr}\left(\mathbf{C}\right)}, \tag{9}$$

where the numerator quantifies the aggregated variance caused by the random variable X_n alone. In analogous fashion, the multivariate counterpart to the total-effect Sobol index (5) is given by

$$G_n^{\mathrm{T}} = \frac{\mathrm{tr}\left(\mathbf{C}_n\right) + \sum_{n<k} \mathrm{tr}\left(\mathbf{C}_{nk}\right) + \cdots + \mathrm{tr}\left(\mathbf{C}_{1,\ldots,N}\right)}{\mathrm{tr}\left(\mathbf{C}\right)}. \tag{10}$$

4. Polynomial chaos expansion

Originating from the work of Norbert Wiener on homogeneous chaos, the PCE was popularized much later as an efficient UQ method [28,51]. Recall from Section 2 the (dependent) random model response $\mathbf{Y} = f(\mathbf{X})$, with $\mathbf{Y} = \left(Y_1, \ldots, Y_M\right)$ and $\mathbf{X} = \left(X_1, \ldots, X_N\right)$, and the fixed response $\mathbf{y} = f(\mathbf{x})$ for $\mathbf{x} = \mathbf{X}(\theta)$. Then, a PCE is a global polynomial approximation of the form

$$f(\mathbf{x}) \approx \widetilde{f}(\mathbf{x}) = \sum_{k=1}^{K} \mathbf{c}_k \Psi_k(\mathbf{x}), \tag{11}$$

where $\mathbf{c}_k \in \mathbb{R}^M$ are the expansion coefficients and Ψ_k are multivariate polynomials that satisfy the orthogonality property

$$\mathbb{E}\left[\Psi_k \Psi_l\right] = \int_{\Xi} \Psi_k(\mathbf{x}) \Psi_l(\mathbf{x}) \varrho_{\mathbf{X}}(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \mathbb{E}\left[\Psi_k^2\right] \delta_{kl}, \tag{12}$$

where δ_{kl} is the Kronecker delta. Depending on the distribution of the random inputs, the polynomials are either selected according to the Wiener-Askey scheme [51] or constructed numerically [52–54]. In the remaining of this paper, we always consider orthonormal polynomials, such that $\mathbb{E}\left[\Psi_k^2\right] = 1$. As mentioned in Section 2, the input random variables are assumed to be independent, hence, the joint PDF is given as $\varrho_{\mathbf{X}}(\mathbf{x}) = \prod_{n=1}^{N} \varrho_{X_n}\left(x_n\right)$. Note that it is possible to construct PCEs for dependent input random variables also [55–57]. Denoting with $\psi_n^{k_n}$ univariate polynomials of degree k_n which are orthogonal, respectively, orthonormal with respect to the corresponding marginal PDFs ϱ_{X_n}, and using the multi-index notation $\mathbf{k} = \left(k_1, \ldots, k_N\right)$, multivariate orthogonal polynomials can be constructed as

$$\Psi_{\mathbf{k}}(\mathbf{x}) = \prod_{n=1}^{N} \psi_n^{k_n}(x_n). \tag{13}$$

The PCE (11) can now be equivalently written as

$$f(\mathbf{x}) \approx \widetilde{f}(\mathbf{x}) = \sum_{\mathbf{k} \in \Lambda} \mathbf{c}_{\mathbf{k}} \Psi_{\mathbf{k}}(\mathbf{x}), \tag{14}$$

where the multi-indices $\mathbf{k}$ in (14) are uniquely associated to the single indices k in (11) and Λ is a multi-index set with cardinality $\#\Lambda = K$.

It is typical to use a multi-index set Λ corresponding to a total-degree (TD) basis, where $\Lambda = \left\{\mathbf{k} : \|\mathbf{k}\|_1 \leq p\right\}$ for maximum univariate polynomial degree $p \in \mathbb{Z}_{\geq 0}$. Another common option is the hyperbolic truncation basis, where $\Lambda = \left\{\mathbf{k} : \|\mathbf{k}\|_q \leq p\right\}$, with $q \in (0, 1)$ and $\|\mathbf{k}\|_q = \left(\sum_{n=1}^{N} \left(k_n\right)^q\right)^{1/q}$. Note that both bases are isotropic, meaning that the same univariate basis terms are considered for all inputs X_n, $n = 1, \ldots, N$. As a result, the basis grows rapidly for an increasing maximum polynomial degree p and, more crucially, for a high-dimensional input parameter vector $\mathbf{X}$, which is a manifestation of the so-called curse of dimensionality.

4.1. Regression-based computation of polynomial chaos expansion coefficients

In this work, the coefficients of the PCE are computed by means of least-squares regression [11,12]. Alternative options include pseudo-spectral projection [58–60] and, less commonly, interpolation [44,61]. Collecting the PCE coefficients $\mathbf{c}_k \in \mathbb{R}^M$, $k = 1, \dots, K$, into a matrix $\mathbf{\Gamma} \in \mathbb{R}^{K \times M}$, such that the coefficient $\mathbf{c}_k$ is the k-th row of $\mathbf{\Gamma}$, the coefficients are computed by solving the least squares regression problem

$$\underset{\mathbf{\Gamma} \in \mathbb{R}^{K \times M}}{\arg\min} \left\{ \frac{1}{Q} \sum_{q=1}^{Q} \left(\mathbf{y}^{(q)} - \sum_{k=1}^{K} \mathbf{c}_k \Psi_k \left(\mathbf{x}^{(q)} \right) \right)^2 \right\}, \tag{15}$$

where $\left\{ \mathbf{x}^{(q)}, \mathbf{y}^{(q)} = f\left(\mathbf{x}^{(q)}\right) \right\}_{q=1}^{Q}$ is a set of input parameter realizations along with the corresponding model responses, called the experimental design or the training data set. In matrix format, the regression problem (15) is equivalently written as

$$\underset{\mathbf{\Gamma} \in \mathbb{R}^{K \times M}}{\arg\min} \left\{ \frac{1}{Q} \| \mathbf{B} - \mathbf{D}\mathbf{\Gamma} \|_2 \right\}, \tag{16}$$

where $\mathbf{D} \in \mathbb{R}^{Q \times K}$ with $d_{qk} = \Psi_k \left(\mathbf{x}^{(q)} \right)$ is the least squares system matrix, also called the design matrix, and $\mathbf{B} \in \mathbb{R}^{Q \times M}$ with $b_{qm} = y_m^{(q)}$ the right-hand side.

For the regression problem to be well posed and uniquely solvable with an OLS solver, the size of the experimental design must be at least equal to the size of the polynomial basis, i.e., $Q \geq K$. For a well conditioned OLS problem, Q is typically chosen to be 2–5 times larger than K, depending on the problem at hand [12,14]. This can lead to severe computational problems if the polynomial basis is too large, e.g., for a TD basis with high maximum polynomial degree p and input dimensionality. This limitation can be overcome by methods that penalize (15)–(16) to compute sparse solutions [15], for example using the least absolute shrinkage and selection operator (LASSO) method [62] or compressive sensing [63]. Sparse solutions are typically computed using iterative algorithms such as LAR [17], OMP [18], or subspace pursuit (SP) [20]. Alternatively, basis-adaptive algorithms can be used [16], where the basis is sequentially enriched with new terms, up to a size that does not lead to conditioning issues. A limitation here is that the sparse/adaptive PCE algorithms available in the literature consider scalar model responses almost exclusively, therefore, must be applied element-wise for each of the M components of a vector-valued model response. This can lead to an undesirable computational cost with respect to the computation of the PCE coefficients if the response is high-dimensional.

4.2. Uncertainty quantification and sensitivity analysis based on polynomial chaos expansion

For a PCE approximation in the form of (14), it is straightforward to show that the mean and variance of the response can be estimated as

$$\mathbb{E}[\mathbf{Y}] \approx \mathbf{c}_{\mathbf{0}}, \tag{17a}$$

$$\mathbb{V}[\mathbf{Y}] \approx \sum_{\mathbf{k} \in \Lambda \setminus \mathbf{0}} \mathbf{c}_{\mathbf{k}}^2, \tag{17b}$$

where the zeroth multi-index is denoted as $\mathbf{0} = (0, \dots, 0)$ and the PCE basis is considered to be orthonormal [28].

For a scalar response $Y = f(\mathbf{X})$, the PCE takes the form of the Sobol decomposition (2) by appropriately ordering its terms. Hence, Sobol sensitivity indices can be estimated by simply post-processing the PCE [7,9]. Indeed, the PCE coefficients $c_{\mathbf{k}}$ for $\mathbf{k} \in \Lambda \setminus \mathbf{0}$ can be interpreted as partial variances due to specific random variable interactions defined by the multi-indices $\mathbf{k}$. The multi-indices corresponding to partial variances caused by X_n, either individually (first-order) or in combination with all other random variables (total-effect), can then be collected into the multi-index sets $\Lambda_n^{\mathrm{F}} \subset \Lambda$ and $\Lambda_n^{\mathrm{T}} \subset \Lambda$, respectively defined as

$$\Lambda_n^{\mathrm{F}} = \{ \mathbf{k} \in \Lambda \ : \ k_n \neq 0 \text{ and } k_l = 0, l = 1, \dots, N, l \neq n \}, \tag{18a}$$

$$\Lambda_n^{\mathrm{T}} = \{ \mathbf{k} \in \Lambda \ : \ k_n \neq 0 \}. \tag{18b}$$

The first-order and total-effect Sobol indices are then estimated as

$$S_n^{\mathrm{F}} \approx \frac{\sum_{\mathbf{k} \in \Lambda_n^{\mathrm{F}}} c_{\mathbf{k}}^2}{\sum_{\mathbf{k} \in \Lambda \setminus \mathbf{0}} c_{\mathbf{k}}^2}, \tag{19a}$$

$$S_n^{\mathrm{T}} \approx \frac{\sum_{\mathbf{k} \in \Lambda_n^{\mathrm{T}}} c_{\mathbf{k}}^2}{\sum_{\mathbf{k} \in \Lambda \setminus \mathbf{0}} c_{\mathbf{k}}^2}. \tag{19b}$$

For a multidimensional response $\mathbf{Y} = f(\mathbf{X})$, multivariate GSA requires the computation of the traces of the covariance matrices with respect to the components Y_m, which depend on specific combinations of the input random variables, as shown in Section 3.2. The traces are equal to the sum of the variances of all response components dependent on the specific input random variable combinations. Similar to the case of a scalar response, the partial variances are easily computed from the PCE coefficients [42]. Then, the generalized sensitivity indices G_n^{F} and G_n^{T} defined in (9) and (10), respectively, can be estimated as

Algorithm 1: Multivariate sensitivity-adaptive polynomial chaos expansion.

