



HAL
open science

Stochastic spectral embedding

S. Marelli, P.-R. Wagner, C. Lataniotis, B. Sudret

► **To cite this version:**

S. Marelli, P.-R. Wagner, C. Lataniotis, B. Sudret. Stochastic spectral embedding. 2020. hal-02539515v2

HAL Id: hal-02539515

<https://hal.science/hal-02539515v2>

Preprint submitted on 26 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

STOCHASTIC SPECTRAL EMBEDDING

S. Marelli, P.-R. Wagner, C. Lataniotis and B. Sudret



Data Sheet

Journal: -

Report Ref.: RSUQ-2020-003

Arxiv Ref.: <https://arxiv.org/abs/2004.04480> [stat.CO, stat.ML, stat.ME]

DOI: -

Date submitted: April 9th, 2020

Date accepted: -

Stochastic Spectral Embedding

S. Marelli¹, P.-R. Wagner¹, C. Lataniotis¹, and B. Sudret¹

¹*Chair of Risk, Safety and Uncertainty Quantification,*

ETH Zurich, Stefano-Franscini-Platz 5, 8093 Zurich, Switzerland

08.04.2020

Abstract

Constructing approximations that can accurately mimic the behavior of complex models at reduced computational costs is an important aspect of uncertainty quantification. Despite their flexibility and efficiency, classical surrogate models such as Kriging or polynomial chaos expansions tend to struggle with highly non-linear, localized or non-stationary computational models.

We hereby propose a novel sequential adaptive surrogate modeling method based on recursively embedding locally spectral expansions. It is achieved by means of disjoint recursive partitioning of the input domain, which consists in sequentially splitting the latter into smaller subdomains, and constructing a simpler local spectral expansions in each, exploiting the trade-off complexity vs. locality. The resulting expansion, which we refer to as “stochastic spectral embedding” (SSE), is a piece-wise continuous approximation of the model response that shows promising approximation capabilities, and good scaling with both the problem dimension and the size of the training set.

We finally show how the method compares favorably against state-of-the-art sparse polynomial chaos expansions on a set of models with different complexity and input dimension.

Keywords: surrogate modeling – spectral expansions – sparse regression – uncertainty quantification

1 Introduction

In the era of *machine learning* (ML) and *uncertainty quantification* (UQ), it is not surprising to see their boundary getting progressively blurred. Cross-fertilization between the two disci-

plines is nowadays the norm, rather than an exception, and for good reasons. Physics-informed neural networks are reaching unprecedented approximation power in UQ applications (see, *e.g.*, (Raissi et al., 2019; Pang et al., 2019)), while sparse polynomial chaos expansions are used as denoising regressors in Torre et al. (2019a), as high-dimensional regression tools in Lataniotis et al. (2020), and on real-world experimental data in Abbiati et al. (2021). UQ-born Gaussian process modeling Santner et al. (2003); Rasmussen and Williams (2006) is now a staple tool in ML Rasmussen and Williams (2006), while support vector machines (Vapnik, 2013) found their way in rare event estimation (Bourinet, 2016; Moustapha et al., 2018).

More in general, the adoption of both surrogate models and ML is becoming mainstream in applied sciences and engineering, with applications in entirely different fields. A few examples from the recent literature include: macroeconomics Harenberg et al. (2019), wind turbine modeling Slot et al. (2020), nuclear engineering Radaideh and Kozlowski (2020), smart grid engineering Wang et al. (2020), crash test simulations Moustapha et al. (2018), dam stability assessment Guo and Dias (2020), and many more. This list could be extended arbitrarily, as does the rich literature on these topics, but this task lies outside the scope of the current paper.

A common aspect across all of these works is the use of efficient and accurate functional approximation tools. Regardless of the specific technique, the general concept is straightforward: given a finite set of input realizations and their corresponding model responses, known as the *training set* (ML) or *experimental design* (UQ), a suitable parametric function is calibrated such that it accurately approximates the underlying (possibly unknown) input-output map. For the sake of consistency, and a little bias towards UQ, we will refer to this process as *surrogate modeling*, acknowledging that it is also known as *emulation*, *metamodeling*, *reduced order-* or *response surface-* modeling, or sometimes simply *regression*. A variety of methods is available in the surrogate modeling literature, which we cluster here in two classes:

- *Localized surrogates*: this includes interpolants (*e.g.* Gaussian process modeling Santner et al. (2003), spline interpolation Reinsch (1967), sparse grids (Bungartz and Griebel, 2004)), but also local regression methods (*e.g.* Gaussian process regression Rasmussen and Williams (2006), multivariate moving averages Lowry et al. (1992) or support vector machines (Vapnik, 2013)). These techniques rely on the availability of local information, *e.g.* through kernels on point-wise distance measures or support vectors, to provide predictions that are more accurate closer to the points in the training set. They therefore tend to perform better in interpolation, rather than extrapolation, tasks.
- *Global surrogates*: they provide global approximations without capitalizing on locally available information. Examples in this class include spectral methods (*e.g.* polynomial chaos expansions (PCE) Xiu and Karniadakis (2002); Blatman and Sudret (2011) and Pointcaré expansions Roustant et al. (2017)), linear regression methods (*e.g.* compressive sensing

Donoho et al. (2006); Lüthen et al. (2020), generalized linear models (Nelder and Wedderburn (1972)), artificial neural networks (ANNs) (Goodfellow et al. (2016)). These techniques tend to achieve better global accuracy (*e.g.* in terms of generalization error), thus offering some degree of extrapolation capabilities, but also worse local accuracy than their localized counterparts.

Each of the two classes have advantages and disadvantages, but they both tend to perform well on models that show homogeneous complexity throughout the input parameter space. Some models of practical engineering relevance, however, can show a highly localized behavior in different regions of the parameter space. Common examples include likelihood functions used in Bayesian inference (Nagel and Sudret (2016)), crash test simulations (Serna and Bucher (2009)), snap-through models (Hrinda (2010)), and discontinuous models in general.

Different approaches with varying degree of complexity have been proposed in the UQ and ML literature to address this kind of behavior. Examples include regression trees (Chipman et al., 2010; Breiman, 2017), multivariate adaptive regression splines (MARS, (Friedman, 1991)), various combinations of Kriging and PCE (PC-Kriging, Schöbi et al. (2015); Kersaudy et al. (2015)), multi-resolution/multi-element polynomial chaos expansions (Maître et al. (2004); Wan and Karniadakis (2006); Foo et al. (2008)) and deep neural networks (Goodfellow et al. (2016)), among others. Such methods can be broadly classified in two macro-families: *global approximations with local refinements* (*e.g.* PC-Kriging), or *domain-decomposition-based methods* (regression trees, MARS, multi-element polynomial chaos expansions). The class of global approximations with local refinements rely on efficiently combining global surrogates (*e.g.* spectral decompositions as polynomial chaos expansions, or global regression models) with local interpolation techniques (*e.g.* Gaussian processes or splines), to provide surrogates with acceptable extrapolation capabilities and good local accuracy. The class of domain-decomposition-based methods relies instead on the idea of partitioning the input parameter space into (often disjoint) subdomains, followed by the use of regression-based surrogates in each subdomain. This *divide-and-conquer* approach is particularly effective in reducing the complexity of the computational model in each subdomain, hence allowing relatively simple techniques to be used as local approximants. A prime example of this class of methods is given by regression trees (Chipman et al., 2010), where the local surrogates are as simple as constant values.

A common trait of most surrogate models used in a UQ context is that they rely on some form of regularity of the underlying computational model (*e.g.* smoothness or symmetry) to achieve an efficient representation based on an experimental design of relatively small size. It is therefore not surprising that they often show limited scalability with both the number of input dimensions (the well known *curse of dimensionality*) and with the size of the experimental design. Indeed, most local surrogates and interpolants rely on either kernel or clustering methods, neither

of which scales linearly with the number of dimensions. Moreover, their training requires the solution of complex optimization problems that often have at least as many parameters as input dimension Rasmussen and Williams (2006); Vapnik (2013). Global regression methods, on the other hand, require the optimization and storage of a large number of parameters or coefficients, which also rarely scales linearly in high dimension for non-trivial models.

To step further into scalability considerations, the number of available samples in the experimental design deserves some discussion. Historically, UQ-based surrogate modeling has taken a *parsimonious* approach: focus on small but informative experimental designs ($N_{ED} \approx 10^{1-2}$), because of the high computational costs associated to engineering models, and to their smooth behavior. On the other hand, ML has seen its expansion in the era of *big data*, focusing on large experimental designs ($N_{ED} \approx 10^{5-7}$), with often noisy data and highly non-smooth behavior. Albeit the gap is closing over time, a *no-man's land* in between the two still exists: computational models that show a complexity that is too high for classical surrogate modeling (*e.g.* extremely non-linear, or highly localized), but are expensive enough to only allow for $N_{ED} \approx 10^{3-4}$, regardless of the input dimension.

It is with this class of problems in mind that we propose a new surrogate modeling technique, namely stochastic spectral embedding (SSE), that combines global spectral representations and adaptive domain decompositions. We demonstrate that SSE can efficiently approximate models with varying degrees of complexity across the input space, while maintaining favorable scaling properties with both the input dimension and the size of the experimental design.

The paper is organized as follows: we first describe the general rationale and the details of the algorithm in Section 2. Then, in Section 3 we tackle the issue of constructing an SSE from an experimental design, in a regression context. In Section 4, we choose a reference spectral decomposition technique (polynomial chaos expansions, PCE) and we apply SSE to two highly complex analytical functions to showcase its capability to adapt to models with non-homogeneous complexity, and its scalability to high dimensions and large experimental designs. Finally, we also tackle two models of engineering complexity that are known to be challenging for classical surrogate modeling methods. We present concluding remarks in Section 5 and discuss extensions of the algorithm that could further improve its performance.

2 Stochastic spectral embedding: rationale and main algorithm

As the name suggests, stochastic spectral embedding (SSE) is a combination of two main ingredients: a *stochastic spectral representation*-based surrogate model and some form of *embedding*, which implies the sequential construction of subdomains of the full input space. In other words,

SSE consists in iteratively refining a spectral surrogate model by means of *embedding* additional surrogate models in subdomains of the parent expansion. In a sense, SSE can be seen as an extension of regression trees Breiman (2017); Chipman et al. (2010) to a much wider class of regression models, with the addition of a strong stochastic component due to the use of spectral representations.

2.1 Spectral expansions

We will consider herein the Hilbert space \mathcal{H} of random variables of the form $Y = \mathcal{M}(\mathbf{X})$ with finite second moments ($\mathbb{E}[Y^2] < \infty$), where \mathbf{X} is an M -dimensional random vector with joint distribution $\mathbf{X} \sim f_{\mathbf{X}}(\mathbf{x})$. Let the space be equipped with the inner product:

$$\langle g(\mathbf{X}), h(\mathbf{X}) \rangle_{\mathcal{H}} \stackrel{\text{def}}{=} \mathbb{E}[g(\mathbf{X})h(\mathbf{X})] = \int_{\mathcal{D}_{\mathbf{X}}} g(\mathbf{x})h(\mathbf{x})f_{\mathbf{X}}(\mathbf{x}) \, d\mathbf{x}, \quad (1)$$

where $\mathcal{D}_{\mathbf{X}} \subseteq \mathbb{R}^M$ is the support of \mathbf{X} . Then, every $Y \in \mathcal{H}$ admits a spectral representation \mathcal{M}_S of the form:

$$Y = \mathcal{M}_S(\mathbf{X}) \stackrel{\text{def}}{=} \sum_{j=1}^{\infty} a_j \Psi_j(\mathbf{X}), \quad (2)$$

where the $a_j \in \mathbb{R}$ are real coefficients, and the Ψ_j 's form a countably infinite orthonormal basis of the space:

$$\mathbb{E}[\Psi_i(\mathbf{X})\Psi_j(\mathbf{X})] = \langle \Psi_i(\mathbf{X}), \Psi_j(\mathbf{X}) \rangle_{\mathcal{H}} = \delta_{ij}, \quad (3)$$

where δ_{ij} is the Kronecker delta. For notational simplicity, the inner product subscript \mathcal{H} is omitted hereinafter.