Data: Training data set $\left\{\mathbf{x}^{(q)}, \mathbf{y}^{(q)} = f\left(\mathbf{x}^{(q)}\right)\right\}_{q=1}^{Q}$ with $\mathbf{x}^{(q)} \in \mathbb{R}^N$ and $\mathbf{y}^{(q)} \in \mathbb{R}^M$, initial downward-closed multi-index set Λ_{init} with $\#\Lambda_{\text{init}} < Q$, maximum condition number κ.
Result: Multi-index set Λ and corresponding PCE.

```
Set Λ = Λ_init.
while True do                                                    // adaptive basis expansion
    Compute the admissible multi-index set Λ^+_adm as in (23).
    Construct the extended multi-index set Λ_ext = Λ ∪ Λ^+_adm.
    if #Λ_ext > Q then                                           // underdetermined OLS problem
        Exit while-loop.
    end
    Solve the OLS problem (15), equivalently (16), for the extended multi-index set Λ_ext.
    if cond(D_ext) > κ then                                      // ill-conditioned OLS problem
        Exit while-loop.
    end
    Find the multi-index k* = arg max_{k∈Λ^+_adm} η_k, where η_k = Σ_{m=1}^M c^2_{m,k} are the sensitivity indicators defined in (24).
    Update the multi-index set Λ, such that Λ ← Λ ∪ {k*}.
end
Set Λ = Λ_ext.
while cond(D) > κ or #Λ > Q do                                   // basis pruning
    Solve the OLS problem (15), equivalently (16), for the multi-index set Λ.
    Find the multi-index k† = arg min_{k∈Λ} η_k.
    Update the multi-index set Λ, such that Λ ← Λ \ {k†}.
end
Solve the OLS problem (15), equivalently (16), for the final multi-index set Λ.
```

$$G_n^{\mathrm{F}} \approx \frac{\sum_{m=1}^{M}\left(\sum_{\mathbf{k}\in\Lambda_n^{\mathrm{F}}} c_{m,\mathbf{k}}^2\right)}{\sum_{m=1}^{M}\left(\sum_{\mathbf{k}\in\Lambda\setminus\mathbf{0}} c_{m,\mathbf{k}}^2\right)}, \tag{20a}$$

$$G_n^{\mathrm{T}} \approx \frac{\sum_{m=1}^{M}\left(\sum_{\mathbf{k}\in\Lambda_n^{\mathrm{T}}} c_{m,\mathbf{k}}^2\right)}{\sum_{m=1}^{M}\left(\sum_{\mathbf{k}\in\Lambda\setminus\mathbf{0}} c_{m,\mathbf{k}}^2\right)}. \tag{20b}$$

Accordingly, a vector-valued PCE coefficient $\mathbf{c}_{\mathbf{k}} = \left(c_{m,\mathbf{k}}\right)_{1\le m\le M}$ yields a contribution to the aggregated variance over the multidimensional model response, that is specific to the multi-index $\mathbf{k}$.

Note that of a common multi-index set Λ is assumed in formulas (17b)-(20), implying a common PCE basis for all response elements. This is not necessarily true, for example, if a sparse/adaptive PCE method is applied element-wise over the response, or for isotropic PCEs with different maximum polynomial degrees per response element. In that case, $\Lambda = \bigcup_{m=1}^{M}\Lambda_m$ is to be used, where if $\mathbf{k}\in\Lambda$ and $\mathbf{k}\notin\Lambda_m$, then $c_{m,\mathbf{k}} = 0$.

5. Multivariate sensitivity-adaptive polynomial chaos expansion

In the MVSA PCE method, the polynomial basis is expanded sequentially, such that the newly added expansion terms are the ones corresponding to the maximum contribution to the aggregated variance of the multidimensional model response (see Sections 3.2 and 4.2). In that way, a common multi-index set Λ and polynomial basis are constructed for the vector-valued response. The candidate terms are selected based on a forward neighbor approach [16,18], such that the multi-index set remains downward-closed [14]. The PCE coefficients are computed with an OLS solver. The sequential basis expansion is terminated if the OLS problem becomes ill-posed or ill-conditioned. The main steps and building blocks of the MVSA PCE method are explained in detail in the following. The complete procedure is also described in Algorithm 1.

5.1. Downward closed multi-index set

In the following, we will require the multi-index set Λ defining the basis of the PCE to be downward-closed, i.e., to satisfy the property

$$\forall\mathbf{k}\in\Lambda \Rightarrow \mathbf{k}-\mathbf{e}_n\in\Lambda, n=1,\dots,N, k_n\neq 0, \tag{21}$$

where $\mathbf{e}_n = \left(\delta_{nv}\right)_{1\le v\le N}$ is the nth unit vector and δ_{nv} denotes the Kronecker delta. Downward closed multi-index sets are also called monotone or lower sets. The corresponding downward-closed polynomial space $\mathbb{P}_\Lambda = \operatorname{span}\left\{\Psi_{\mathbf{k}}(\mathbf{x}), \mathbf{k}\in\Lambda\right\}$ satisfies desirable properties such as differentiation in any variable and invariance by a change of basis [14].

5.2. Forward neighbors and admissible multi-indices

Given a multi-index set Λ, the set of forward neighbors is defined as

$$\Lambda^+ = \left\{ \mathbf{k} + \mathbf{e}_n \notin \Lambda \mid n = 1, \dots, N, \mathbf{k} \in \Lambda \right\}. \tag{22}$$

Assuming that the multi-index set Λ is downward-closed, it can easily be observed that $\Lambda \cup \Lambda^+$ is not necessarily downward-closed, i.e., there may exist multi-indices $\mathbf{k} \in \Lambda^+$ for which $\Lambda \cup \{\mathbf{k}\}$ is not a downward-closed set. To retain the downward-closedness property, the set of admissible forward neighbors is defined as

$$\Lambda^+_{\text{adm}} = \left\{ \mathbf{k} \in \Lambda^+ \mid \mathbf{k} - \mathbf{e}_n \in \Lambda, n = 1, \dots, N, k_n \neq 0 \right\}. \tag{23}$$

That is, for any admissible multi-index $\mathbf{k} \in \Lambda^+_{\text{adm}}$, the set $\Lambda \cup \{\mathbf{k}\}$ is downward-closed. Equivalently, $\Lambda \cup \Lambda^+_{\text{adm}}$ is downward-closed.

5.3. Adaptive basis expansion

Assuming that the polynomial basis of a currently available PCE is based on a downward-closed multi-index set Λ, we require that the polynomial basis will be expanded by PCE terms corresponding to admissible multi-indices $\mathbf{k} \in \Lambda^+_{\text{adm}}$. If a downward-closed PCE basis is not readily available, the adaptive basis expansion starts with the zero multi-index, i.e., $\Lambda = \{\mathbf{0}\}$.

In each basis expansion step, the set of admissible neighbors Λ^+_{adm} is computed and an extended multi-index set $\Lambda_{\text{ext}} = \Lambda \cup \Lambda^+_{\text{adm}}$ is formed. Using the polynomial basis corresponding to Λ_{ext} and the available training data $\left\{ \mathbf{x}^{(q)}, \mathbf{y}^{(q)} = f\left(\mathbf{x}^{(q)}\right) \right\}_{q=1}^{Q}$, the OLS problem (15), equivalently (16), is solved to compute the vector-valued coefficients $\mathbf{c}_{\mathbf{k}}$, $\mathbf{k} \in \Lambda_{\text{ext}}$. As previously shown in Section 4.2, the value of the sum

$$\eta_{\mathbf{k}} = \sum_{m=1}^{M} c^2_{m,\mathbf{k}}, \tag{24}$$

quantifies the contribution of the $\mathbf{k}$-indexed polynomial term to the aggregated variance of the M-dimensional model response, see equation (17b). Looking at the PCE-based estimation of the multivariate sensitivity indices given in equations (20), $\eta_{\mathbf{k}}$ can be interpreted as an indicator of the sensitivity of the multidimensional response to the $\mathbf{k}$-indexed polynomial term. Therefore, in every step of the basis expansion process, the multi-index set Λ is expanded as

$$\Lambda \leftarrow \Lambda \cup \left\{ \mathbf{k}^* \right\}, \text{ where } \mathbf{k}^* = \underset{\mathbf{k} \in \Lambda^+_{\text{adm}}}{\arg\max}\, \eta_{\mathbf{k}}, \tag{25}$$

such that the currently available downward-closed multi-index set is expanded with the admissible multi-index that corresponds to the maximum variance contribution, equivalently, to the maximum multivariate sensitivity indicator. For that reason, we call the basis expansion procedure *multivariate sensitivity-adaptive (MVSA)*.

5.4. Termination criteria and basis pruning

The adaptive basis expansion procedure described above continues as long as two conditions are respected. The first is that the OLS system cannot become underdetermined when computing the PCE coefficients. The second is related to the conditioning of the OLS problem. For that reason, a limit value $\kappa \in \mathbb{R}$ is set for the condition number of the OLS system matrix, also called the design matrix (see Section 4.1). The latter is essentially a hyperparameter and its optimal value differs for each considered problem. Additionally, the value of κ affects the accuracy of the obtained PCE and its computation time. In the numerical experiments presented in Section 6, the value $\kappa = 100$ is chosen, following the results of our own prior work [27] regarding the sensitivity of surrogate modeling performance with to this hyperparameter. This proved to be a good practical choice for all test cases examined in the present work.

Denoting with $\mathbf{D}_{\text{ext}}$ the design matrix corresponding to the extended multi-index set Λ_{ext}, the adaptive PCE basis expansion proceeds as follows. As long as the conditions $\#\Lambda_{\text{ext}} \leq Q$ and $\text{cond}\left(\mathbf{D}_{\text{ext}}\right) \leq \kappa$ are satisfied, the polynomial basis is expanded as described in Section 5.3. If either condition is violated, the adaptive basis expansion stops and a PCE is computed using the multi-index set $\Lambda = \Lambda_{\text{ext}}$. Of course, this choice violates at least one of the considered OLS stability criteria, most commonly the one related to the condition number of the design matrix. To correct this, a basis pruning procedure is followed, where polynomial terms of comparatively reduced contribution to the PCE are removed from Λ until the OLS stability criteria are satisfied. In each step of the basis pruning procedure, the multi-index that corresponds to the minimum sensitivity indicator $\eta_{\mathbf{k}}$ is identified and removed from the multi-index set, such that

$$\Lambda \leftarrow \Lambda \setminus \left\{ \mathbf{k}^\dagger \right\}, \text{ where } \mathbf{k}^\dagger = \underset{\mathbf{k} \in \Lambda}{\arg\min}\, \eta_{\mathbf{k}}. \tag{26}$$

The corresponding polynomial term is then removed from the PCE basis. The removal of multi-indices and polynomial terms continues until the conditions $\#\Lambda \leq Q$ and $\text{cond}\left(\mathbf{D}\right) \leq \kappa$ are satisfied, where $\mathbf{D}$ is the design matrix corresponding to the multi-index set Λ.

Note that the final multi-index set resulting from the basis pruning procedure is not necessarily downward-closed. The downward-closedness constraint is crucial during the adaptive basis expansion, as it prevents the admissible multi-index set from becoming

intractable due to the curse of dimensionality. Once the adaptive basis expansion has been terminated, the goal is to retain the most significant basis terms, which are here selected according to the (multivariate) sensitivity of the vector-valued response. The removal of insignificant multi-indices can result in a final multi-index set that is not downward-closed.

6. Numerical experiments

In the following, the MVSA PCE method is applied for surrogate modeling and UQ to different engineering problems, each featuring a high-dimensional model response and moderately high-dimensional random inputs. In all considered test cases, the MVSA PCE is computed for training data sets of increasing size and then compared against other methods with respect to its approximation accuracy, uncertainty estimation accuracy, and computational efficiency. The latter concerns both accuracy versus training data demand and computation time. We note the following remarks with respect to the presented numerical results:

Surrogate model accuracy The approximation accuracy of a surrogate model is evaluated using the root-mean-square error (RMSE) metric, which is computed using a test data set. Naturally, the test data are not included in the training data set.

Uncertainty estimates The estimated uncertainty metrics, in particular, expected values and standard deviations of the vector-valued responses, are compared against reference values obtained with MCS. The performance of MCS is not affected from the input or output dimensionality of the model, hence this choice. However, MCS converges slowly and therefore needs a large number of model evaluations for accurate uncertainty estimates.