Spectral decompositions of the form of Eq. (2) have a property that is particularly important for surrogate modelling, namely the fact that due to the orthogonality of the basis in Eq. (3), their (finite) second moment is given by:

$$\mathbb{E}[\mathcal{M}(\mathbf{X})^2] = \langle \mathcal{M}_S(\mathbf{X}), \mathcal{M}_S(\mathbf{X}) \rangle = \sum_{j=1}^{\infty} a_j^2 < +\infty. \quad (4)$$

The converging sum in Eq. (4) implies therefore that the coefficients a_j must decay at least geometrically when sorted by decreasing absolute value. This property is sometimes referred to as *compressibility*, because it essentially means that most of the information on the model variability is contained in a finite set of coefficients/basis elements. This allows one to truncate the spectral decomposition in Eq. (2) even if in principle it has an infinite number of terms. The truncated version of Eq. (2) is given by:

$$\mathcal{M}_S(\mathbf{X}) \approx \widehat{\mathcal{M}}_S(\mathbf{X}) = \sum_{j \in \mathcal{A}} a_j \Psi_j(\mathbf{X}), \quad (5)$$

where \mathcal{A} is a truncation set (typically related to the complexity of the basis functions, *e.g.*, maximum frequency in Fourier expansions, or maximum polynomial degree in PCE). The rapid

decay in the coefficients of spectral expansions is the main reason why many powerful surrogate modeling techniques that belong to the so-called class of *compressive sensing* (Donoho et al., 2006; Candès and Wakin, 2008), have proven to be very effective in various recent applications (Blatman and Sudret, 2011; Torre et al., 2019a; Lüthen et al., 2020). Compressive sensing uses sparse regression tools to identify the best truncation set \mathcal{A} based on the available information in the experimental design.

Because of the truncation introduced in Eq. (5), the expansion is in general not exact, hence we define the *residual* $\mathcal{R}(\mathbf{X})$ as:

$$\mathcal{R}(\mathbf{X}) = \mathcal{M}(\mathbf{X}) - \widehat{\mathcal{M}}_S(\mathbf{X}). \quad (6)$$

Due to the convergent behavior of the truncated expansion, it follows that $\text{Var}[\mathcal{R}(\mathbf{X})] \ll \text{Var}[\widehat{\mathcal{M}}_S(\mathbf{X})]$. By definition, spectral expansions belong to the class of global representations, i.e. the basis functions in Eq. (5) have support on the entire domain $\mathcal{D}_{\mathbf{X}}$. Therefore, highly localized models, or those with inhomogeneous behavior throughout the input domain tend to require an extremely large number of terms in the truncated expansion to achieve satisfactory approximation accuracy (Gibbs phenomenon). As an example, the number of terms in the well-established polynomial chaos expansion (Xiu and Karniadakis, 2002; Blatman and Sudret, 2011) can grow very fast when the underlying model has strongly localized behavior, because a high polynomial degree is required for an accurate representation.

2.2 A sequential partitioning approach

To alleviate this limitation, while still capitalizing on the powerful convergence properties of spectral methods, SSE constructs a sequence of spectral expansions of manageable complexity on increasingly smaller subdomains of the original domain. Such subdomains are denoted $\mathcal{D}_{\mathbf{X}}^{\ell,p} \subseteq \mathcal{D}_{\mathbf{X}}$, where ℓ is the expansion level and p is a subdomain index within the level. The expansion is performed only on the local residual from the previous level.

For illustration purposes, Figure 1 shows an example of sequential partitioning for a simple 2D bounded domain, obtained by splitting each subdomain in two equal subdomains across a random dimension. When $\ell = 0$, there is only a single subdomain $\mathcal{D}_{\mathbf{X}}^{0,1} \stackrel{\text{def}}{=} \mathcal{D}_{\mathbf{X}}$ and the residual is $\widehat{\mathcal{R}}_S^{0,1}(\mathbf{X}) = \widehat{\mathcal{M}}_S(\mathbf{X})$ from Eq. (6).

While the idea of partitioning the input space in smaller subdomains and constructing a surrogate model in each is certainly not new in UQ (see, e.g. Maître et al. (2004); Wan and Karniadakis (2006)), the use of a sequence of residuals in SSE sets it apart from other divide and conquer methods.

In more formal terms, SSE in its basic form can be written as a multi-level expansion of the

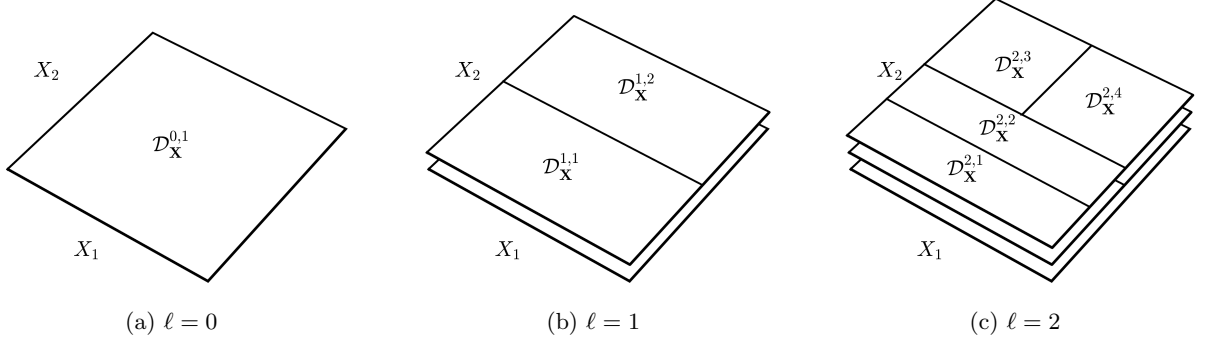


Figure 1: Example of a possible partitioning sequence for an SSE with $L = 2$ and $P_\ell = 2^\ell$, and bounded uniform marginal distributions. In this simple example, each subdomain is split in two equal parts across a random direction.

form:

$$\mathcal{M}_{\text{SSE}}(\mathbf{X}) = \sum_{\ell=0}^L \sum_{p=1}^{P_\ell} \mathbf{1}_{\mathcal{D}_X^{\ell,p}}(\mathbf{X}) \widehat{\mathcal{R}}_S^{\ell,p}(\mathbf{X}), \quad (7)$$

where L is the total number of expansion levels considered, P_ℓ is the number of subdomains at level $\ell \in \{1, \dots, L\}$, $\mathbf{1}_{\mathcal{D}_X^{\ell,p}}(\mathbf{X})$ is the indicator function of the subdomain $\mathcal{D}_X^{\ell,p}$. Finally, $\widehat{\mathcal{R}}_S^{\ell,p}(\mathbf{X})$ is the truncated expansion of the residual of the SSE up to level $\ell - 1$, $\mathcal{R}^\ell(\mathbf{X})$,

$$\mathcal{R}^\ell(\mathbf{X}) = \mathcal{M}(\mathbf{X}) - \sum_{k=0}^{\ell-1} \sum_{p=1}^{P_k} \mathbf{1}_{\mathcal{D}_X^{k,p}}(\mathbf{X}) \widehat{\mathcal{R}}_S^{k,p}(\mathbf{X}). \quad (8)$$

In the above equations, each residual term is expanded onto a local orthonormal basis as follows:

$$\widehat{\mathcal{R}}_S^{k,p}(\mathbf{X}) = \sum_{j \in \mathcal{A}^{k,p}} a_j^{k,p} \Psi_j^{k,p}(\mathbf{X}). \quad (9)$$

A local inner product is defined in the domain $\mathcal{D}_X^{k,p}$:

$$\left\langle \Psi_i^{k,p}(\mathbf{X}), \Psi_j^{k,p}(\mathbf{X}) \right\rangle_{k,p} = \int_{\mathcal{D}_X^{k,p}} \Psi_i^{k,p}(\mathbf{x}) \Psi_j^{k,p}(\mathbf{x}) f_{\mathbf{X}}^{k,p}(\mathbf{x}) d\mathbf{x}, \quad (10)$$

where:

$$f_{\mathbf{X}}^{k,p}(\mathbf{x}) = \mathbf{1}_{\mathcal{D}_X^{k,p}}(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{\mathcal{V}^{k,p}} \quad (11)$$

is the joint PDF of the input parameters restricted to the subdomain $\mathcal{D}_X^{k,p}$ and rescaled by its probability mass $\mathcal{V}^{k,p}$:

$$\mathcal{V}^{k,p} = \int_{\mathcal{D}_X^{k,p}} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}. \quad (12)$$

A crucial aspect of SSE is that by partitioning the entire input domain \mathcal{D}_X into smaller subdomains $\mathcal{D}_X^{\ell,p}$, it trades the complexity of the single, often global expansion in Eq. (5) for a

(possibly large) number of local expansions with much smaller truncation sets. In cases where the spectral basis is continuous in Eq. (5), SSE results in a final piecewise continuous approximation, but no continuity is ensured on the boundaries of the subdomains. Mean-square convergence of the procedure is guaranteed by the spectral convergence in each level, which implies that the residual local variance in each subdomain is in expectation decreasing rapidly. In other words, for each increasing level ℓ in Eq. (7), new discontinuity bounds are generated during the partitioning step, but the variance of the overall residual is reduced, thus resulting, in expectation, in lower amplitude discontinuities. This behavior is analogous to that of regression trees Friedman (1991); Breiman (2017).

2.3 The SSE algorithm

Algorithmically, SSE consists of a local refinement sequence of a global spectral expansion into sequentially smaller subdomains $\mathcal{D}_{\mathbf{X}}^{\ell,p}$. For notational simplicity, we introduce here a set of local random vectors distributed according to the local PDF in Eq. (11): $\mathbf{X}^{\ell,p} \sim f_{\mathbf{X}}^{\ell,p}(\mathbf{x})$. We further choose a certain *partitioning strategy* that is discussed in Section 3.2.

Then, the SSE algorithm can be written as:

1. Initialization:

- (a) $\ell = 0, p = 1$
- (b) $\mathcal{D}_{\mathbf{X}}^{\ell,p} = \mathcal{D}_{\mathbf{X}}$
- (c) $\mathcal{R}^{\ell}(\mathbf{X}) = \mathcal{M}(\mathbf{X})$

2. For each subdomain $\mathcal{D}_{\mathbf{X}}^{\ell,p}, p = 1, \dots, P_{\ell}$:

- (a) Calculate the truncated expansion $\widehat{\mathcal{R}}_S^{\ell,p}(\mathbf{X}^{\ell,p})$ of the residual $\mathcal{R}^{\ell}(\mathbf{X}^{\ell,p})$ in the current subdomain
- (b) Update the residual in the current subdomain $\mathcal{R}^{\ell+1}(\mathbf{X}^{\ell,p}) = \mathcal{R}^{\ell}(\mathbf{X}^{\ell,p}) - \widehat{\mathcal{R}}_S^{\ell,p}(\mathbf{X}^{\ell,p})$
- (c) Split the current subdomain $\mathcal{D}_{\mathbf{X}}^{\ell,p}$ in N_S subdomains $\mathcal{D}_{\mathbf{X}}^{\ell+1, \{s_1, \dots, s_{N_S}\}}$ based on a partitioning strategy
- (d) If $\ell < L$, $\ell \leftarrow \ell + 1$, go back to 2a, otherwise terminate the algorithm

3. Termination

- (a) Return the full sequence of $\mathcal{D}_{\mathbf{X}}^{\ell,p}$ and $\widehat{\mathcal{R}}_S^{\ell,p}(\mathbf{X}^{\ell,p})$ needed to compute Eq. (7).

Note that in steps 2a and 2b of the previous algorithm the residual $\mathcal{R}^{\ell}(\mathbf{X}^{\ell,p})$ is only indexed by ℓ , but not by the subdomain index p . This is because the residual is fully defined with respect to the previous level ℓ , which is independent on the particular subdomain under consideration (see Figure 1).

3 Building a stochastic spectral embedding from data

For it be useful in practical applications, SSE needs to be “trainable” from a finite-size experimental design. Hereinafter, we consider an experimental design $\mathcal{X} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$ and its corresponding model evaluations $\mathcal{Y} = \{y^{(1)}, \dots, y^{(N)}\}$ as the only data available for training.

Upon closer inspection of the algorithm in Section 2.3, the training phase of the SSE representation consists in estimating the following quantities from the available experimental design:

1. The expansion coefficients of the local residual in each level and subdomain $\mathbf{a}^{\ell,p}$ (Eq. (9)).
2. A partitioning strategy at each level.
3. The total number of splitting levels, L .