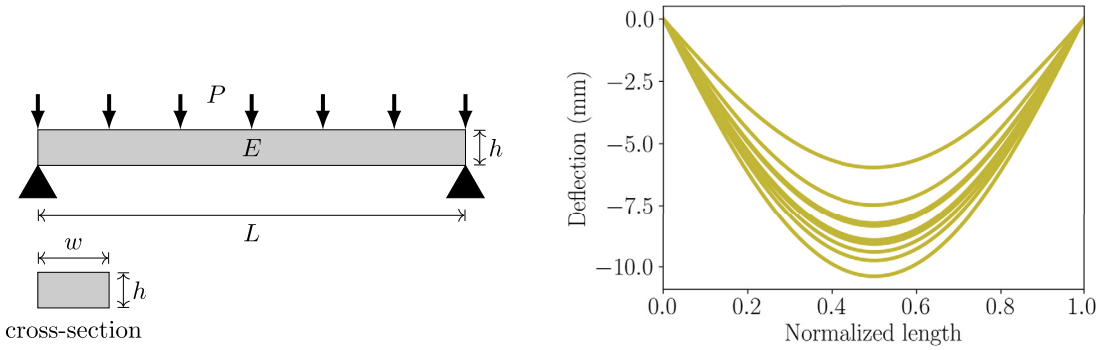
Computation time Where applicable, we report the time needed to compute the PCEs as a further efficiency metric next to accuracy versus training data demand. Despite commonly regarded as negligible next to the cost of data acquisition, a faster computation time - in our case, often several orders of magnitude faster - can be an important advantage in various use cases. One such use case concerns online surrogate modeling, which is a common element of digital twin applications. Therein, data-driven regression models must often be computed using sensor data obtained during the operation of the physical asset. These models must be made available as quickly as possible, so that they can be used in real time during operation, especially since changes in the operating conditions might render a data-driven model invalid, in which case a new model must be computed. Another use case concerns the use of a PCE method for providing UQ metrics, e.g., variance-based confidence intervals, to be used for a reliability- or safety-related decision within a short time window. Other relevant use cases arise when multiple data-driven models must be computed using the same method, as is done in, e.g., k-fold cross-validation, bootstrapping, or ensemble modeling. Then, the computation time for a single model accumulates, possibly amounting to an undesired overall computational cost. In all aforementioned use cases, computation time can be a deciding factor regarding the suitability of a surrogate modeling or UQ method.

Training and test data sets The training and test data sets are generated by randomly sampling the considered models according to the joint probability distribution of the input parameters. We consider a small-data training regime, as is the usual case in real-world engineering applications. The size of the training data set Q depends on the specific test case. The size of the test data set is $Q' = 10^3$, unless stated otherwise. All numerical experiments are repeated for ten seeds, equivalently, for ten different training and test data sets.

Comparisons among PCE methods The MVSA PCE is compared against TD PCEs and, if feasible, sparse/adaptive PCEs. For the latter, we opt for the degree- and q-norm-adaptive (p/q-adaptive) LAR PCE method [17], which was consistently found to be the best option compared to alternatives, for example, p/q-adaptive OMP [18] and SP [20] PCEs. Note that the p/q-adaptive LAR PCE method must be applied element-wise for a vector-valued response, see for example [31–33].

Implementation specifics For the MVSA PCE method, we use an in-house `Python` implementation,[1] which is partially based on the `OpenTURNS` library [64]. The `OpenTURNS` library is also used for computing TD PCEs. Note that the employed OLS solver can resolve underdetermined systems – often arising when TD PCEs are computed – by returning the minimal 2-norm solution among multiple minimizing solutions. The p/q-adaptive LAR PCE method is implemented using the `MATLAB`-based `UQLab` software [65]. Also note that our in-house implementation is at disadvantage concerning its performance in terms of computation time, since `MATLAB` outperforms `Python` significantly in terms of computation speed. Still, the MVSA algorithm computes multidimensional PCEs much faster, as the numerical results show. We expect that this advantage will be even more pronounced if the MVSA PCE method is implemented with a more performant programming language. All computations have been performed using the 64-bit double-precision floating-point format on a standard x64-based Windows machine equipped with an Intel® Core™ i5-1245U processor (1.6 GHz, 10 cores, 12 logical processors) and 32 GB RAM.

[1] https://github.com/dlouk/mvsa-pce.



(a) Simply supported beam under uniform load. (b) Deflection over normalized beam length.

Fig. 1. (a) Sketch of the simply supported beam model. (b) Beam deflection for 10 realizations of the simply supported beam model parameters.

Table 1
Parameters of the simply supported beam model. All parameters follow a log-normal distribution with the given values for mean and standard deviation.

Parameter	Units	Notation	Mean	St. dev.
Width	m	w	$15 \cdot 10^{-2}$	$75 \cdot 10^{-4}$
Height	m	h	$3 \cdot 10^{-1}$	$15 \cdot 10^{-3}$
Length	m	L	5	$5 \cdot 10^{-2}$
Young's modulus	Pa	E	$3 \cdot 10^{10}$	$45 \cdot 10^{8}$
Load	N/m	P	10^4	$2 \cdot 10^3$
Dummy	–	d_1–d_{15}	10	1

6.1. Deflection of simply supported beam

We first consider the model of simply supported beam under uniform load, the input dimensions of which are artificially increased. Using this simple toy problem, our main goal is to illustrate the limitations of TD and sparse/adaptive PCEs that must be applied element-wise, especially for cases where high-dimensional inputs and outputs need to be addressed.

A sketch illustration of the simply supported beam model is shown in Fig. 1a. The beam's geometry is defined by its width w, height h, and length L. The Young's modulus of the beam is denoted with E. Under the uniform load P, the beam's deflection at a longitudinal coordinate ℓ_m along its length is given by

$$\delta(\ell_m) = \frac{P\,\ell_m\left(L^3 - 2\,\ell_m^2\,L + \ell_m^3\right)}{2\,E\,w\,h^3}, \tag{27}$$

where the coordinates ℓ_m, $m = 1, \dots, M$, are uniformly distributed along the length of the beam excluding the points of support, such that $\ell_m = mL/(M+1)$. These five parameters are assumed to be random variables following a log-normal distribution, see Table 1. The beam's deflection for different realizations of the input model parameters is shown in Fig. 1b.

Additionally, we consider 15 dummy parameters, denoted as d_i, $i = 1, \dots, 15$, each of which also follows a log-normal distribution, see Table 1. These parameters have no impact on the response of the model, hence, their distribution is irrelevant. Nevertheless, the artificially increased input dimensionality still affects the performance of the considered PCE methods, particularly that of the TD PCEs, for which the curse of (input) dimensionality can be detrimental.

In the following, three model response dimensions and three training data set sizes are considered, namely, $M \in \{10, 100, 1000\}$ and $Q \in \{50, 100, 150\}$. The beam deflection model (27) is approximated by MVSA, p/q-adaptive LAR, and TD PCEs, the latter for maximum polynomial degrees $p = 2$ and $p = 3$. Note that the PCEs are computed with respect to $\log(\mathbf{X})$ (normally distributed) instead of $\mathbf{X}$ (log-normally distributed), where $\mathbf{X} = \left(w, h, L, E, P, d_1, d_2, \dots, d_{15}\right)$ denotes the input random vector. Hence, according to the Wiener-Askey scheme [51], the polynomial basis consists of Hermite polynomials.

Fig. 2 shows the RMSE over the beam's length for all PCE surrogate models and combinations of response dimension M and training data set size Q. As is obvious, the resolution of the response and, accordingly, of the vector-valued RMSE depends on the response dimension M. For all PCEs, the RMSE is minimum near the points of support and maximum at the middle of the beam's length, as would be expected from the deflection variations observed in Fig. 1b. The TD PCEs result in comparatively low approximation accuracy, with little to no improvement as the size of the training data set increases. This is attributed to the high-dimensional input random vector ($N = 20$ inputs), despite the fact that the 15 dummy parameters have no impact on the response at all. The p/q-adaptive LAR and MVSA PCEs result in much more accurate surrogate models, which also improve significantly when

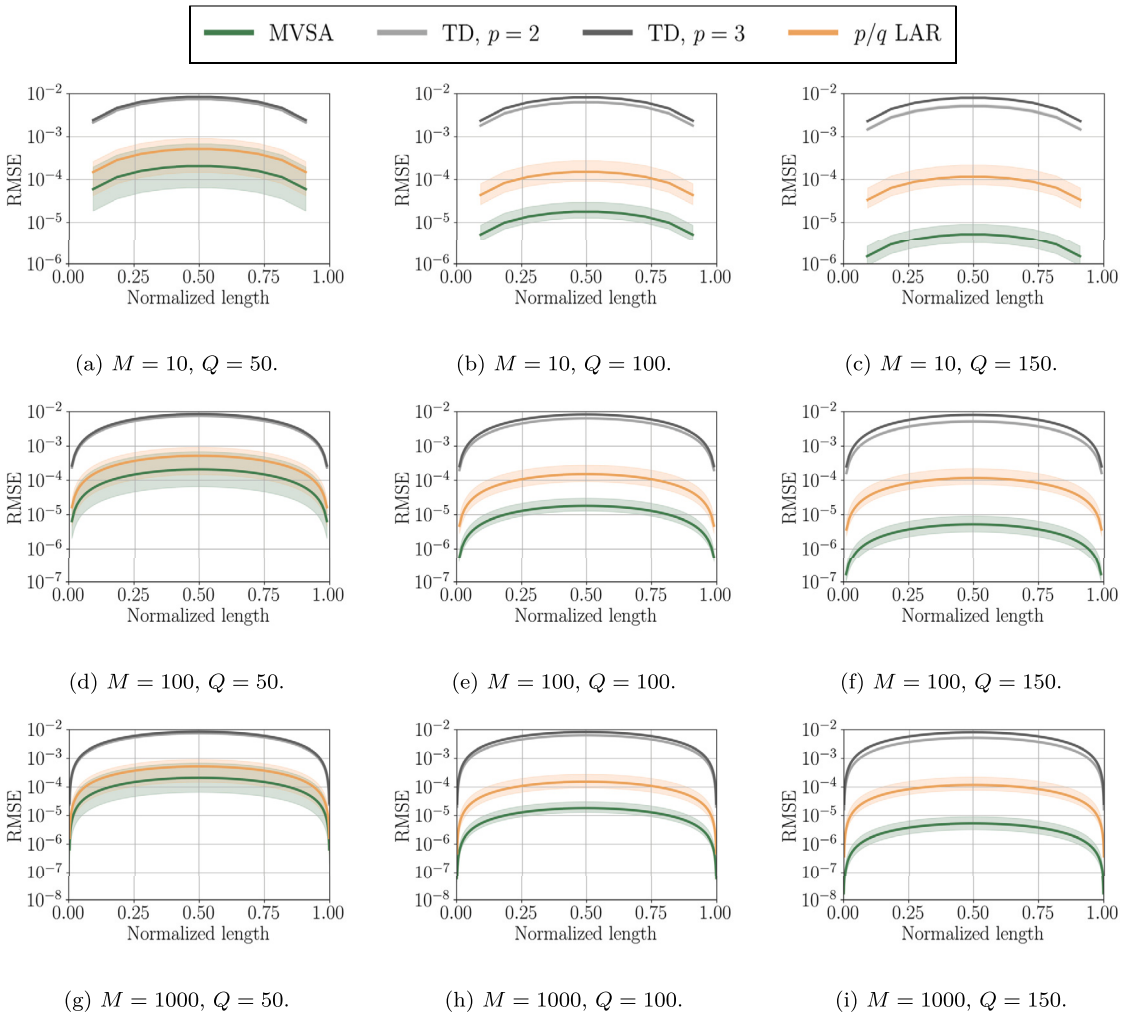
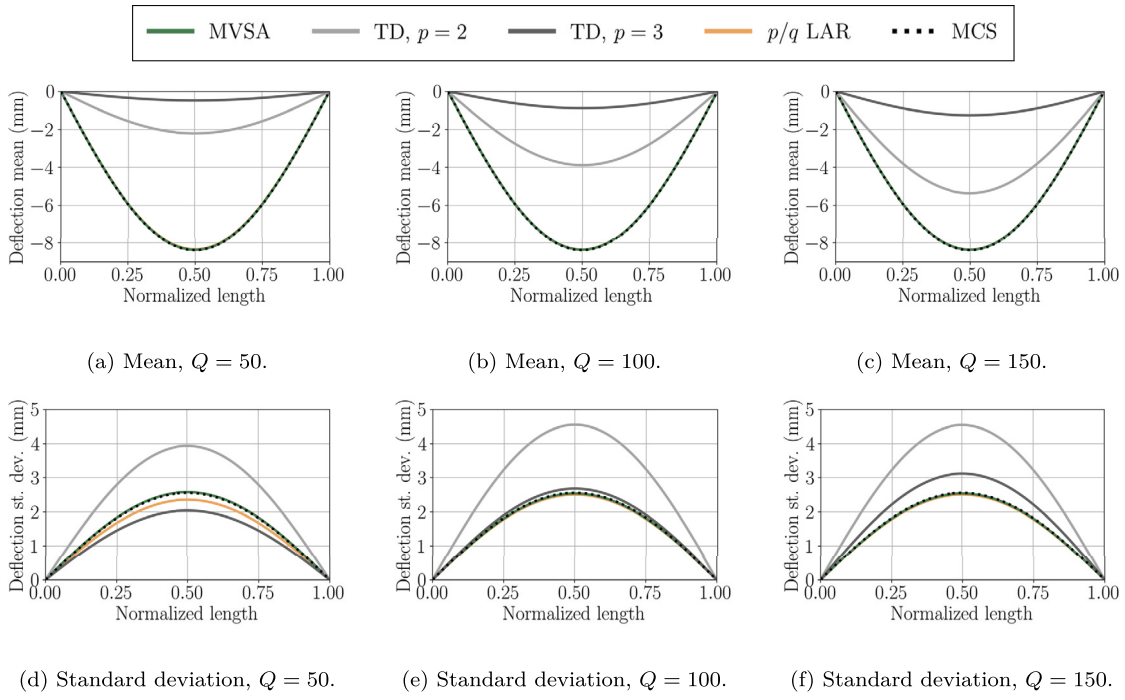


Fig. 2. Simply supported beam: RMSE over model response of the different PCEs for response dimension $M \in \{10, 100, 1000\}$ and training data set size $Q \in \{50, 100, 150\}$. The solid lines show average error values over 10 different training and test data sets. The shaded areas represent the difference between minimum and maximum errors over these 10 data sets.

more training data become available, exactly due to their ability to discern important from negligible input parameter contributions to the approximation. In this test case, the MVSA PCE is more accurate than the p/q-adaptive LAR and the TD PCEs by one and three orders of magnitude, respectively, for the same training data set size. Note that these observations regarding surrogate model accuracy remain consistent for all three response dimensions $M \in \{10, 100, 1000\}$.

Fig. 3 shows mean and standard deviation estimates obtained with each PCE method, along with MCS-based references computed with 10^5 random samples. In this case, only the response dimension $M = 1000$ is considered, such that the resolution of the response is the maximum one. Looking at the mean deflection, the estimates given by the MVSA and p/q-adaptive LAR PCEs are indistinguishable from one another, with both methods needing only $Q = 50$ training data points for an estimate identical to the MCS reference. In the case of the standard deviation, the MVSA PCE is again as accurate as the MCS reference for only $Q = 50$ training data points, while the p/q-adaptive LAR PCE is obviously less accurate. For $Q = 100$ and $Q = 150$, both methods yield standard deviation estimates identical to the reference. The mean and standard deviation estimates of the TD PCEs are mostly very inaccurate, as expected due to the curse of (input) dimensionality.

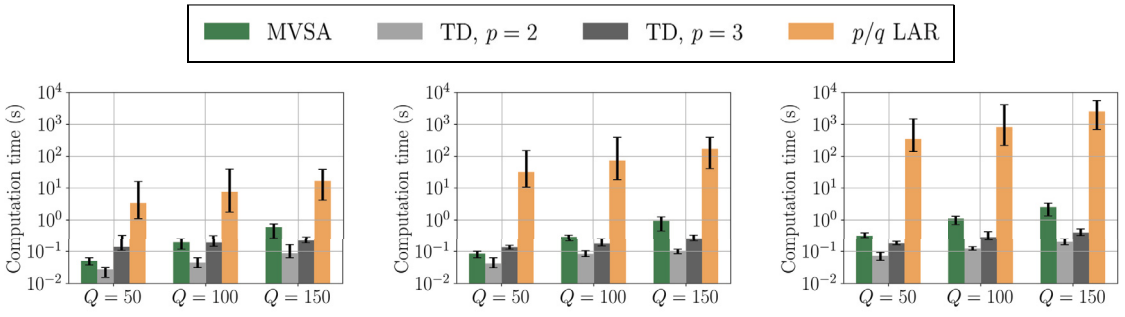
Fig. 4 shows the time needed to compute each PCE for all combinations of response dimension M and training data set size Q. This figure illustrates clearly the limitations of applying sprase/adaptive PCEs element-wise, particularly if the model's response is high-dimensional. As can be observed, the p/q-adaptive LAR PCE results in much longer computation times than the other PCE methods, reaching more than 40 minutes (on average) for $M = 1000$ and $Q = 150$. Contrarily, the MVSA PCE needs at most 3.5 seconds to be computed, while also resulting in a higher surrogate modeling and uncertainty estimation accuracy in this test case. The TD PCEs are computed even faster due to their fixed polynomial bases, however, this comes at no benefit due to their comparatively very low accuracy.



(a) Mean, $Q = 50$. (b) Mean, $Q = 100$. (c) Mean, $Q = 150$.

(d) Standard deviation, $Q = 50$. (e) Standard deviation, $Q = 100$. (f) Standard deviation, $Q = 150$.

Fig. 3. Simply supported beam: Mean and standard deviation of the model's response with dimension $M = 1000$, estimated with the different PCEs for training data set size $Q \in \{50, 100, 150\}$. The reference mean and standard deviation are computed via MCS with 10^5 random samples.



(a) $M = 10$. (b) $M = 100$. (c) $M = 1000$.

Fig. 4. Simply supported beam: Computation time of the different PCEs for response dimension $M \in \{10, 100, 1000\}$ and training data set size $Q \in \{50, 100, 150\}$. The colored bars show the average computation time over 10 different training data sets. The black error bars show the difference between minimum and maximum computation time over these 10 data sets.

Last, Table 2 shows the maximum total degree, i.e., $\max_{\mathbf{k}\in\Lambda} |\mathbf{k}|_1$, and the maximum univariate degree of the polynomial bases resulting from the MVSA and the p/q-adaptive LAR PCE methods. Note that in the case of the p/q-adaptive LAR PCE, the maximum degrees over all polynomial bases (i.e., one basis per output) are presented. Interestingly, the MVSA PCE results in higher total and univariate maximum degrees compared to the p/q-adaptive LAR PCE, which could to some extent explain its better performance. However, this is not a general phenomenon, as we will see in the next section.

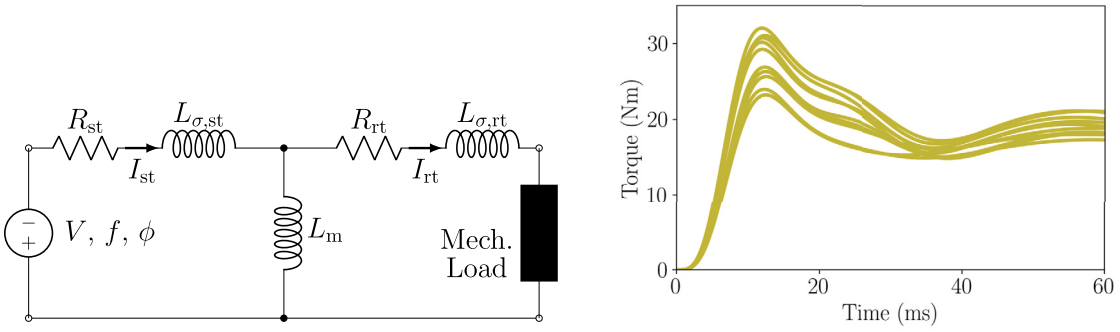
6.2. Start-up torque of induction motor

As second test case, we consider an induction motor connected to a power grid and to a mechanical load. The power grid has phase-voltage amplitude $V = 210$ V, frequency $f = 50$ Hz, and initial phase $\phi = 0°$. The motor consists of a stator, where a winding is organized in three phases and $n_{\text{pp}} = 2$ pole pairs, and a rotor that rotates at the mechanical speed ω_{me}. The torque of the mechanical load is given by

$$T_{\text{ld}}\left(\omega_{\text{me}}\right) = \text{sign}\left(\omega_{\text{me}}\right)\left(\gamma\omega_{\text{me}}^2 + \beta\left|\omega_{\text{me}}\right| + \alpha\right), \tag{28}$$

Table 2
Simply supported beam: Maximum total and univariate degrees of the MVSA and the p/q-adaptive LAR PCEs.

Training data set size Q	Maximum total degree		Maximum univariate degree	
	MVSA	p/q LAR	MVSA	p/q LAR
50	3	3	3	3
100	4	3	4	3
150	5	3	4	3



(a) Equivalent circuit of induction motor. (b) Electromagnetic torque during motor start-up.

Fig. 5. (a) Equivalent circuit model of an induction motor. (b) Electromagnetic torque generated during the start-up of the induction motor for 10 realizations of the model parameters.

where the coefficients α, β, and γ are used for modeling static, sliding, and air friction, respectively. The mechanical equation of motion then reads

$$\frac{\mathrm{d}\omega_{\mathrm{me}}}{\mathrm{d}t} = \frac{T_{\mathrm{em}} - T_{\mathrm{ld}}\left(\omega_{\mathrm{me}}\right)}{J_{\mathrm{rt}} + J_{\mathrm{ld}}}, \tag{29}$$

where J_{rt} and J_{ld} denote the moment of inertia of the rotor and the load, respectively, and T_{em} is the electromagnetic torque. Using the per-phase equivalent circuit model of the motor depicted in Fig. 5a, the electromagnetic torque is given by

$$T_{\mathrm{em}} = \frac{3R_{\mathrm{rt}}\left|I_{\mathrm{rt}}\right|^2}{\omega_{\mathrm{syn}} - \omega_{\mathrm{me}}}, \tag{30}$$

where $\omega_{\mathrm{syn}} = 2\pi f / n_{\mathrm{pp}}$ is the synchronous speed of the electromagnetic field generated by the stator winding within the air gap of the motor. The rotor resistance R_{rt} is assumed to be known, while the rotor current I_{rt} can be derived from the equivalent circuit's voltage and current equations [66]. To simulate the start-up phase of the induction machine, where the rotor is initially at standstill ($\omega_{\mathrm{me}} = 0$), the mechanical equation of motion (29) is coupled to the electrical ordinary differential equation

$$v(t) = Ri(t) + L\frac{\mathrm{d}i(t)}{\mathrm{d}t}, \tag{31}$$

which is derived separately for the three phases in the stator and the rotor, thus resulting in a system of six equations [66].

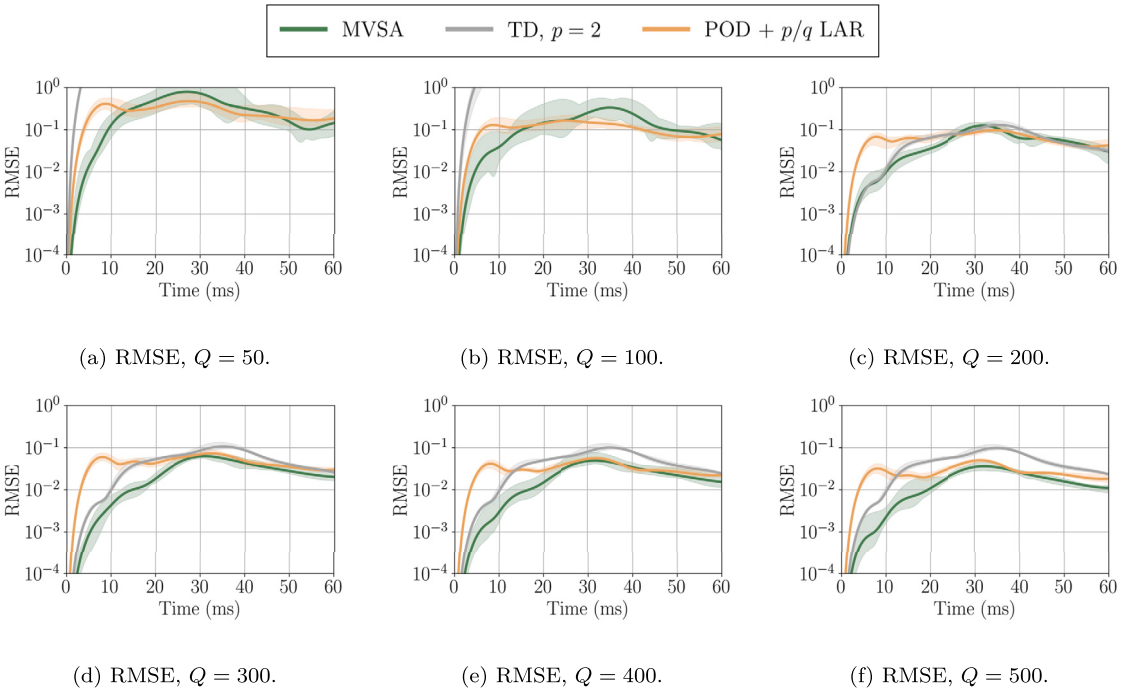
The simulation model is implemented using the `gym-motor` software [67] and comprises the $N = 13$ input parameters listed in Table 3. The grid parameters, i.e., phase-voltage amplitude, frequency, and initial angle, vary uniformly within given ranges, while the equivalent circuit and mechanical load parameters follow normal distributions with mean equal to their nominal value and standard deviation equal to 5% of the nominal value, see Table 3. Fig. 5b shows the electromagnetic torque of the induction motor for different parameter realizations, where each torque trajectory contains $M = 1201$ time steps.