In the following sections we introduce a comprehensive adaptive strategy based on sparse linear regression to perform each of these steps from a given experimental design.

3.1 Calculating the residual expansion coefficients

For a specific subdomain $\mathcal{D}_{\mathbf{X}}^{\ell,p}$, a local spectral expansion of the residual \mathcal{R}_S^ℓ needs to be constructed from the available experimental design. We therefore define a so-called *local experimental design* $\mathcal{X}^{\ell,p} \subseteq \mathcal{X}$, the subset of the original experimental design lying within the subdomain $\mathcal{D}_{\mathbf{X}}^{\ell,p}$:

$$\mathcal{X}^{\ell,p} \stackrel{\text{def}}{=} \left\{ \mathbf{x}^{(j)}, j = 1, \dots, N^{\ell,p}, \text{ such that } \mathbf{x}^{(j)} \in \left(\mathcal{X} \cap \mathcal{D}_{\mathbf{X}}^{\ell,p} \right) \right\}. \quad (13)$$

A similar notation is used to identify the corresponding model responses, $\mathcal{Y}^{\ell,p}$. Using the auxiliary local random vector $\mathbf{X}^{\ell,p}$ introduced in the previous section, the residual expansion in Eq. (9) reads:

$$\widehat{\mathcal{R}}_S^{\ell,p}(\mathbf{X}^{\ell,p}) = \sum_{j \in \mathcal{A}^{\ell,p}} a_j^{\ell,p} \Psi_j^{\ell,p}(\mathbf{X}^{\ell,p}). \quad (14)$$

Given the local experimental design $\mathcal{X}^{\ell,p}$ and a truncated local spectral basis $\Psi_j^{\ell,p}$, $j \in \mathcal{A}^{\ell,p}$, the task of identifying the coefficients $\mathbf{a}^{\ell,p} \stackrel{\text{def}}{=} \{a_j^{\ell,p}, j \in \mathcal{A}^{\ell,p}\}$ can then be cast as a linear regression problem (see, *e.g.*, Berveiller et al. (2006)):

$$\mathbf{a}^{\ell,p} \approx \widehat{\mathbf{a}}^{\ell,p} = \arg \min_{\mathbf{a}} \sum_{\mathbf{x}^{(i)} \in \mathcal{X}^{\ell,p}} \left(\mathcal{R}^\ell(\mathbf{x}^{(i)}) - \sum_{j \in \mathcal{A}^{\ell,p}} a_j^{\ell,p} \Psi_j^{\ell,p}(\mathbf{x}^{(i)}) \right)^2. \quad (15)$$

While in principle the regression problem in Eq. (15) can be solved through ordinary least squares, recent literature on the topic of compressive sensing has amply demonstrated that sparse regression approaches can provide great benefits in terms of accuracy, especially for relatively small experimental designs Donoho et al. (2006); Blatman and Sudret (2011); Lüthen et al. (2020). A review of the available techniques for this purpose lies outside the scope of this paper and is extensively explored for one popular class of spectral representations (polynomial chaos expansions) in Lüthen et al. (2020).

3.2 Partitioning strategy

A second step necessary to construct SSE from data is to identify a proper partitioning strategy between levels. Any strategy for the partitioning of the input domain $\mathcal{D}_{\mathbf{X}}$ can be employed for Eq. (7), under the sole condition that at each level ℓ :

$$\bigcup_{p=1}^{P_\ell} \mathcal{D}_{\mathbf{X}}^{\ell,p} = \mathcal{D}_{\mathbf{X}}. \quad (16)$$

While a comprehensive study on different partitioning strategies would be interesting, for the sake of simplicity we adopt hereinafter a rather simple approach, similar in spirit to regression trees Breiman (2017). In other words, we split every subdomain in two parts of equal probability mass along one of the input directions, $d^{\ell,p} \in \{1, \dots, M\}$. Note that the direction in itself can be different for each subdomain, even on the same level.

Under very general conditions, it is possible to bijectively map any random vector \mathbf{X} with joint distribution $F_{\mathbf{X}}$, to the uniform independent random vector $\mathbf{U} \sim \mathcal{U}(0,1)^M$ through an appropriate isoprobabilistic transform (*e.g.*, the Rosenblatt transform Rosenblatt (1952); Torre et al. (2019b)):

$$\begin{aligned} \mathbf{U} &= g(\mathbf{X}) \\ \mathbf{X} &= g^{-1}(\mathbf{U}), \end{aligned} \quad (17)$$

where $g(\cdot)$ denotes the isoprobabilistic transform. This mapping simplifies the proposed partitioning strategy: splitting is performed in the uniformly distributed *quantile space* \mathbf{U} , and the resulting split domains $\mathcal{D}_{\mathbf{U}}^{\ell,p}$ are mapped back to the input space \mathbf{X} via the inverse transform (see Eq. (17)). This has several computational benefits, including proper treatment of unbounded variables. Figure 2 shows graphically a two-dimensional example of partitioning in the quantile (uniform) space \mathbf{U} , and its corresponding mapping to unbounded random variables in the physical space \mathbf{X} .

In the general case, a strategy is needed to choose a specific splitting direction $d^{\ell,p} \in \{1, \dots, M\}$ for each existing subdomain $\mathcal{D}_{\mathbf{X}}^{\ell,p}$. Different heuristic reasoning can be used to make this choice, including purely random splitting (as in Figure 2), using the direction of maximum residual difference, or estimates of the variability of the $\widehat{\mathcal{R}}_S^{\ell,p}$ in each direction (following the same rationale as in Shields (2018)). The optimal criterion can be application-specific, because it may in principle depend on the chosen spectral representation.

3.3 Sparse tree representation and expansion truncation

In a regression context, it is difficult to choose *a priori* a truncation on the maximum number of levels L in the expansion in Eq. (7). Because the samples may be unequally distributed, some subdomains at each level may be empty, or more formally $\mathcal{X}^{\ell,p} = \emptyset$ for some combinations of

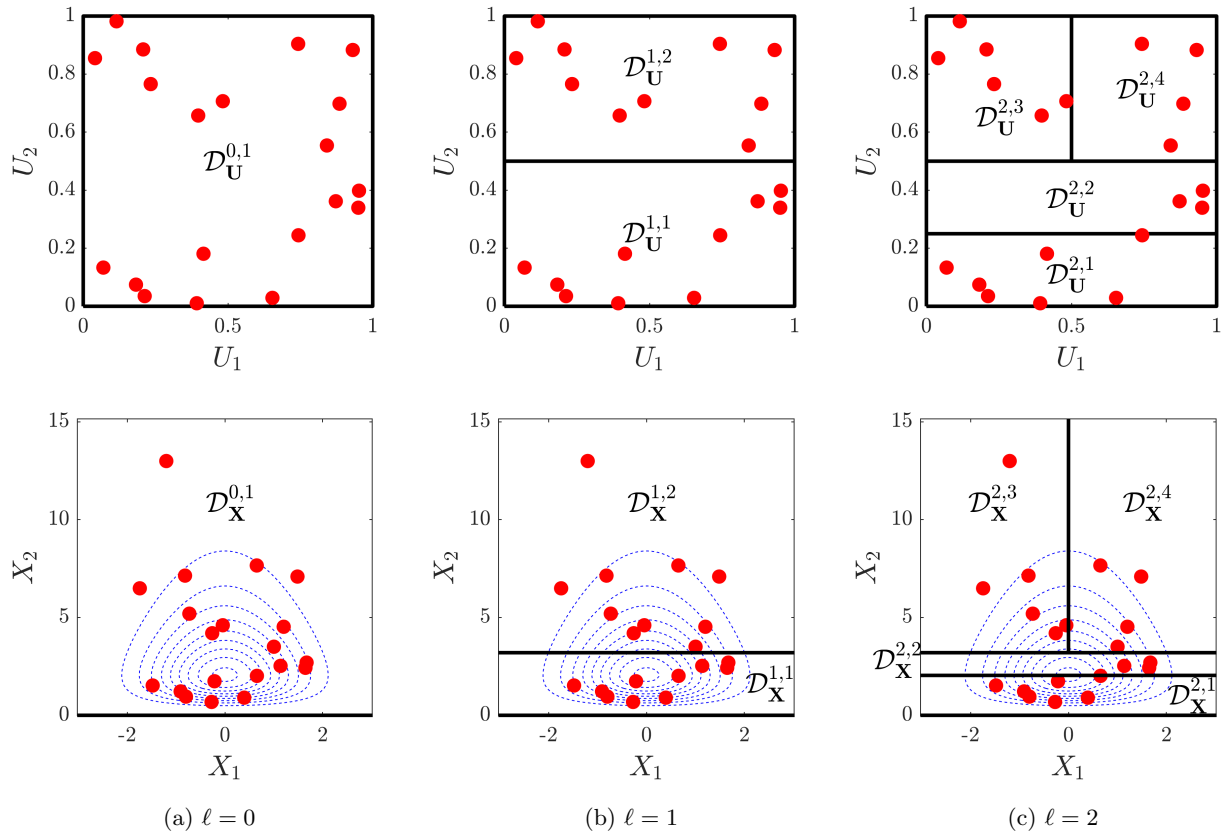


Figure 2: Graphical representation of the partitioning strategy described in section 3.2 for a two-dimensional problem with independent random variables. Upper row: partitioning in the quantile space; Lower row: partitioning in the original space. Red dots show a random sampling from the original distributions in both spaces, and serve as a visual aid to recognize the mapping between the two probability spaces from Eq. (17). The splitting direction in each subdomain is determined randomly in this example.

ℓ and p . We therefore take a straightforward approach to obviate this issue, by initializing the residual at every level and subdomain to the null function, hence $\mathbf{a}^{\ell,p} = \mathbf{0}$. The coefficients are then updated only in the subdomains that satisfy $|\mathcal{X}^{\ell,p}| \geq N_{\min}$, where N_{\min} is a parameter of the SSE algorithm that represents the minimum number of points in a subdomain required to justify an expansion. The SSE expansion is truncated when no new updates are possible given the current experimental design \mathcal{X} , or more formally:

$$L = \min \{ \ell : |\mathcal{X}^{\ell,p}| < N_{\min}, \forall p \in \{1, \dots, P_\ell\} \} - 1. \quad (18)$$

In addition to providing a suitable stopping criterion for the algorithm in Section 2.3, an added benefit of this strategy is that only the residual expansions that were effectively updated need to be stored in memory. This provides a degree of sparsity in the representation and potentially significantly reduces the memory fingerprint of the method, especially in the case of a large number of points in the experimental design.

Note that, for an experimental design of size N and a minimum number of points per expansion N_{\min} , the following holds:

$$2^{\bar{L}} N_{\min} \leq N, \quad (19)$$

where \bar{L} is the expected value of the maximum L in (18) and therefore

$$\bar{L} \leq \left\lceil \log_2 \frac{N}{N_{\min}} \right\rceil. \quad (20)$$

3.4 Error estimation

In the context of surrogate modeling, assessing the accuracy of the approximation is an important task. Arguably the best known accuracy estimator in function approximation is the so-called generalization error E_{GEN} , which for SSE is given by

$$E_{\text{GEN}} \stackrel{\text{def}}{=} \mathbb{E} [(\mathcal{M}(\mathbf{X}) - \mathcal{M}_{\text{SSE}}(\mathbf{X}))^2]. \quad (21)$$

A direct estimation of this quantity is in general impossible, as it would require the availability of an extensive validation set. Instead, because we adopt a regression approach to calibrate the spectral decompositions in each subdomain, we estimate the generalization error through leave-one-out cross-validation (Chapelle et al., 2002; Blatman and Sudret, 2010) that is available for each of the local expansions.