The full response dimension is prohibitive for the element-wise application of the p/q-adaptive LAR PCE, therefore a dimension reduction of the response by means of the POD method is attempted. The number of POD modes to be retained is computed using the optimal shrinkage of singular values approach of Gavish and Donoho [68–70], in order to avoid heuristics that often lead to inaccurate approximations, e.g., based on energy or variance preservation considerations (see [35]). This results in $M' \ll M$ POD modes, where typically M' varies depending on the available training data set. In this test case, $M' \in [48, 51]$ for $Q \geq 100$ and $M' = 24$ for $Q = 50$. Contrarily, the MVSA and TD PCEs are applied to the full model response.

For each PCE method and for training data sets of size $Q = 50, \ldots, 500$, Fig. 6 presents the RMSE over the model's response. For $Q = 50, 100$ training data points, the MVSA and POD-based p/q-adaptive LAR PCEs are significantly more accurate than the TD PCE, which is here considered for a maximum polynomial degree $p = 2$ only, as this is the only choice with acceptable accuracy results.

Table 3
Parameters of the induction motor model. Grid parameters vary uniformly, namely, the phase-voltage amplitude V within $[85\%, 100\%]$ of its nominal value, the frequency f within $[49, 51]$ Hz, and the initial phase ϕ within $[0, 15]$ degrees. The remaining parameters follow normal distributions with their nominal value as mean and 5% of the nominal value as standard deviation.

Parameter	Units	Notation	Nominal value
Stator resistance	Ω	R_{st}	$2933.8 \cdot 10^{-3}$
Rotor resistance	Ω	R_{rt}	$1355.0 \cdot 10^{-3}$
Magnetization inductance	H	L_{m}	$1437.5 \cdot 10^{-4}$
Stator leakage inductance	H	$L_{\sigma,\mathrm{st}}$	$58.7 \cdot 10^{-4}$
Rotor leakage inductance	H	$L_{\sigma,\mathrm{rt}}$	$58.7 \cdot 10^{-4}$
Moment of inertia of rotor	$\mathrm{kg\,m^2}$	J_{rt}	$1.1 \cdot 10^{-3}$
Moment of inertia of load	$\mathrm{kg\,m^2}$	J_{ld}	10^{-3}
Constant load torque coefficient	N m	α	10^{-3}
Linear load torque coefficient	N m s	β	10^{-3}
Quadratic load torque coefficient	$\mathrm{N\,m\,s^2}$	γ	10^{-3}
Phase-voltage amplitude	V	V	210.0
Frequency	Hz	f	50.0
Initial phase	°	ϕ	0.0



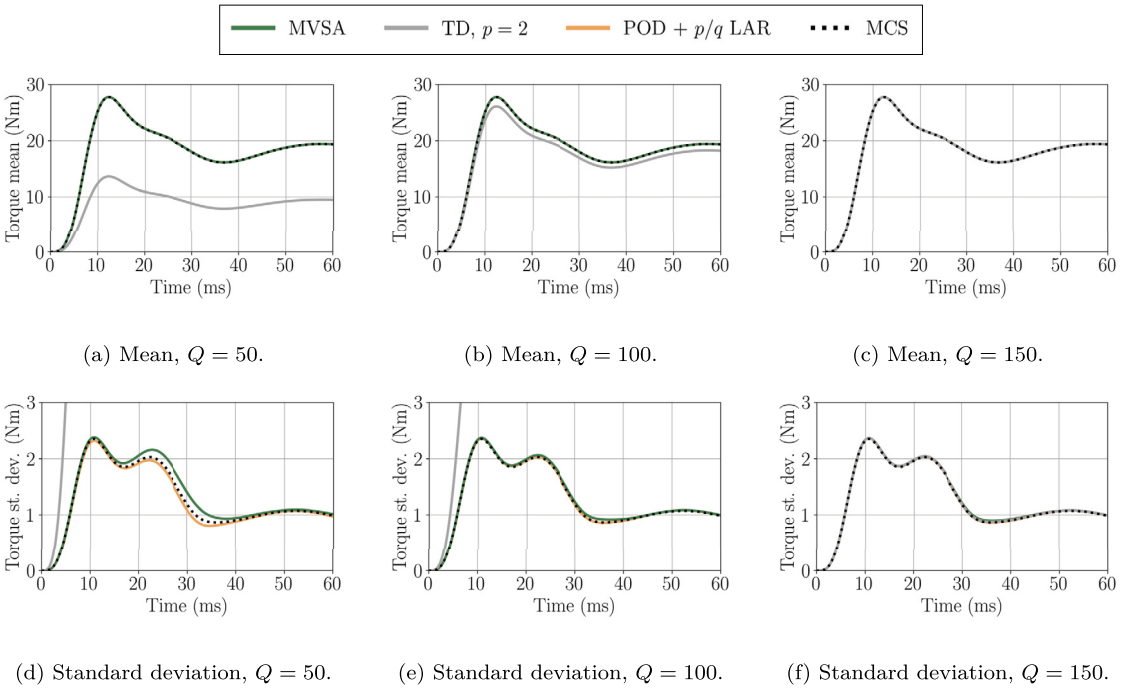
(a) RMSE, $Q = 50$. (b) RMSE, $Q = 100$. (c) RMSE, $Q = 200$.

(d) RMSE, $Q = 300$. (e) RMSE, $Q = 400$. (f) RMSE, $Q = 500$.

Fig. 6. Induction motor: RMSE over model response of the different PCEs for training data set size $Q = 50, \ldots, 500$. The MVSA and TD PCEs are applied to the full response ($M = 1201$). The p/q-adaptive LAR PCE is applied to a reduced response dimension ($M' = 24$ for $Q = 50$ and $M' \in [48, 51]$ for $Q \geq 100$) based on POD. The solid lines show average error values over 10 different training and test data sets. The shaded areas represent the difference between minimum and maximum errors over these 10 data sets.

For $Q = 200$, the TD PCE shows comparable accuracy to the other two PCEs, however, it does not improve further as the training data set size increases and is eventually outperformed. For $Q \geq 200$ training data points, the MVSA PCE is consistently more accurate than the TD PCE and comparably or more accurate than the POD-based p/q-adaptive LAR PCE over the full time range. Interestingly, the latter is consistently less accurate than the other two PCEs in the time range $0 - 15$ ms.

Mean and standard deviation estimates in comparison with MCS references are presented in Fig. 7. Already for $Q = 50$ training data points, the MVSA and POD-based LAR PCEs yield mean estimates identical to the reference. The latter is slightly more accurate than the former in its standard deviation estimate for $Q = 50$, while the estimates of both PCEs are almost indistinguishable to the reference for $Q = 100$ and $Q = 150$. The TD PCE also results in very accurate moment estimates for $Q = 150$, however, it is significantly less accurate for smaller training data set sizes, especially concerning the standard deviation.



(a) Mean, $Q = 50$. (b) Mean, $Q = 100$. (c) Mean, $Q = 150$.

(d) Standard deviation, $Q = 50$. (e) Standard deviation, $Q = 100$. (f) Standard deviation, $Q = 150$.

Fig. 7. Induction motor: Mean and standard deviation of the model's response, estimated with the different PCEs for training data set size $Q \in \{50, 100, 150\}$. The reference mean and standard deviation are computed via MCS with $15 \cdot 10^3$ random samples. The MVSA and TD PCEs are applied for the full response ($M = 1201$). The p/q-adaptive LAR PCE is applied for a reduced dimension ($M' = 24$ for $Q = 50$ and $M' \in [48, 51]$ for $Q \geq 100$) based on POD.

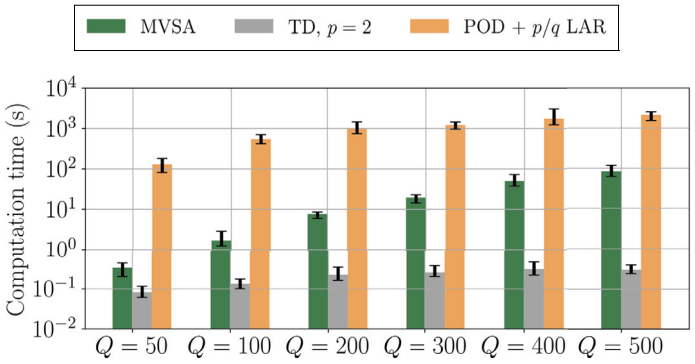


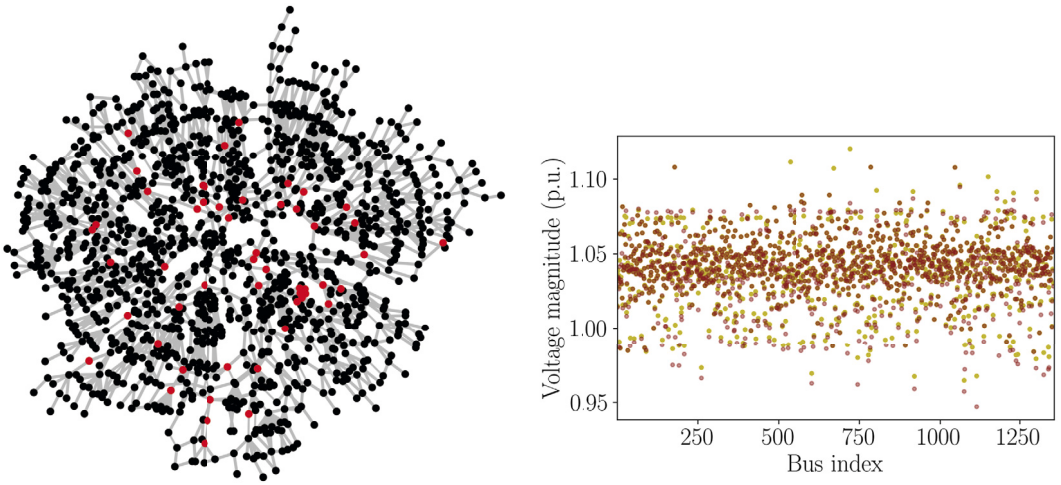
Fig. 8. Induction motor: Computation time of the different PCEs for training data set size $Q = 50, \ldots, 500$. The MVSA and TD PCEs are applied for the full response ($M = 1201$). The p/q-adaptive LAR PCE is applied for a reduced dimension ($M' = 24$ for $Q = 50$ and $M' \in [48, 51]$ for $Q \geq 100$) based on POD. The colored bars show the average computation time over 10 different training data sets. The black error bars show the difference between minimum and maximum computation time over these 10 data sets.

The computation time of each PCE for different training data set sizes is shown in Fig. 8. As expected, the TD PCE is computed almost instantaneously due to its fixed basis. However, for the same reason, it is comparatively less accurate in terms of surrogate modeling and uncertainty estimation accuracy. The MVSA PCE is computed, on average, $\times 400 - \times 25$ faster than the POD-based p/q-adaptive LAR PCE, despite the latter being applied for a $\times 23$ reduced response dimension at minimum. Notably, even for a reduced dimension of maximum size $M' = 51$, the computation of the POD-based p/q-adaptive LAR PCE still requires a substantial amount of time, especially for larger training data sets, e.g., 35 minutes (on average) for $Q = 500$. Comparatively, for the same training data set size, the MVSA PCE applied to the full model response ($M = 1201$) is computed in less than 2 minutes in the worst case.