For notational convenience, we introduce here the set of *terminal domains* $\mathcal{D}_{\mathcal{T}} = \{ \mathcal{D}_{\mathbf{X}}^{L,1}, \dots, \mathcal{D}_{\mathbf{X}}^{L,P_L} \}$, *i.e.* those domains that belong to the last expansion level L in Eq. (7). By definition $\mathcal{D}_{\mathcal{T}}$ is a complete partition of the input domain $\mathcal{D}_{\mathbf{X}}$. Because of the sequential nature of SSE, which locally refines the previous approximation level with the expansion of the residual in the current subdomain, an accurate estimate of the local generalization error in each of the terminal domains

would then suffice to provide an estimate of the overall E_{GEN} of the full SSE. If we denote the local residual error:

$$E_{\text{GEN}}^{\ell,p} = \mathbb{E} \left[\left(\mathcal{R}^\ell(\mathbf{X}^{\ell,p}) - \widehat{\mathcal{R}}_S^{\ell,p}(\mathbf{X}^{\ell,p}) \right)^2 \right], \quad (22)$$

then the global generalization error is simply given by the average error in each terminal domain:

$$E_{\text{GEN}} = \mathbb{E} \left[E_{\text{GEN}}^{L,p} \right] = \sum_{p=1}^{P_L} E_{\text{GEN}}^{L,p} \cdot \mathcal{V}^{L,p}, \quad (23)$$

which is equal to the sum of the terminal domain errors weighted by the corresponding probability mass in Eq. (12).

To provide an estimator based on the available experimental design \mathcal{X} , we only need an estimator of $E_{\text{GEN}}^{L,p}$. Arguably the most common tool for the estimation of generalization error in regression problems is k -fold cross-validation (see, *e.g.*, Vapnik (2013)). The special case of $k = N$ is also known as *leave-one-out* error and marked E_{LOO} . For ordinary least square regression it can be calculated analytically from the expansion coefficients and the basis functions (Chapelle et al., 2002; Blatman and Sudret, 2010).

Given the sparse tree representation described in Section 3.3, it cannot be guaranteed that the residual \mathcal{R}^ℓ is expanded in every terminal domain. Therefore, during the splitting phase of the SSE algorithm (Step 2c of the algorithm in Section 2.3) we initialize the error of all the subdomains $\mathcal{D}^{\ell+1, \{s_1, \dots, s_{N_S}\}}$ of the current subdomain $\mathcal{D}^{\ell,p}$ to the leave-one-out error of the latter $E_{\text{LOO}}^{\ell,p}$:

$$E_{\text{LOO}}^{\ell+1, \{s_1, \dots, s_{N_S}\}} = E_{\text{LOO}}^{\ell,p}. \quad (24)$$

Then, we update the error estimate in each subdomain during Step 2a only if the conditions for its expansion hold (see Section 3.3). As a result, every terminal domain is either assigned its own leave-one-out error if it contains a residual expansion, or inherits the leave-one-out error from the last ancestor domain that was expanded.

By using the leave-one-out error in each terminal domain as an estimator of its generalization error $\widehat{E}_{\text{GEN}}^{L,p} = \widehat{E}_{\text{LOO}}^{L,p}$, the empirical estimator of Eq. (23) reads:

$$\widehat{E}_{\text{GEN}} = \sum_{p=1}^{P_L} \widehat{E}_{\text{LOO}}^{L,p} \cdot \mathcal{V}^{L,p}. \quad (25)$$

In most metamodeling applications, it is customary to normalize the estimated error by the variance of the experimental design, to obtain a dimensionless error measure. The relative error is thus defined as:

$$\widehat{\epsilon}_{\text{GEN}} = \frac{1}{\text{Var}[\mathcal{Y}]} \sum_{p=1}^{P_L} \widehat{E}_{\text{LOO}}^{L,p} \cdot \mathcal{V}^{L,p}. \quad (26)$$

4 Applications

In this section we aim at showing the performance of SSE on a set of applications that can prove challenging for standard metamodeling techniques. Because of its widespread use in the uncertainty quantification of engineering models, we choose as a spectral decomposition technique polynomial chaos expansions Xiu and Karniadakis (2002); Blatman and Sudret (2011) (hereinafter PCE). This choice is also quite convenient due to several specific properties of PCE, that combine well with SSE.

4.1 Synergies with polynomial chaos expansions

By using the same notation as in Section 2, and assuming that \mathbf{X} has independent components, the truncated polynomial chaos expansion of a finite variance model can be written as Le Gratiet et al. (2016):

$$\mathcal{M}_{\text{PCE}}(\mathbf{X}) = \sum_{\alpha \in \mathcal{A}} a_{\alpha} \Psi_{\alpha}(\mathbf{X}), \quad (27)$$

where α is a multi-index that identifies the polynomial degree in each variable, \mathcal{A} is a suitable truncation set (*e.g.* $\mathcal{A} = \mathcal{A}^{M,d}$ containing all multivariate polynomials with degree $\leq d$), and the $\Psi_{\alpha}(\mathbf{X})$ form an orthogonal basis of multivariate polynomials. The latter can be obtained via tensor product of univariate polynomials as follows:

$$\Psi_{\alpha}(\mathbf{X}) = \prod_{i=1}^M \Phi_{\alpha_i}^{(i)}(\mathbf{X}_i), \quad (28)$$

where $\Phi_{\alpha_i}^{(i)}$ is a polynomial of degree α_i that belongs to the family of univariate polynomials orthogonal with respect to the input PDF of $X_i \sim f_{X_i}(x_i)$ and the inner product in Eq. (1).

An interesting property of the univariate polynomials that synergizes well with our proposed SSE, is that it is possible to construct polynomials orthogonal to almost any input PDF through Gram-Schmidt orthogonalization (for an extensive review, see Gautschi (2004); Ernst et al. (2012)). In the context of SSE, this property has a powerful implication: in each subdomain $\mathcal{D}^{\ell,p}$ the basis elements $\Psi_{\alpha}^{\ell,p}(\mathbf{X}^{\ell,p})$ in Eq. (9) are still polynomial functions of the original input variables \mathbf{X} .

This property, together with the analytical integrability of polynomials, allows us to derive several statistics of interest of PCE-based SSE analytically. Let us first introduce the notion of *flattened representation*: because SSE is a polynomial in the original variables in every level and subdomain, this also holds for the terminal domains introduced in Section 3.4. Therefore, one can project the full SSE in Eq. (7) as a local PCE onto each terminal domain:

$$\mathcal{M}_{\text{SSE}}^F(\mathbf{X}^{L,p}) = \sum_{\alpha \in \mathcal{A}^{\tau}} c_{\alpha}^p \Psi_{\alpha}^{L,p}(\mathbf{X}^{L,p}), \quad (29)$$

where \mathcal{A}^T is a suitable truncation set for the projection to be exact, and the c_{α}^p are the corresponding coefficients. The latter can easily be computed either analytically or exactly through quadrature. Note that, while the basis elements in the PCE in Eq. (29) correspond to the $\Psi_j^{k,P}$ in Eq. (9) (they only depend on the input PDF in Eq. (11)), in general the coefficients will not be the same, *i.e.* $c_{\alpha}^p \neq a_{\alpha}^{L,p}$ in Eq. (9).

Because PCE contains as a basis element the constant term, it is straightforward to demonstrate that the expected value of Eq. (7) reads:

$$\mathbb{E}[\mathcal{M}_{\text{SSE}}(\mathbf{X})] = \sum_{p=1}^{P_L} c_{\mathbf{0}}^p \nu^{L,p}, \quad (30)$$

which is the weighted mean of all the mean values of the flattened representation in Eq. (29).

Similarly, the variance can be calculated as:

$$\text{Var}[\mathcal{M}_{\text{SSE}}(\mathbf{X})] = \left(\sum_{p=1}^{P_L} \nu^{L,p} \sum_{\alpha \in \mathcal{A}^T} (c_{\alpha}^p)^2 \right) - \mathbb{E}[\mathcal{M}_{\text{SSE}}(\mathbf{X})]^2. \quad (31)$$

A number of other quantities of engineering interest (*e.g.* conditional variances, Sobol' sensitivity indices, etc.) can be derived similarly from the flattened representation. A selection of those is reported in A.

From a technical perspective, the flattened representation in Eq. (29) contains all the information needed to evaluate Eq. (7) on a new point, but at a much lower storage cost, as only the final sets of coefficients c_{α}^p and basis indices \mathcal{A}^T need to be stored. This has additional advantages during the prediction of the response on new points, because it only requires the prediction of a single local expansion in the appropriate terminal domains, rather than that of all of its ancestors as in the original formulation in Eq. (7). More formally, for a point $\mathbf{x}_0 \in \mathcal{D}_{\mathbf{X}}$ it is sufficient to find $p_0 \in 1, \dots, P_L$ for which $\mathbf{x}_0 \in \mathcal{D}_{\mathbf{X}}^{L,p_0}$ and evaluate the flattened SSE from Eq. (29) for $p = p_0$.

4.2 Example applications and testing strategy

To compare the performance of SSE over sparse PCE, we choose four reference problems of increasing complexity: (i) a one-dimensional analytical function with localized non-polynomial behavior, (ii) a 100-dimensional analytical function with decreasing parametric importance in higher dimensions, (iii) an 8-dimensional engineering model describing the performance function of a damped oscillator and (iv) a three-dimensional discontinuous engineering model describing the snap-trough behavior of a truss structure.

Among the ingredients identified in Section 2.3 is a partitioning strategy, to choose the splitting direction in every subdomain. After extensive testing, we found that splitting according to the direction of highest variability of $\widehat{\mathcal{R}}_S^{\ell,p}$ proved to be the most effective, especially for smaller

experimental designs. We therefore split each subdomain $\mathcal{D}_X^{\ell,p}$ into two subdomains with equal probability mass, *i.e.* $N_S = 2$, along the direction that has the maximum first order Sobol' indexSobol' (1993), as analytically derived from the coefficients of $\widehat{\mathcal{R}}_S^{\ell,p}$ (Sudret, 2008).

In all applications we compare the convergence behavior of SSE vs. its spectral counterpart PCE as a function of the experimental design (ED) sizes N_{ED} . To assess the robustness of the results, we consider 10 independent replications of each ED, and provide the results in Tukey box-plots. For each experimental design size, SSE construction is terminated for $N_{\min} = \min\{5M, 50\}$ (see Section 3.3).

As a spectral technique, we adopt the adaptive sparse-PCE based on LARS approach developed in Blatman and Sudret (2011) in its numerical implementation in UQLab (Marelli and Sudret, 2014, 2019). Each $\widehat{\mathcal{R}}_S^{\ell,p}$ is therefore a degree- and q -norm-adaptive polynomial chaos expansion. We further introduce a rank truncation of $r = 2$ to cope with high dimensional problems, *i.e.* we limit the maximum number of input interactions (Marelli and Sudret, 2019) to 2 variables at a time. The truncation set for each spectral expansion (Eq. (27)) thus reads:

$$\mathcal{A}^{M,p,q,r} = \{\boldsymbol{\alpha} \in \mathbb{N}^M : \|\boldsymbol{\alpha}\|_q \leq p, \|\boldsymbol{\alpha}\|_0 \leq r\}, \quad (32)$$

where

$$\|\boldsymbol{\alpha}\|_q = \left(\sum_{i=1}^M \alpha_i^q \right)^{\frac{1}{q}}, \quad q \in (0, 1]; \quad \|\boldsymbol{\alpha}\|_0 = \sum_{i=1}^M 1_{\{\alpha_i > 0\}}. \quad (33)$$

The q -norm is adaptively increased between $q = \{0.5, \dots, 0.8\}$ while the maximum polynomial degree is adaptively increased in the interval $p = \{0, 1, \dots, p_{\max}^{\text{SSE}}\}$, where the maximum degree p_{\max}^{SSE} is a parameter for each case study.

In all examples the SSE performance is compared to standard polynomial chaos expansions on the same ED. These PCEs are constructed with the same adaptive approach used for the SSE expansions. Their maximum degree, however, is denoted by p_{\max}^{PCE} . It is set to the highest value our computational budget admitted for a given dimensionality but at least to $p_{\max}^{\text{PCE}} > 2 \cdot p_{\max}^{\text{SSE}}$.

We compare the performance of SSE and PCE in terms of the relative mean squared error (MSE) η , a well known error estimator defined as

$$\eta \stackrel{\text{def}}{=} \frac{\mathbb{E} \left[\left(\mathcal{M}(X) - \tilde{\mathcal{M}}(X) \right)^2 \right]}{\text{Var} [\mathcal{M}(X)]} \quad (34)$$

where $\tilde{\mathcal{M}}$ is either the PCE or SSE surrogate model. This error measure was estimated with standard Monte Carlo simulations using a large sample of size $N = 10^6$.