Last, Table 4 shows the maximum total degree, i.e., $\max_{\mathbf{k} \in \Lambda} |\mathbf{k}|_1$, and the maximum univariate degree of the polynomial bases resulting from the MVSA and the p/q-adaptive LAR PCE methods. Note that in the case of the p/q-adaptive LAR PCE, the maximum degrees over all polynomial bases (i.e., one basis per output) are presented. Unlike Section 6.1, in this test case, the p/q-adaptive LAR PCE yields higher total and univariate degrees than the MVSA PCE. This discrepancy could also explain the difference in their approximation results, especially for the smaller training data set sizes. Also note that a TD PCE with maximum polynomial degree equal to 7, i.e., the maximum total degree of the MVSA PCE method, would result in 77520 coefficients, rendering the method usable only a big data training regime.

Table 4
Induction motor: Maximum total and univariate degrees of the MVSA and the p/q-adaptive LAR PCEs.

Training data set size Q	Maximum total degree		Maximum univariate degree	
	MVSA	p/q LAR	MVSA	p/q LAR
50	4	8	4	7
100	4	9	4	7
200	4	9	4	8
300	5	9	4	8
400	7	9	7	9
500	7	9	7	9



(a) Power grid 1354-PEGASE. (b) Voltage magnitude per network bus.

Fig. 9. (a) The 1354-PEGASE power grid. Network buses are shown in black, transmission lines in gray, and the 52 random loads in red. (b) Voltage magnitude per network bus in the per-unit (p.u.) system for two realizations of the 52 random loads, respectively shown in yellow and brown.

6.3. Power flow in electrical grid

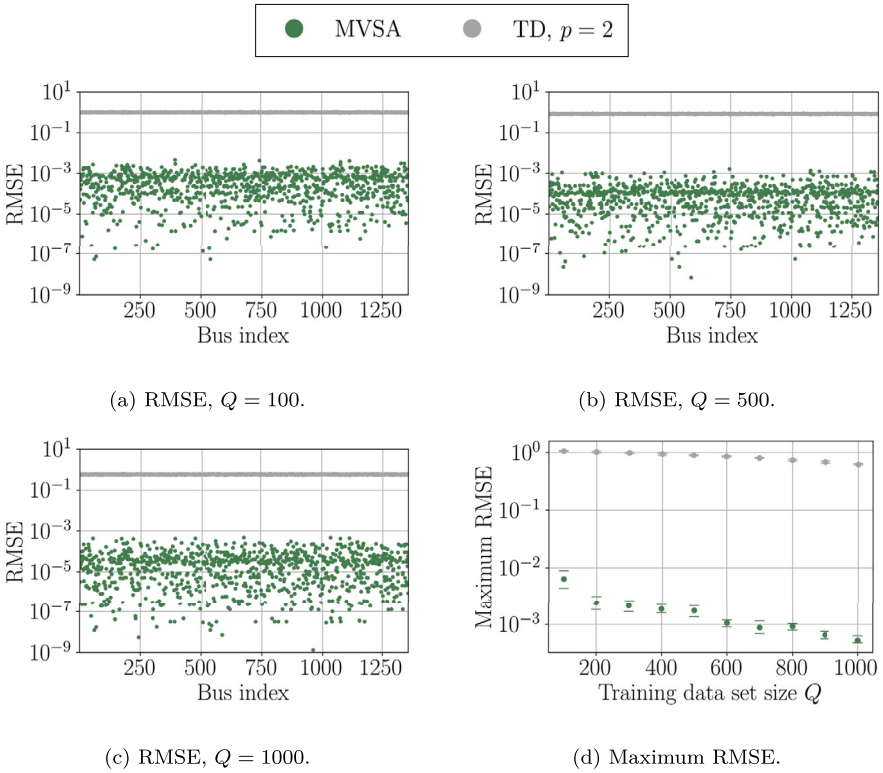
As third and final test case, we consider a power flow (also referred to as load flow) study based on a power grid model that represents part of the European high-voltage transmission network. This network is a standard power system test case known as the 1354-PEGASE grid [71], where the number 1354 refers to the number of buses, i.e., network nodes upon which other elements, e.g., generators, loads, transformers, transmission lines, etc., are connected. An implementation of this grid is available in the open-source software `pandapower` [72]. The network is shown in Fig. 9a.

The power flow problem is formulated using the so-called power balance equations

$$0 = -\mathrm{Re}\left\{S_i\right\} + \sum_{j=1}^{J} \left|V_i\right| \left|V_j\right| \left(G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}\right), \tag{32a}$$

$$0 = -\mathrm{Im}\left\{S_i\right\} + \sum_{j=1}^{J} \left|V_i\right| \left|V_j\right| \left(G_{ij}\sin\theta_{ij} + B_{ij}\cos\theta_{ij}\right), \tag{32b}$$

where S_i denotes the apparent power injected at the i-th bus, $\mathrm{Re}\left\{S_i\right\} = \left|S_i\right|\cos\phi_i$ and $\mathrm{Im}\left\{S_i\right\} = \left|S_i\right|\sin\phi_i$ are the active and reactive power for power factor ϕ_i, G_{ij} and B_{ij} denote the real and imaginary parts of the elements of the network's nodal admittance matrix, $\left|V_i\right|$ and $\left|V_j\right|$ denote the voltage magnitude at buses i and j, and $\theta_{ij} = \theta_i - \theta_j$ is the voltage angle difference between buses i and j. The known and unknown variables in the balance equations (32) are determined based on the type of the i-th bus. That is, the active and reactive power are known for so-called load buses, while the active power and the voltage magnitude are known for so-called generator buses. One bus is selected as angular reference, known as the slack or swing bus. The resulting nonlinear system of equations is solved numerically, using variations of the Newton-Raphson method [73].



(a) RMSE, $Q = 100$.

(b) RMSE, $Q = 500$.

(c) RMSE, $Q = 1000$.

(d) Maximum RMSE.

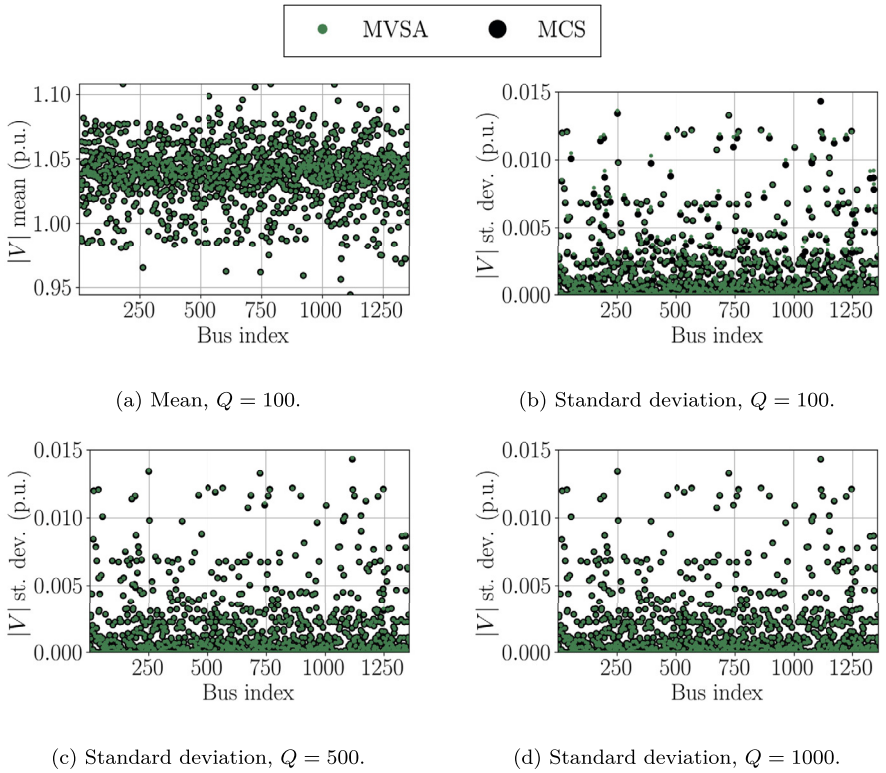
Fig. 10. Power grid: RMSE in voltage magnitude per network bus and maximum RMSE over all buses for the MVSA and TD PCEs trained with data sets of increasing size. (a)-(c) show average error values over 10 different training and test data sets. Figure (d) shows the average (circles) and the minimum/maximum (horizontal bars) values of the maximum RMSE over these 10 data sets.

Following prior works on uncertainty modeling for power grids [74], the loads of 52 buses are considered to be uncertain. In particular, we introduce the random variables X_n and the corresponding Beta distributions $B_n\,(\alpha = 3, \beta = 2, l = 0, u = 1)$, $n = 1, \dots, N = 52$, where α, β are shape parameters, and l, u are the lower and upper limit of the random variable's support. The random loads are then given as $S_n = S_n^{\max} X_n$, where $S_n^{\max}$ denotes the maximum allowed load value. Note that the maximum loads vary significantly, i.e., $S_n^{\max} \in [2.8, 1175.0]$ MVA. The power factors ϕ_n remain unaffected. The network buses corresponding to the random loads are marked with red color in Fig. 9a. The quantity of interest is chosen to be the voltage magnitude $|V_i|$ on each of the $M = 1354$ buses, as this is the most critical size regarding network stability. Each bus is affected much differently by the random loads due to their position on the grid and the different maximum load values. Fig. 9b shows the voltage magnitude per network bus for two realizations of the random loads. Note that the voltage magnitude is computed in the per-unit (p.u) system, that is, relative to the nominal rated voltage of each bus, as is commonly done in power flow studies. The rated voltage is 220 kV for 1113 buses and 380 kV for 241 buses.

In this test case, the MVSA and TD PCEs are employed for training data sets of size $Q = 100, 200, \dots, 1000$. The TD PCE has maximum polynomial degree $p = 2$, however, as will be observed, even the largest training datasets are not sufficient for its training, due to the curse of (input) dimensionality. The p/q-adaptive LAR PCE is here omitted, due to the fact that the output dimension is too large for its element-wise application and POD does not result in sufficient dimension reduction.

Surrogate model accuracy results are shown in Fig. 10, where 10a, 10b, and 10c show the RMSE per network bus for three training data set sizes, and 10d shows the maximum RMSE over all buses for all training data set sizes. The large input dimensionality ($N = 52$) is prohibitive for training the TD PCE with the available data sets and only minor improvement in accuracy is observed, despite the tenfold increase in training data set size. In comparison, the maximum RMSE of the MVSA PCE is consistently more than two orders of magnitude smaller and is reduced by more than one order of magnitude as the training data set size increases. However, for most buses, the RMSE is several orders of magnitude smaller than the maximum one. Note that a number of buses have been omitted in Figs. 10a, 10b, and 10c, as the voltage magnitude of said buses is only minimally or not at all affected by the random load variations, hence, the corresponding RMSEs are almost zero.

The differences in impact of the random load variations upon the network bus voltages become even more evident by looking at the mean and standard deviation results presented in Fig. 11. That is, for several buses, a very small to negligible standard deviation can be observed, while the standard deviation is quite significant for other buses. The MVSA PCE approximates the mean very accurately with only $Q = 100$ training data points. Contrarily, the same training data set size leads to a poor standard deviation estimate for many network buses. A significant improvement can be observed for $Q = 500$ and the estimate becomes almost identical to the MCS-based reference for $Q = 1000$. The results of the TD PCE are here omitted, as both mean and standard deviation estimates are very inaccurate.



(a) Mean, $Q = 100$.

(b) Standard deviation, $Q = 100$.

(c) Standard deviation, $Q = 500$.

(d) Standard deviation, $Q = 1000$.

Fig. 11. Power grid: Mean and standard deviation of the voltage magnitude per network bus, estimated with the MVSA PCE for training data set size $Q \in \{100, 500, 1000\}$. The reference mean and standard deviation are computed via MCS with $15 \cdot 10^3$ random samples.

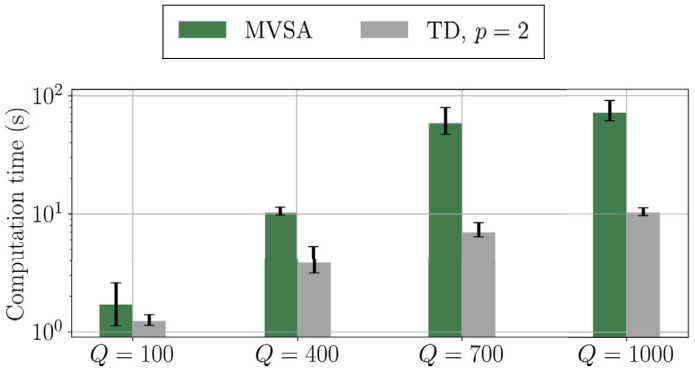


Fig. 12. Power grid: Computation time of the MVSA and TD PCEs for training data set size $Q \in \{100, 400, 700, 1000\}$. The colored bars show the average computation time over 10 different training data sets. The black error bars show the difference between minimum and maximum computation time over these 10 data sets.

Fig. 12 presents the computation times of the MVSA and TD PCEs for training data set sizes ranging from $Q = 100$ to $Q = 1000$ training data points. Similar to the previous use cases, the TD PCE is computed much faster due to its a priori fixed basis, however, for the same reason results in poor approximation and uncertainty estimation results. In comparison, the computation time of the MVSA PCE remains acceptable, i.e., 91 s in the worst case, while also yielding accurate surrogate models and uncertainty estimates.

Last, Table 5 shows the maximum total degree, i.e., $\max_{\mathbf{k}\in\Lambda} |\mathbf{k}|_1$, and the maximum univariate degree of the polynomial bases resulting from the MVSA PCE method. Note that a TD PCE with maximum polynomial degree equal to 3, i.e., the maximum total degree of the MVSA PCE method, would in this case result in 26235 coefficients, rendering the method usable only a big data training regime.

7. Discussion, conclusion, and outlook

This work presented a novel basis-adaptive method for the construction of polynomial chaos expansions (PCEs) of high-dimensional, vector-valued model responses. The corresponding algorithm is called multivariate sensitivity-adaptive (MVSA) due

Table 5
Power grid: Maximum total and univariate degrees of the MVSA PCE method.

Training data set size Q	Maximum total degree	Maximum univariate degree
100	2	2
500	3	3
1000	3	3

to the fact that it employs a multivariate sensitivity analysis metric as criterion for the adaptive expansion of the polynomial basis. The method has the advantage of being able to address problems with moderately high-dimensional model inputs, i.e., in the orders of tens, and very high-dimensional model responses i.e., up to the order of thousands, at the same time.

The MVSA PCE method has been applied to three numerical test cases, each featuring moderately high-dimensional model inputs and very high-dimensional model responses. For all three test cases, the MVSA PCE has been evaluated in terms of computational efficiency, surrogate modeling accuracy, and uncertainty estimation accuracy. Computational efficiency refers to training data demand versus approximation and uncertainty estimation accuracy, as well as to the time needed to compute the PCE. The method was compared against standard total-degree (TD) PCEs with an a priori fixed polynomial basis, and against a state-of-the-art degree- and q-norm-adaptive (p/q-adaptive) PCE method based on least angle regression (LAR). Due to the fixed basis, TD PCEs are computed faster than the MVSA PCE. However, the latter is typically much more accurate for the same size of training data set. Additionally, the MVSA PCE remains applicable in cases with comparatively high-dimensional model inputs, where TD PCEs become unusable due to the curse of (input) dimensionality. Compared to the p/q-adaptive LAR PCE, the MVSA PCE was found to offer either better or at least comparable accuracy for the same training data set sizes. Additionally, the MVSA PCE is computed several times faster, even for cases where the p/q-adaptive LAR PCE is combined with a dimension reduction method. Last, the MVSA PCE remains applicable in cases where the p/q-adaptive LAR PCE cannot be used at all, e.g., for high-dimensional model responses without effective dimension reduction.

The MVSA PCE method developed in this work can be considered as a useful surrogate modeling and uncertainty quantification (UQ) tool already in its current form. Nonetheless, a number of possible developments could further enhance its capabilities. First, problems with reduced regularity could be addressed by integrating multi-element approaches [44]. Second, the method could be combined with dedicated active learning techniques in an effort to further reduced training data demand [75]. Last, recent developments on physics-informed PCEs [76] offer the possibility of physically conforming PCE-based surrogate models, rather than pure data-driven ones. Such developments will be investigated in dedicated follow-up works.

CRediT authorship contribution statement

Dimitrios Loukrezis: Writing – original draft, Validation, Software, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Eric Diehl:** Writing – review & editing, Validation, Software, Investigation. **Herbert De Gersem:** Writing – review & editing, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The work of Dimitrios Loukrezis and Herbert De Gersem is partially supported by the joint DFG/FWF Collaborative Research Centre CREATOR (DFG: Project-ID 492661287/TRR 361; FWF: 10.55776/F90) at TU Darmstadt, TU Graz and JKU Linz. The authors would like to thank the reviewers and editors for their constructive feedback.

Data availability

Data will be made available on request.

References

[1] D. Shen, H. Wu, B. Xia, et al., Polynomial chaos expansion for parametric problems in engineering systems: a review, IEEE Syst. J. 14 (3) (2020) 4500–4514, https://doi.org/10.1109/JSYST.2019.2957664.
[2] H. Lim, L. Manuel, Distribution-free polynomial chaos expansion surrogate models for efficient structural reliability analysis, Reliab. Eng. Syst. Saf. 205 (2021) 107256, https://doi.org/10.1016/j.ress.2020.107256.
[3] D. Loukrezis, H. De Gersem, Power module heat sink design optimization with ensembles of data-driven polynomial chaos surrogate models, e-Prime – Adv. Electr. Eng. Electron. Energy 2 (2022) 100059, https://doi.org/10.1016/j.prime.2022.100059.
[4] A. Suryawanshi, D. Ghosh, Reliability based optimization in aeroelastic stability problems using polynomial chaos based metamodels, Struct. Multidiscip. Optim. 53 (2016) 1069–1080, https://doi.org/10.1007/s00158-015-1322-0.

[5] A. Thelen, X. Zhang, O. Fink, et al., A comprehensive review of digital twin—part 1: modeling and twinning enabling technologies, Struct. Multidiscip. Optim. 65 (12) (2022) 354, https://doi.org/10.1007/s00158-022-03425-4.
[6] A. Thelen, X. Zhang, O. Fink, et al., A comprehensive review of digital twin—part 2: roles of uncertainty quantification and optimization, a battery digital twin, and perspectives, Struct. Multidiscip. Optim. 66 (1) (2023) 1, https://doi.org/10.1007/s00158-022-03410-x.
[7] T. Crestaux, O. Le Maitre, J.-M. Martinez, Polynomial chaos expansion for sensitivity analysis, Reliab. Eng. Syst. Saf. 94 (7) (2009) 1161–1172, https://doi.org/10.1016/j.ress.2008.10.008.
[8] O.M. Knio, O. Le Maitre, Uncertainty propagation in CFD using polynomial chaos decomposition, Fluid Dyn. Res. 38 (9) (2006) 616, https://doi.org/10.1016/j.fluiddyn.2005.12.003.
[9] B. Sudret, Global sensitivity analysis using polynomial chaos expansions, Reliab. Eng. Syst. Saf. 93 (7) (2008) 964–979, https://doi.org/10.1016/j.ress.2007.04.002.
[10] J. Zhang, Modern Monte Carlo methods for efficient uncertainty quantification and propagation: a survey, WIREs Comput. Stat. 13 (5) (2021) e1539, https://doi.org/10.1002/wics.1539.
[11] M. Hadigol, A. Doostan, Least squares polynomial chaos expansion: a review of sampling strategies, Comput. Methods Appl. Mech. Eng. 332 (2018) 382–407, https://doi.org/10.1016/j.cma.2017.12.019.
[12] G. Migliorati, F. Nobile, E. Von Schwerin, et al., Analysis of discrete L2 projection on polynomial spaces with random evaluations, Found. Comput. Math. 14 (3) (2014) 419–456, https://doi.org/10.1007/s10208-013-9186-4.
[13] E. Torre, S. Marelli, P. Embrechts, et al., Data-driven polynomial chaos expansion for machine learning regression, J. Comput. Phys. 388 (2019) 601–623, https://doi.org/10.1016/j.jcp.2019.03.039.
[14] A. Cohen, G. Migliorati, Multivariate approximation in downward closed polynomial spaces, in: Contemporary Computational Mathematics-A Celebration of the 80th Birthday of Ian Sloan, 2018, pp. 233–282.
[15] N. Lüthen, S. Marelli, B. Sudret, Sparse polynomial chaos expansions: literature survey and benchmark, SIAM/ASA J. Uncertain. Quantificat. 9 (2) (2021) 593–649, https://doi.org/10.1137/20M1315774.
[16] N. Lüthen, S. Marelli, B. Sudret, Automatic selection of basis-adaptive sparse polynomial chaos expansions for engineering applications, Int. J. Uncertain. Quantificat. 12 (3) (2022) 49–74, https://doi.org/10.1615/Int.J.UncertaintyQuantification.2021036153.
[17] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, J. Comput. Phys. 230 (6) (2011) 2345–2367, https://doi.org/10.1016/j.jcp.2010.12.021.
[18] J.D. Jakeman, M.S. Eldred, K. Sargsyan, Enhancing l1-minimization estimates of polynomial chaos expansions using basis selection, J. Comput. Phys. 289 (2015) 18–34, https://doi.org/10.1016/j.jcp.2015.02.025.
[19] N. Alemazkoor, H. Meidani, Divide and conquer: an incremental sparsity promoting compressive sampling approach for polynomial chaos expansions, Comput. Methods Appl. Mech. Eng. 318 (2017) 937–956, https://doi.org/10.1016/j.cma.2017.01.039.
[20] P. Diaz, A. Doostan, J. Hampton, Sparse polynomial chaos expansions via compressed sensing and D-optimal design, Comput. Methods Appl. Mech. Eng. 336 (2018) 640–666, https://doi.org/10.1016/j.cma.2018.03.020.
[21] K. Sargsyan, C. Safta, H.N. Najm, et al., Dimensionality reduction for complex models via Bayesian compressive sensing, Int. J. Uncertain. Quantificat. 4 (1) (2014) 63–93, https://doi.org/10.1615/Int.J.UncertaintyQuantification.2013006821.
[22] P. Tsilifis, X. Huan, C. Safta, et al., Compressive sensing adaptation for polynomial chaos expansions, J. Comput. Phys. 380 (2019) 29–47, https://doi.org/10.1016/j.jcp.2018.12.010.
[23] B.-Y. Zhang, Y.-Q. Ni, A novel sparse polynomial chaos expansion technique with high adaptiveness for surrogate modelling, Appl. Math. Model. 121 (2023) 562–585, https://doi.org/10.1016/j.apm.2023.05.005.
[24] S. Abraham, M. Raisee, G. Ghorbaniasl, et al., A robust and efficient stepwise regression method for building sparse polynomial chaos expansions, J. Comput. Phys. 332 (2017) 461–474, https://doi.org/10.1016/j.jcp.2016.12.015.
[25] W. He, Y. Zeng, G. Li, An adaptive polynomial chaos expansion for high-dimensional reliability analysis, Struct. Multidiscip. Optim. 62 (4) (2020) 2051–2067, https://doi.org/10.1007/s00158-020-02594-4.
[26] M. Thapa, S.B. Mulani, R.W. Walters, Adaptive weighted least-squares polynomial chaos expansion with basis adaptivity and sequential adaptive sampling, Comput. Methods Appl. Mech. Eng. 360 (2020) 112759, https://doi.org/10.1016/j.cma.2019.112759.
[27] D. Loukrezis, A. Galetzka, H. De Gersem, Robust adaptive least squares polynomial chaos expansions in high-frequency applications, Int. J. Numer. Model. 33 (6) (2020) e2725, https://doi.org/10.1002/jnm.2725.
[28] R. Ghanem, P.D. Spanos, Polynomial chaos in stochastic finite elements, J. Appl. Mech. 57 (1) (1990) 197–202, https://doi.org/10.1115/1.2888303.
[29] C. Lataniotis, S. Marelli, B. Sudret, Extending classical surrogate modeling to high dimensions through supervised dimensionality reduction: a data-driven approach, Int. J. Uncertain. Quantificat. 10 (1) (2020), https://doi.org/10.1615/Int.J.UncertaintyQuantification.2020031935.
[30] K. Kontolati, D. Loukrezis, D.G. Giovanis, et al., A survey of unsupervised learning methods for high-dimensional uncertainty quantification in black-box-type problems, J. Comput. Phys. 464 (2022) 111313, https://doi.org/10.1016/j.jcp.2022.111313.
[31] C.V. Mai, B. Sudret, Surrogate models for oscillatory systems using sparse polynomial chaos expansions and stochastic time warping, SIAM/ASA J. Uncertain. Quantificat. 5 (1) (2017) 540–571, https://doi.org/10.1137/16M1083621.
[32] V. Yaghoubi, S. Marelli, B. Sudret, et al., Sparse polynomial chaos expansions of frequency response functions using stochastic frequency transformation, Probab. Eng. Mech. 48 (2017) 39–58, https://doi.org/10.1016/j.probengmech.2017.04.003.
[33] G. Blatman, B. Sudret, Sparse polynomial chaos expansions of vector-valued response quantities, in: G. Deodatis, B.R. Ellingwood, D.M. Frangopol (Eds.), Safety, Reliability, Risk and Life-Cycle Performance of Structures and Infrastructures, CRC Press/Balkema, 2013, pp. 3245–3252.
[34] B. Bhattacharyya, E. Jacquelin, D. Brizard, Uncertainty quantification of stochastic impact dynamic oscillator using a proper orthogonal decomposition-polynomial chaos expansion technique, J. Vib. Acoust. 142 (6) (2020) 061013, https://doi.org/10.1115/1.4047359.
[35] E. Jacquelin, N. Baldanzini, B. Bhattacharyya, et al., Random dynamical system in time domain: a POD-PC model, Mech. Syst. Signal Process. 133 (2019) 106251, https://doi.org/10.1016/j.ymssp.2019.106251.
[36] L. Hawchar, C.-P. El Soueidy, F. Schoefs, Principal component analysis and polynomial chaos expansion for time-variant reliability problems, Reliab. Eng. Syst. Saf. 167 (2017) 406–416, https://doi.org/10.1016/j.ress.2017.06.024.
[37] J.B. Nagel, J. Rieckermann, B. Sudret, Principal component analysis and sparse polynomial chaos expansions for global sensitivity analysis and model calibration: application to urban drainage simulation, Reliab. Eng. Syst. Saf. 195 (2020) 106737, https://doi.org/10.1016/j.ress.2019.106737.
[38] K. Kontolati, D. Loukrezis, K.R. dos Santos, et al., Manifold learning-based polynomial chaos expansions for high-dimensional surrogate models, Int. J. Uncertain. Quantificat. 12 (4) (2022) 39–64, https://doi.org/10.1615/Int.J.UncertaintyQuantification.2022039936.
[39] D.G. Giovanis, D. Loukrezis, I.G. Kevrekidis, et al., Polynomial chaos expansions on principal geodesic Grassmannian submanifolds for surrogate modeling and uncertainty quantification, preprint, arXiv:2401.16683, 2024, https://doi.org/10.48550/arXiv.2401.16683.
[40] F. Gamboa, A. Janon, T. Klein, et al., Sensitivity analysis for multidimensional and functional outputs, Electron. J. Stat. 8 (1) (2014) 575–603, https://doi.org/10.1214/14-EJS895.
[41] I.M. Sobol, Sensitivity estimates for nonlinear mathematical models, Math. Model. Comput. Exp. 1 (4) (1993) 407–414, https://www.andreasaltelli.eu/file/repository/sobol1993.pdf.
[42] O. Garcia-Cabrejo, A. Valocchi, Global sensitivity analysis for multivariate output using polynomial chaos expansion, Reliab. Eng. Syst. Saf. 126 (2014) 25–36, https://doi.org/10.1016/j.ress.2014.01.005.