4.3 Application 1: one-dimensional analytical function

We first present a simple one-dimensional example that is meant to illustrate how SSE behaves with a model on which PCE is expected to fail. The model is given by:

$$\mathcal{M}(X) = -X + 0.1 \sin(30X) + \exp(-(50(X - 0.65))^2), \quad (35)$$

where $X \sim \mathcal{U}(0, 1)$ is a uniformly distributed random variable. The first two terms in Eq. (35) (polynomial and sinusoidal) can be accurately approximated by a low degree PCE, while the third term (squared exponential) causes PCE to require extremely high degree due to the localized peak it introduces at $x = 0.65$ (see Figure 3). In the same Figure we detail four SSE refinement stages, with $N_{ED} = 200$ and $p_{\max}^{\text{SSE}} = 5$. For every step we show on the top panel a graphical representation of the various subdomains identified by the algorithm, with the subdomains of level ℓ highlighted in orange. In the middle panel we plot the true model (orange solid line) and the current SSE approximation \mathcal{M}_{SSE} as a dashed blue line. In the bottom panel, we plot the corresponding residual $\mathcal{M}(X) - \mathcal{M}_{\text{SSE}}(X)$ as a solid blue line in the same vertical scale as in the middle panel, for comparison.

In the first step in Figure a the main trend of the function is identified, leaving a residual that mainly consists of the sine oscillation and the exponential peak. In the following step (Figure b) the approximation is not greatly improved in the subdomain $\mathcal{D}_X^{1,1} : [0, 0.5]$, because the available maximum degree p_{\max}^{SSE} is not sufficiently high, resulting in a mostly constant polynomial correction. In subdomain $\mathcal{D}_X^{1,2} : [0.5, 1]$, the same problem is observed and the insufficient maximum degree results only in a small global improvement. In the next step (Figure c), the residual in $\mathcal{D}_X^{2,1}$, $\mathcal{D}_X^{2,2}$ and $\mathcal{D}_X^{2,4}$ is significantly reduced to a very small oscillation around 0. After the final step (Figure d), the overall approximation is has a high accuracy.

From the residual progression it can be seen that the algorithm needs more levels to accurately approximate the target function near regions of high complexity, *i.e.*, near the exponential peak. While this property does not affect the convergence when an experimental design of fixed size is chosen, it can be exploited in adaptive experimental design settings (Wagner et al., 2020).

As expected, the final SSE accuracy increases with the size of the experimental design. In Figure 4, we compare SSE and PCE on a set of experimental design sizes of $N_{ED} = \{10, 50, 100, 200\}$ in terms of their relative mean squared error (MSE, Eq. (34)). PCE is constructed with a maximum adaptive degree of $p_{\max}^{\text{PCE}} = 20$. At the extremely small experimental design of $N_{ED} = 10$, the SSE approach is comparable to PCE. As the available experimental design points increase, SSE exhibits faster convergence in RMSE than PCE, and from $N_{ED} = 50$ onwards SSE consistently outperforms PCE in this problem. At larger experimental designs, SSE can accurately reproduce the localized behavior of this test function, while not being constrained by the global nature of PCE basis functions defined on the full domain. At the final ED size of $N_{ED} = 200$

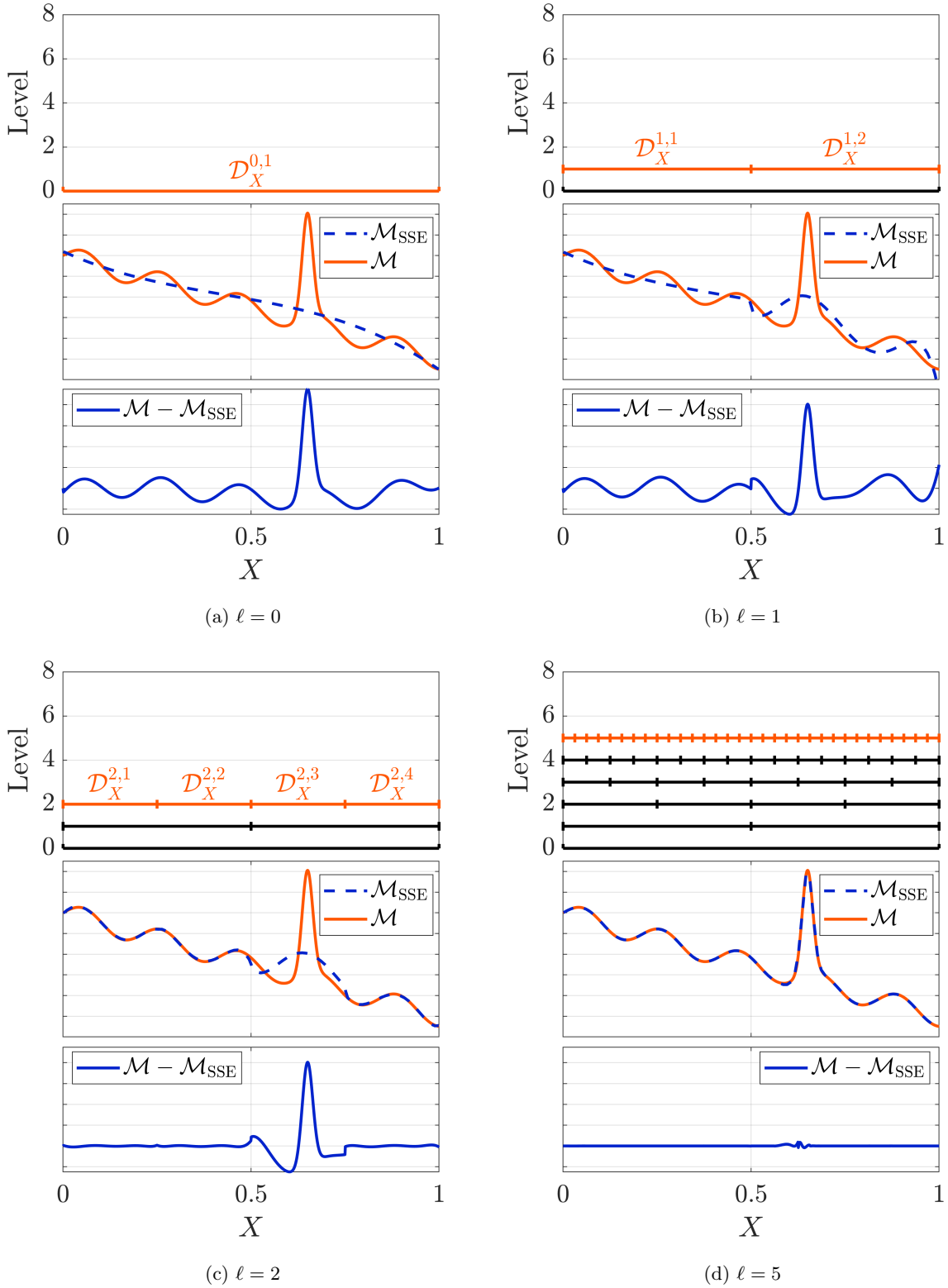


Figure 3: *One-dimensional analytical function*: selected steps of the SSE construction and the resulting domains, residuals and total approximation. The terminal domains (Eq. (29)) are highlighted in orange.

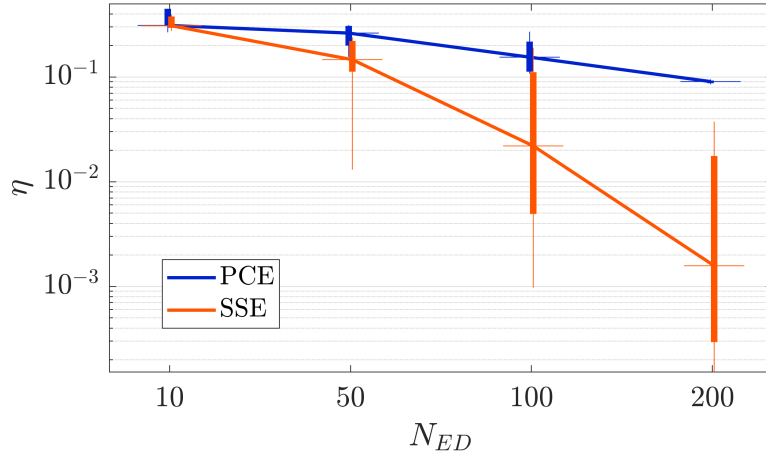


Figure 4: *One-dimensional analytical function*: comparison of RMSE convergence between PCE and SSE as a function of the number of points in the experimental design. A slight horizontal offset is added to improve readability.

the SSE relative MSE is at least one order of magnitude smaller than the PCE error. There is considerable variability in the relative MSE between individual realizations. This can be attributed to the squared exponential peak in Eq. (35): depending on the input realizations in the experimental design, it is captured better or worse by the available data.

4.4 Application 2: 100-dimensional analytical function

With this example we want to explore the scalability of SSE in high dimensional problems. This example uses a variant of a test function introduced in Zhou (1998). We modified the function to have a high nominal dimensionality ($M = 100$), and relatively low effective dimensionality meaning that the majority of the variability can be attributed to a small number of input parameters. It takes the form

$$\mathcal{M}(\mathbf{X}) = \frac{10^M}{2} [\varphi(10 \cdot (\mathbf{X} - 1/3)) + \varphi(10 \cdot (\mathbf{X} - 2/3))], \quad (36)$$

where $\varphi(\mathbf{x}) \stackrel{\text{def}}{=} (2\pi)^{-M/2} \exp\left(-\frac{1}{2} \sum_{i=1}^M a_i^2 x_i^2\right)$.

The factor $\mathbf{a} = (a_1, \dots, a_M)$ modifies the original function and serves as a dimension-dependent weight that decays exponentially, as:

$$a_i = e^{-(i-1)}, \quad \text{with } i \in \{1, \dots, M\}. \quad (37)$$

The input random vector \mathbf{X} is distributed according to a multivariate standard uniform distribution with independent marginals, $\mathbf{X} \sim f_{\mathbf{X}}(\mathbf{X}) = \prod_{i=1}^{100} \mathcal{U}(0, 1)$.

Two contour cross-section plots of this function are shown between dimensions $\{X_1, X_2\}$ in Figure a and between $\{X_1, X_{10}\}$ in Figure b. The effect of the decay factor α is clearly visible and results in a *close-to* constant behavior along the parametric dimension X_{10} .

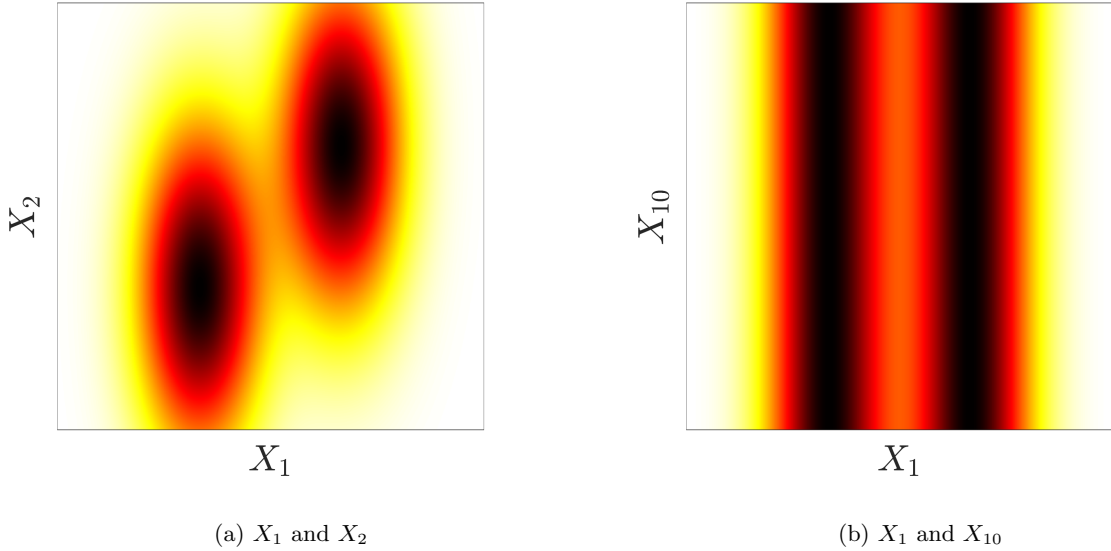


Figure 5: *100-dimensional analytical function*: bivariate contour cross-section plots.