[43] X. Sun, Y.Y. Choi, J.-I. Choi, Global sensitivity analysis for multivariate outputs using polynomial chaos-based surrogate models, Appl. Math. Model. 82 (2020) 867–887, https://doi.org/10.1016/j.apm.2020.02.005.
[44] A. Galetzka, D. Loukrezis, N. Georg, et al., An hp-adaptive multi-element stochastic collocation method for surrogate modeling with information re-use, Int. J. Numer. Methods Eng. 124 (12) (2023) 2902–2930, https://doi.org/10.1002/nme.7234.
[45] A. Saltelli, M. Ratto, T. Andres, et al., Global Sensitivity Analysis: The Primer, John Wiley & Sons, 2008.
[46] I.M. Sobol, Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates, Math. Comput. Simul. 55 (1–3) (2001) 271–280, https://doi.org/10.1016/S0378-4754(00)00270-6.
[47] K. Campbell, M.D. McKay, B.J. Williams, Sensitivity analysis when model outputs are functions, Reliab. Eng. Syst. Saf. 91 (10–11) (2006) 1468–1472, https://doi.org/10.1016/j.ress.2005.11.049.
[48] M. Lamboni, D. Makowski, S. Lehuger, et al., Multivariate global sensitivity analysis for dynamic crop models, Field Crops Res. 113 (3) (2009) 312–320, https://doi.org/10.1016/j.fcr.2009.06.007.
[49] M. Lamboni, H. Monod, D. Makowski, Multivariate sensitivity analysis to measure global contribution of input factors in dynamic models, Reliab. Eng. Syst. Saf. 96 (4) (2011) 450–459, https://doi.org/10.1016/j.ress.2010.12.002.
[50] A.W. Van der Vaart, Asymptotic Statistics, vol. 3, Cambridge University Press, 1998.
[51] D. Xiu, G.E. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, SIAM J. Sci. Comput. 24 (2) (2002) 619–644, https://doi.org/10.1137/S1064827501387826.
[52] C. Soize, R. Ghanem, Physical systems with random uncertainties: chaos representations with arbitrary probability measure, SIAM J. Sci. Comput. 26 (2) (2004) 395–410, https://doi.org/10.1137/S1064827503424505.
[53] X. Wan, G.E. Karniadakis, Multi-element generalized polynomial chaos for arbitrary probability measures, SIAM J. Sci. Comput. 28 (3) (2006) 901–928, https://doi.org/10.1137/050627630.
[54] S. Oladyshkin, W. Nowak, Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion, Reliab. Eng. Syst. Saf. 106 (2012) 179–190, https://doi.org/10.1016/j.ress.2012.05.002.
[55] J. Feinberg, V.G. Eck, H.P. Langtangen, Multivariate polynomial chaos expansions with dependent variables, SIAM J. Sci. Comput. 40 (1) (2018) A199–A223, https://doi.org/10.1137/15M1020447.
[56] J.D. Jakeman, F. Franzelin, A. Narayan, et al., Polynomial chaos expansions for dependent random variables, Comput. Methods Appl. Mech. Eng. 351 (2019) 643–666, https://doi.org/10.1016/j.cma.2019.03.049.
[57] S. Rahman, A polynomial chaos expansion in dependent random variables, J. Math. Anal. Appl. 464 (1) (2018) 749–775, https://doi.org/10.1016/j.jmaa.2018.04.032.
[58] P.G. Constantine, M.S. Eldred, E.T. Phipps, Sparse pseudospectral approximation method, Comput. Methods Appl. Mech. Eng. 229–232 (2012) 1–12, https://doi.org/10.1016/j.cma.2012.03.019.
[59] P.R. Conrad, Y.M. Marzouk, Adaptive Smolyak pseudospectral approximations, SIAM J. Sci. Comput. 35 (6) (2013) A2643–A2670, https://doi.org/10.1137/120890715.
[60] J. Winokur, D. Kim, F. Bisetti, et al., Sparse pseudo spectral projection methods with directional adaptation for uncertainty quantification, J. Sci. Comput. 68 (2) (2016) 596–623, https://doi.org/10.1007/s10915-015-0153-x.
[61] G.T. Buzzard, Efficient basis change for sparse-grid interpolating polynomials with application to T-cell sensitivity analysis, Comput. Biol. J. (1) (2013) 562767, https://doi.org/10.1155/2013/562767.
[62] R. Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc., Ser. B, Methodol. 58 (1) (1996) 267–288.
[63] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (4) (2006) 1289–1306, https://doi.org/10.1109/TIT.2006.871582.
[64] M. Baudin, A. Dutfoy, B. Iooss, et al., OpenTURNS: an industrial software for uncertainty quantification in simulation, in: R. Ghanem, D. Higdon, H. Owhadi (Eds.), Handbook of Uncertainty Quantification, Springer International Publishing, 2016, pp. 1–38.
[65] S. Marelli, B. Sudret, UQLab: a framework for uncertainty quantification in Matlab, in: Vulnerability, Uncertainty, and Risk: Quantification, Mitigation, and Management, 2014, pp. 2554–2563.
[66] J. Pyrhonen, T. Jokinen, V. Hrabovcova, Design of Rotating Electrical Machines, John Wiley & Sons, 2013.
[67] P. Balakrishna, G. Book, W. Kirchgässner, et al., Gym-electric-motor (GEM): a Python toolbox for the simulation of electric drive systems, J. Open Sour. Softw. 6 (58) (2021) 2498, https://doi.org/10.21105/joss.02498.
[68] M. Gavish, D.L. Donoho, The optimal hard threshold for singular values is $4/\sqrt{3}$, IEEE Trans. Inf. Theory 60 (8) (2014) 5040–5053, https://doi.org/10.1109/TIT.2014.2323359.
[69] M. Gavish, D.L. Donoho, Optimal shrinkage of singular values, IEEE Trans. Inf. Theory 63 (4) (2017) 2137–2152, https://doi.org/10.1109/TIT.2017.2653801.
[70] A. Falini, A review on the selection criteria for the truncated SVD in data science applications, J. Comput. Math. Data Sci. 5 (2022) 100064, https://doi.org/10.1016/j.jcmds.2022.100064.
[71] S. Fliscounakis, P. Panciatici, F. Capitanescu, et al., Contingency ranking with respect to overloads in very large power systems taking into account uncertainty, preventive, and corrective actions, IEEE Trans. Power Syst. 28 (4) (2013) 4909–4917, https://doi.org/10.1109/TPWRS.2013.2251015.
[72] L. Thurner, A. Scheidler, F. Schäfer, et al., Pandapower—an open-source Python tool for convenient modeling, analysis, and optimization of electric power systems, IEEE Trans. Power Syst. 33 (6) (2018) 6510–6521, https://doi.org/10.1109/TPWRS.2018.2829021.
[73] F. Milano, Power System Modelling and Scripting, Springer-Verlag, Berlin Heidelberg, 2010.
[74] A.R. Jordehi, How to deal with uncertainties in electric power systems? A review, Renew. Sustain. Energy Rev. 96 (2018) 145–155, https://doi.org/10.1016/j.rser.2018.07.056.
[75] L. Novák, M. Vořechovskỳ, V. Sadílek, et al., Variance-based adaptive sequential sampling for polynomial chaos expansion, Comput. Methods Appl. Mech. Eng. 386 (2021) 114105, https://doi.org/10.1016/j.cma.2021.114105.
[76] L. Novák, H. Sharma, M.D. Shields, Physics-informed polynomial chaos expansions, J. Comput. Phys. 506 (2024) 112926, https://doi.org/10.1016/j.jcp.2024.112926.