To manage the computational complexity, in this example we limit the maximum polynomial degree in SSE to $p_{\max}^{\text{SSE}} = 2$ and compute the SSE with total experimental design sizes of $N_{ED} = \{1,000; 2,000; 5,000; 10,000\}$. For PCE, we choose a maximum degree of $p_{\max}^{\text{SSE}} = 7$, which is the maximum degree we could run on a standard desktop computer with 16GB of RAM before incurring memory issues. The resulting comparison between PCE and SSE with respect to the relative MSE is plotted in Figure 6. In this scenario, the SSE algorithm outperforms PCE on all investigated experimental designs. In fact, PCE seems to benefit from increasing the experimental design only marginally, with a relative error of $\eta \approx 0.25$ for all considered values of N_{ED} . SSE shows instead a convincing convergence behavior, by reducing its residual by almost two orders of magnitude across the various experimental designs. Given the high dimensionality of this problem, the approximation power of sparse PCE is limited by the curse of dimensionality, rather than the lack of data. By reducing the complexity of the spectral representation at each level, SSE can better exploit informative datasets without incurring similar computational bottlenecks.

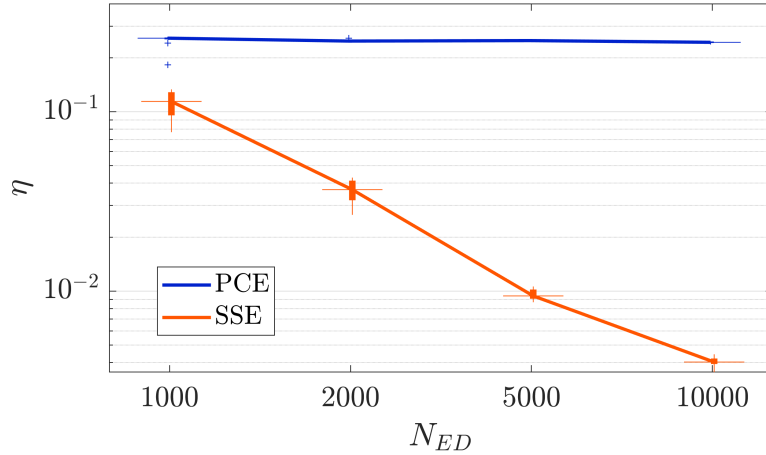


Figure 6: *100-dimensional analytical function*: comparison of RMSE convergence between PCE and SSE as a function of the number of points in the experimental design. A slight horizontal offset is added to improve readability.

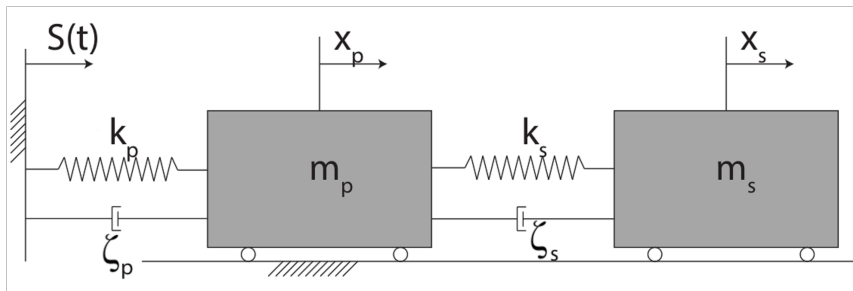


Figure 7: *8-dimensional damped oscillator*: model setup.

4.5 Application 3: damped oscillator

Damped oscillators are a class of engineering models that is commonly used in structural reliability problems (Dubourg, 2011). This class of problems is known to be often difficult to surrogate, due its high non-linearity and often local behavior. A sketch of the oscillator considered in this example is displayed in Figure 7. It consists of a primary and secondary system with masses m_p, m_s , stiffnesses k_p, k_s and damping ratios ζ_p, ζ_s . The subscripts p and s denote the *primary* and *secondary* system properties, respectively.

In this example we consider the limit state function of the damped oscillator given by

$$\mathcal{M}(\mathbf{X}) = F_s - p \cdot k_s \sqrt{\mathbb{E}_S [x_S^2]}, \quad (38)$$

where F_s is the force capacity of the secondary spring, p is the *so-called* peak factor and x_S is the relative displacement between the primary and secondary systems. The mean-square relative

Variable	Description	Distribution	Mean	C.O.V.
m_p	primary mass	Lognormal	1.50	0.1
m_s	secondary mass	Lognormal	0.01	0.1
k_p	primary spring stiffness	Lognormal	1.00	0.2
k_s	secondary spring stiffness	Lognormal	0.01	0.2
ζ_p	primary damping ratio	Lognormal	0.05	0.4
ζ_s	secondary damping ratio	Lognormal	0.02	0.5
S_0	white noise intensity	Lognormal	100.00	0.1
F_s	secondary spring force capacity	Lognormal	15.00	0.1

Table 1: *8-dimensional damped oscillator*: marginal distributions.

displacement of the secondary spring under a white noise base acceleration S is analytically given by:

$$\mathbb{E}_S [x_S^2] = \pi \frac{S_0}{4\zeta_s\omega_s^3} \frac{\zeta_a\zeta_s}{\zeta_p\zeta_s(4\zeta_a^2 + \theta^2) + \gamma\zeta_a^2} \frac{(\zeta_p\omega_p^3 + \zeta_s\omega_s^3)\omega_p}{4\zeta_a\omega_a^4}, \quad (39)$$

where S_0 is the white noise intensity, $\omega_p = \sqrt{k_p/m_p}$ and $\omega_s = \sqrt{k_s/m_s}$ are the natural frequencies of the two subsystems, and the further abbreviations are used: $\gamma = m_s/m_p$, $\omega_a = (\omega_p + \omega_s)/2$, $\zeta_a = (\zeta_p + \zeta_s)/2$ and $\theta = (\omega_p - \omega_s)/\omega_a$.

All variables but the peak factor (set to $p = 3$) are modelled as independent random variables and are summarized in the random vector $\mathbf{X} = \{m_p, m_s, k_p, k_s, \zeta_p, \zeta_s, S_0, F_s\}$. Their marginal distributions are lognormal, with the parameters given in Table 1.

For the convergence study, we choose experimental design sizes of $N_{ED} = \{1,000; 5,000; 10,000; 20,000\}$. The maximum degrees are set to $p_{\max}^{\text{SSE}} = 4$ and $p_{\max}^{\text{PCE}} = 10$ for SSE and PCE, respectively. Figure 8 summarizes the results. This benchmark is known to be quite difficult to approximate with standard surrogate model techniques, as it is clear from the scale of the RMSE in Figure 8. For the two smaller experimental design sizes $N_{ED} = \{1,000; 5,000\}$, both PCE and SSE perform quite poorly, with SSE showing a similar median behavior, but much higher variability. For larger experimental designs, however, the SSE performance improves significantly over that of PCE, until at $N_{ED} = 20,000$ the relative MSE of SSE is half that of PCE. This behavior is in line with the previous findings: provided enough information, SSE can provide higher expressive power than its static counterpart.

4.6 Application 4: truss with discontinuous snap-through behavior

As a last example, we address another problem of engineering interest: the geometrically nonlinear two-bar truss structure shown in Figure 9. The structure itself is defined by the initial

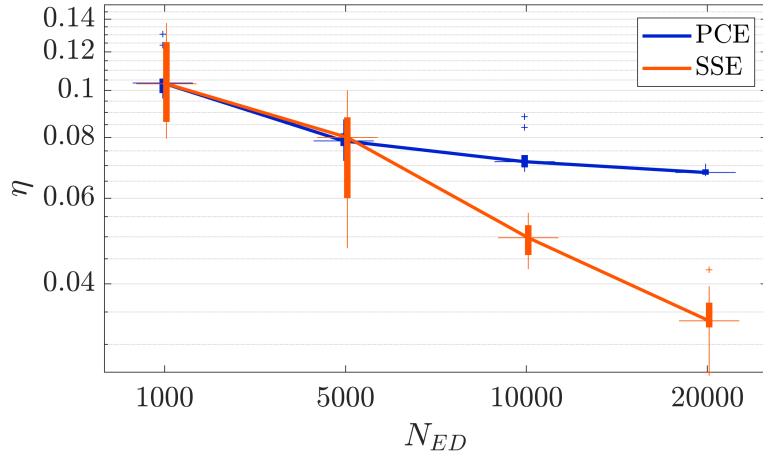


Figure 8: *8-dimensional damped oscillator*: comparison of RMSE convergence between PCE and SSE as a function of the number of points in the experimental design. A slight horizontal offset is added to improve readability.

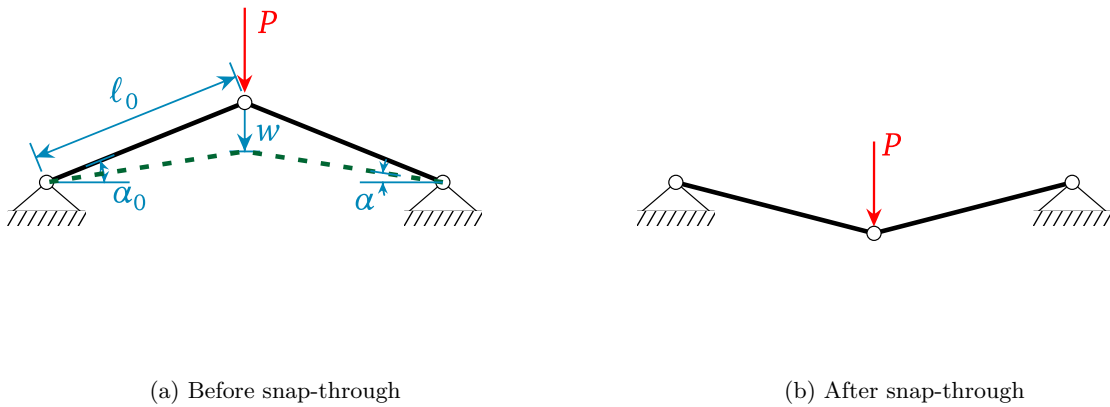


Figure 9: *3-dimensional snap through truss*: illustration of the truss structure subject to snap-through

Variable	Description	Distribution	Mean	C.O.V.
P	load	Gumbel	430	0.20
E	Young's modulus	Lognormal	210	0.10
A	cross sectional area	Gaussian	10	0.05

Table 2: *3-dimensional snap-through truss*: marginal distributions.

inclination α_0 and length ℓ_0 of the two bars. A peculiarity of this structure is that it exhibits the so-called *snap-through* behavior. At first, the vertical displacement w of such a structure typically increases linearly with an increasing load P (Figure a). Once a specific critical load is exceeded, the structure *snaps through* to another equilibrium point, at which the load can be increased further (Figure b). The main implication of this kind of behavior is that it is discontinuous in some critical regions of the input space, which are in general unknown *a priori*.

The vertical displacement w of the truss tip is related to the angle α by

$$\mathcal{M}(\mathbf{X}) = w = \ell_0 \cos \alpha_0 (\tan \alpha_0 - \tan \alpha(\mathbf{X})). \quad (40)$$

At the same time, α needs to satisfy the following constitutive equation that depends on the random vector $\mathbf{X} = \{P, E, A\}$:

$$P = -2EA \tan \alpha (\cos \alpha_0 - \cos \alpha). \quad (41)$$

For a given realization of \mathbf{X} , this equation can be solved numerically for α , the value of which then is used in Eq. (40) to estimate the corresponding vertical displacement.

In this study we set the constants $\ell_0 = 5$ m and $\alpha_0 = 10^\circ$, and treat the parameters \mathbf{X} as independent random variables with marginals listed in Table 2 (Moustapha and Sudret, 2019). We investigate experimental designs of sizes $N_{ED} = \{100; 500; 1,000; 2,000; 5,000\}$ and set the maximum polynomial degrees to $p_{\max}^{\text{SSE}} = 4$ and $p_{\max}^{\text{PCE}} = 10$ for SSE and PCE, respectively.

Figure 10 summarizes the convergence behavior of PCE and SSE in this benchmark. For all experimental designs, SSE outperforms sparse PCE. The dispersion of the RMSE is also significantly improved. These observations can be explained with the well-known *Gibbs phenomenon* in spectral representations, that leads to large discrepancies close to discontinuities. The effect is far less severe (although still present) for SSE than for PCE because it is restricted to those subdomains that actually contain the discontinuity. This behavior is investigated more closely in Figure 11, where a cross section through \mathcal{M} is shown. It is created by setting A to its mean value and drawing a map proportional to the point-wise discrepancy in the remaining directions. We adjust transparency of the model response to reflect the underlying joint PDF: solid colors correspond to high probability, fading ones to low probability. Figures a and b show the relative

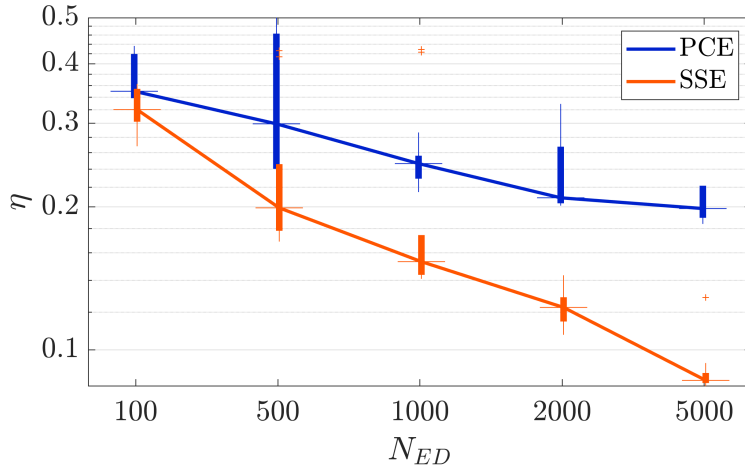


Figure 10: *3-dimensional snap-through truss*: comparison of RMSE convergence between PCE and SSE as a function of the number of points in the experimental design. A slight horizontal offset is added to improve readability.

point-wise error at an experimental design size of $N_{ED} = 5,000$ for PCE and SSE, respectively. Figure c shows instead the domain-wise error estimator $\widehat{E}_{LOO}^{\ell,p}$ from Eq. (25).

All plots clearly show the effect of the function discontinuity. For PCE the Gibbs phenomenon is clearly visible. It causes a large error near the discontinuity, with an oscillating error at large distances to the discontinuity. Naturally, SSE also suffers from the same problem, but its effect is only localized close to the discontinuity and it does not affect further regions. Furthermore, the available domain-wise local error estimator in Figure c gives a clear indication of local loss of accuracy of SSE. In practical applications this information can be crucial, as it allows one to assess the confidence of the surrogate model predictions, and could be used to adaptively enrich the experimental design close to critical regions (Wagner et al., 2020).

4.7 Error estimation accuracy

For practical applications, it is important that surrogate models offer insights into their prediction accuracy. Most surrogate models offer global confidence bounds on their predictions that are typically computed through cross-validation techniques (*e.g.*, leave-one-out error), while some techniques also offer point-wise confidence bounds (*e.g.*, Kriging Santner et al. (2003) and bootstrap PCE (Marelli and Sudret, 2018)).

For SSE, the domain-wise error estimators of the local expansions give some insight into the local accuracy of SSE, as shown in the last case study (Section 4.6). The weighted sum of those domain-wise estimators can be used as a global estimator of the generalization error as proposed

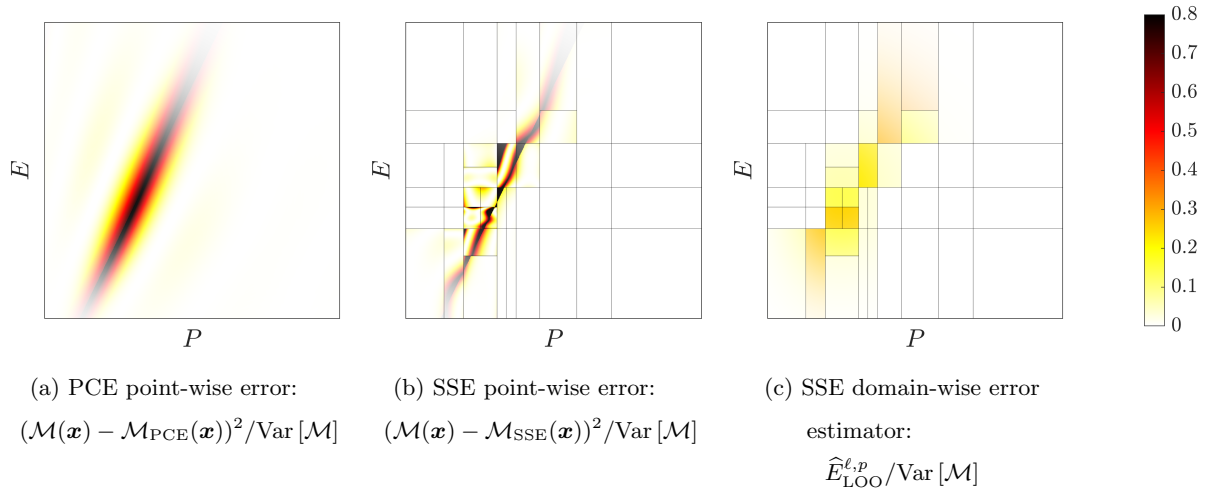
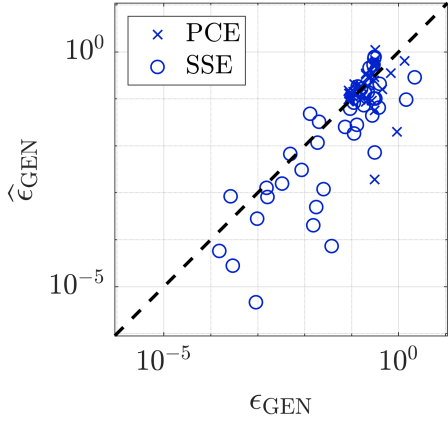


Figure 11: *3-dimensional snap-through truss*: comparison between PCE/SSE pointwise error and SSE domain-wise error estimator from Eq. (25)) normalized by the model variance. The fine lines show the SSE subdomains.

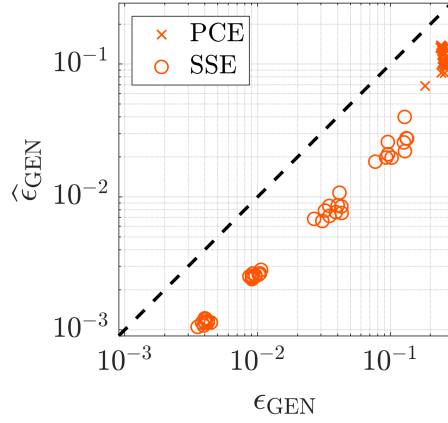
in Section 3.4, Eq. (26). To assess the accuracy of this global estimator, in Figure 12 we plot it against the relative MSE on a validation set for all presented case studies. The diagonal dashed line corresponds to perfect error estimation: $\eta = \hat{\varepsilon}_{\text{GEN}}$. In all the applications $\hat{\varepsilon}_{\text{GEN}}$ significantly underestimates the true error. For comparison, we show the corresponding LOO estimator from PCE, which also exhibits a bias towards lower errors. On the other hand, it is clear that there is a strong correlation the SSE estimators and the true validation error across all applications, which still makes $\hat{\varepsilon}_{\text{GEN}}$ a strong potential candidate for model selection in future extensions of the method to adaptively select the SSE hyperparameters (*e.g.*, adaptively choosing $p_{\text{max}}^{\text{SSE}}$).

4.8 Considerations on computational costs and scalability

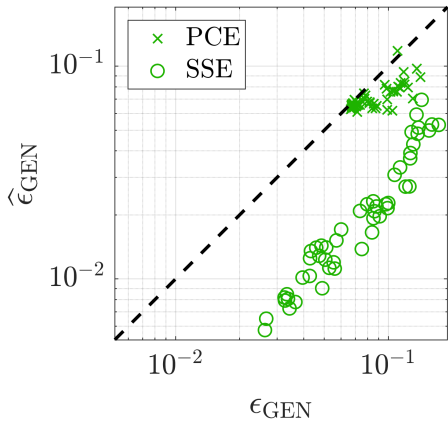
For a metamodeling technique to be relevant in an engineering context, the associated training and prediction costs need to be negligible with respect to the costs of the underlying physics-based computational model. In this sense, SSE performs quite well, as its computational costs scale primarily with the size of the experimental design. Indeed, because the residual expansions (Eq. (9)) can be chosen with very low degree or otherwise strict truncation, the driving cost is the total number of subdomains that are expanded in the SSE sparse tree (see Section 3.3). Interestingly, the expected number of expansions N_E depends only on the ratio between the total experimental design size N and the minimum number of points in each subdomain needed



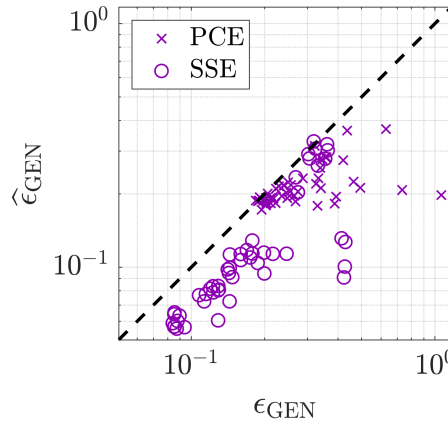
(a) One-dimensional analytical function



(b) 100-dimensional analytical function



(c) 8-dimensional damped oscillator



(d) 3-dimensional snap-through truss

Figure 12: Comparison of the relative generalization error ϵ_{GEN} with the relative generalization error estimator $\hat{\epsilon}_{\text{GEN}}$ which for SSE is given by Eq. (26) and for PCE is the relative leave-one-out error $E_{\text{LOO}}/\text{Var}[Y]$.

to perform a residual expansion, N_{\min} , as in (see also Eq. (20)):

$$N_{\text{PCE}} = \sum_{\ell=0}^{\bar{L}} 2^\ell = 2^{\bar{L}+1} - 1 = 2^{\lceil \log_2(N/N_{\min}) \rceil + 1} - 1 \leq 2N/N_{\min} - 1. \quad (42)$$

In other words, the computational complexity increases at most linearly with the number of points in the experimental design. Additionally, the storage costs can be further reduced to $2^{\bar{L}} = N/N_{\min}$ expansions when using the flattened representation in Eq. (29). Therefore, as required, the training and evaluation costs of SSE are normally negligible with respect to those needed to produce a training set for any realistic engineering application.

5 Conclusions

In an effort to extend the applicability of the powerful class of spectral decomposition-based metamodels, we propose a novel metamodel technique called *stochastic spectral embedding* (SSE), that exploits both recent advances in UQ (sparse spectral expansions) and in machine learning (regression trees). While our presentation was general in nature, we showed how well this approach synergizes with sparse polynomial chaos expansions. We also provided analytical formulas to calculate several statistical properties of the resulting model by means of the so-called *flattened representation*, which has additional benefits in terms of computational costs.

We tested the performance of SSE on both simple test functions and engineering-like examples of varying dimensionality and complexity, using varying experimental design sizes, and compared it to our best performing sparse PCE. Its generalization capabilities, especially for highly complex models and large experimental designs, outperform PCE in most cases.

We also demonstrated that the associated computational costs of the training of SSE scale linearly in expectation with the number of points in the experimental design. This compares favorably with most metamodeling techniques common in the UQ community (*e.g.* PCE or Kriging).

This performance, however, comes at the cost of trading the continuity of PCE for the piecewise continuity of SSE. This also implies the loss of the effective generalization error estimate provided by $\hat{\epsilon}_{\text{LOO}}$ in linear regression. To mitigate this issue, we proposed the error estimate $\hat{\epsilon}_{\text{GEN}}$ (Eq. (26)). Despite its absolute scale being biased towards lower values, it still shows high correlation with the actual generalization error for all experimental design sizes and dimensions. This is a promising property for further research into providing automatic selection of the hyperparameters of the algorithm (which at the moment are the maximum degree of the residual expansions $p_{\text{max}}^{\text{SSE}}$, as well as the minimum number of points in each subdomain N_{\min} required to expand the residual) and further enhance its performance.

Additional research is ongoing towards the use of the approximate local error measures pro-

vided by $\widehat{E}_{\text{GEN}}^{\ell,p}$ (Eq. (22)) for goal-oriented adaptive experimental design construction, a topic explored by the authors in Wagner et al. (2020).

Acknowledgements

The PhD thesis of the second author is supported by ETH grant #44 17-1.

A Postprocessing PCE-based SSE

If polynomials chaos expansions are used to construct the residual expansions, several quantities of engineering interest can be computed analytically as a post-processing step of the final SSE. In this section we derive expressions for (i) *conditional expectations*, (ii) *partial variances* and (iii) Sobol' indices.

A.1 Conditional expectations

Conditional expectations describe the expectation of a multivariate function of a random vector $\mathbf{X} \sim f_{\mathbf{X}}(\mathbf{x})$, conditioned on a subset of \mathbf{X} assuming a fixed value. Let $\mathbf{X} = \{X_i\}_{i=1,\dots,M} \in \mathcal{D}_{\mathbf{X}}$ be an independent random vector with PDF $f_{\mathbf{X}}(\mathbf{x}) = \prod_{i=1}^M f_{X_i}(x)$. Denote further by \mathbf{u} and \mathbf{v} two disjoint index sets such that $\mathbf{u} \cup \mathbf{v} = \{1, \dots, M\}$ and by $\mathbf{X}_{\mathbf{u}} \stackrel{\text{def}}{=} \{X_i\}_{i \in \mathbf{u}} \in \mathcal{D}_{\mathbf{X}_{\mathbf{u}}}$ a random sub-vector with PDF $f_{\mathbf{X}_{\mathbf{u}}}(\mathbf{x}_{\mathbf{u}}) = \prod_{i \in \mathbf{u}} f_{X_i}(x)$. Additionally define the complementary random vector $\mathbf{X}_{\mathbf{v}} \stackrel{\text{def}}{=} \{X_i\}_{i \in \mathbf{v}} \in \mathcal{D}_{\mathbf{X}_{\mathbf{v}}}$. The conditional expectation of $\mathcal{M}_{\text{SSE}}(\mathbf{X})$ w.r.t. $\mathbf{X}_{\mathbf{u}}$ can then be written as

$$\mathbb{E}[\mathcal{M}_{\text{SSE}}(\mathbf{X}) | \mathbf{X}_{\mathbf{u}}] \stackrel{\text{def}}{=} \int_{\mathcal{D}_{\mathbf{X}_{\mathbf{v}}}} \mathcal{M}_{\text{SSE}}^F(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}_{\mathbf{v}}, \quad (43)$$

where we used the *flattened representation* from Eq. (29). This corresponds to marginalizing over the parameters $\mathbf{X}_{\mathbf{v}}$. Due to the local orthonormality of the SSE representation, an analytical expression for this integral can be found as

$$\mathbb{E}[\mathcal{M}_{\text{SSE}}(\mathbf{X}) | \mathbf{X}_{\mathbf{u}}] = \sum_{p=1}^{P_L} \mathcal{V}_{\mathbf{v}}^{L,p} \sum_{\alpha \in \mathcal{A}_{\mathbf{v}=0}^T} c_{\alpha}^p \Psi_{\alpha, \mathbf{u}}^{L,p}(\mathbf{X}_{\mathbf{u}}^{L,p}), \quad (44)$$

where $\mathcal{V}_{\mathbf{v}}^{L,p} \stackrel{\text{def}}{=} \int_{\mathcal{D}_{\mathbf{X}_{\mathbf{v}}}} f_{\mathbf{X}_{\mathbf{v}}}(\mathbf{x}_{\mathbf{v}}) d\mathbf{x}_{\mathbf{v}}$ is the input mass in the marginalized dimensions and $\mathcal{A}_{\mathbf{v}=0}^T \stackrel{\text{def}}{=} \{\alpha \in \mathcal{A}^T : \alpha_i = 0 \Leftrightarrow i \in \mathbf{v}\}$. Further, $\mathbf{X}_{\mathbf{u}}^{L,p} \in \mathcal{D}_{\mathbf{X}_{\mathbf{u}}^{L,p}}$ is an auxiliary random variable that is only defined in the (L,p) -subdomain and $\Psi_{\alpha, \mathbf{u}}^{L,p}(\mathbf{X}_{\mathbf{u}}^{L,p}) \stackrel{\text{def}}{=} \prod_{i \in \mathbf{u}} \Phi_{\alpha_i}^{(i),L,p}(X_i^{L,p})$ is a polynomial basis function of the non-marginalized variables.

The marginalization process in Eq. (43) creates an additional problem: this expression generally contains overlapping subdomains $\{\mathcal{D}_{\mathbf{X}_{\mathbf{u}}^{L,p}}\}_{p=1,\dots,P_L}$ due to the fact that terminal subdomains

in the full input space are not necessarily terminal subdomains in the lower dimensional, conditional expectation input space defined by \mathbf{u} . However, because the basis functions are polynomials, it is once again possible to perform the *flattening* process (see Section 3.4) and obtain disjoint subdomains. By denoting as $\mathcal{P} \subseteq \{1, \dots, P_L\}$ the set of terminal subdomains in the conditional input variables $\mathbf{X}_{\mathbf{u}}$, we can rewrite Eq. (44) as

$$\mathbb{E}[\mathcal{M}_{\text{SSE}}(\mathbf{X}) | \mathbf{X}_{\mathbf{u}}] = \sum_{p \in \mathcal{P}} \sum_{\alpha \in \mathcal{A}_{\mathbf{v}=0}^p} d_{\alpha}^p \Psi_{\alpha, \mathbf{u}}^{L,p}(\mathbf{X}_{\mathbf{u}}^{L,p}), \quad (45)$$

where $\mathcal{A}_{\mathbf{v}=0}^p$ is a suitable multi-index set allowing an exact representation of the polynomials and d_{α}^p are the corresponding coefficients.

A.2 Partial variance and Sobol' indices

The Sobol' Hoeffding decomposition (Sobol', 1993; Le Gratiet et al., 2016) of the SSE representation \mathcal{M}_{SSE} reads

$$\mathcal{M}_{\text{SSE}}(\mathbf{X}) = \mathcal{M}_{\text{SSE}}^0 + \sum_{\substack{\mathbf{u} \subset \{1, \dots, M\} \\ \mathbf{u} \neq \emptyset}} \mathcal{M}_{\text{SSE}}^{\mathbf{u}}(\mathbf{X}_{\mathbf{u}}), \quad (46)$$

where $\mathcal{M}_{\text{SSE}}^0 \stackrel{\text{def}}{=} \mathbb{E}[\mathcal{M}_{\text{SSE}}(\mathbf{X})]$ and the remaining terms can be computed recursively by

$$\mathcal{M}_{\text{SSE}}^i(X_i) = \mathbb{E}_{\mathbf{X}_{\sim i}}[\mathcal{M}_{\text{SSE}}(\mathbf{X})] - \mathcal{M}_{\text{SSE}}^0, \quad (47)$$

$$\mathcal{M}_{\text{SSE}}^{ij}(\mathbf{X}_{ij}) = \mathbb{E}_{\mathbf{X}_{\sim ij}}[\mathcal{M}_{\text{SSE}}(\mathbf{X})] - \mathcal{M}_{\text{SSE}}^i(X_i) - \mathcal{M}_{\text{SSE}}^j(X_j) - \mathcal{M}_{\text{SSE}}^0, \quad (48)$$

$$\dots = \dots \quad (49)$$

$$\mathcal{M}_{\text{SSE}}^{\mathbf{u}}(\mathbf{X}_{\mathbf{u}}) = \mathbb{E}_{\mathbf{X}_{\mathbf{v}}}[\mathcal{M}_{\text{SSE}}(\mathbf{X})] - \sum_{\substack{\mathbf{w} \subset \mathbf{u} \\ \mathbf{w} \neq \emptyset}} \mathcal{M}_{\text{SSE}}^{\mathbf{w}}(\mathbf{X}_{\mathbf{w}}) - \mathcal{M}_{\text{SSE}}^0. \quad (50)$$

The decomposition of Eq. (46), allows the definition of the so-called *partial variance*, *i.e.*, the fraction of the variance $\text{Var}[\mathcal{M}_{\text{SSE}}(\mathbf{X})]$ that can be attributed to $\mathbf{X}_{\mathbf{u}}$, defined by

$$V_{\mathbf{u}} \stackrel{\text{def}}{=} \int_{\mathcal{D}_{\mathbf{X}_{\mathbf{u}}}} (\mathcal{M}_{\text{SSE}}^{\mathbf{u}}(\mathbf{x}_{\mathbf{u}}))^2 f_{\mathbf{X}_{\mathbf{u}}}(\mathbf{x}_{\mathbf{u}}) d\mathbf{x}_{\mathbf{u}}. \quad (51)$$

Using Eq. (47), the so-called *first order partial variance* can therefore be written as

$$V_i = \int_{\mathcal{D}_{X_i}} (\mathbb{E}_{\mathbf{X}_{\sim i}}[\mathcal{M}_{\text{SSE}}(\mathbf{X})] - \mathcal{M}_{\text{SSE}}^0)^2 f_{X_i}(x_i) dx_i. \quad (52)$$

With the expression for the conditional expectation from Eq. (45), this integral can be solved analytically as

$$V_i = \int_{\mathcal{D}_{X_i}} \left(\sum_{p \in \mathcal{P}} \sum_{\alpha \in \mathcal{A}_{\sim i=0}^p} d_{\alpha}^p \Phi_{\alpha_i}^{(i),L,p}(X_i^{L,p}) - \mathcal{M}_{\text{SSE}}^0 \right)^2 f_{X_i}(x_i) dx_i \quad (53)$$

$$= \sum_{p \in \mathcal{P}} \mathcal{V}_i^{L,p} \sum_{\alpha \in \mathcal{A}_{\sim i=0}^p} (d_{\alpha}^p)^2 - (\mathcal{M}_{\text{SSE}}^0)^2. \quad (54)$$

With the availability of the partial variances in Eq. (52) and the total variance from Eq. (31), one can analytically derive the *first order Sobol' indices*:

$$S_i \stackrel{\text{def}}{=} \frac{V_i}{\text{Var}[\mathcal{M}_{\text{SSE}}(\mathbf{X})]}. \quad (55)$$

Higher order indices can be computed in a similar way.

References

- Abbiati, G., S. Marelli, N. Tsokanas, B. Sudret, and B. Stojadinović (2021). A global sensitivity analysis framework for hybrid simulation. *Mechanical Systems and Signal Processing* 146, 106997.
- Berveiller, M., B. Sudret, and M. Lemaire (2006). Stochastic finite elements: a non intrusive approach by regression. *European Journal of Mechanics* 15(1–3), 81–92.
- Blatman, G. and B. Sudret (2010). An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. *Probabilistic Engineering Mechanics* 25(2), 183–197.
- Blatman, G. and B. Sudret (2011). Adaptive sparse polynomial chaos expansion based on Least Angle Regression. *Journal of Computational Physics* 230, 2345–2367.
- Bourinet, J.-M. (2016). Rare-event probability estimation with adaptive support vector regression surrogates. *Reliability Engineering & System Safety* 150, 210 – 221.
- Breiman, L. (2017). *Classification and regression trees*. Routledge.
- Bungartz, H.-J. and M. Griebel (2004). Sparse grids. *Acta Numerica* 13, 147–269.
- Candès, E. J. and M. B. Wakin (2008). An introduction to compressive sampling: A sensing/sampling paradigm that goes against the common knowledge in data acquisition. *IEEE Signal Processing Magazine* 25(2), 21–30.
- Chapelle, O., V. Vapnik, and Y. Bengio (2002). Model selection for small sample regression. *Journal of Machine Learning Research* 48(1), 9–23.
- Chipman, H. A., E. I. George, and R. E. McCulloch (2010, March). Bart: Bayesian additive regression trees. *Annals of Applied Statistics* 4(1), 266–298.
- Donoho, D. L. et al. (2006). Compressed sensing. *IEEE Transactions on information theory* 52(4), 1289–1306.

- Dubourg, V. (2011). *Adaptive surrogate models for reliability analysis and reliability-based design optimization*. Ph. D. thesis, Université Blaise Pascal, Clermont-Ferrand, France.
- Ernst, O., A. Mugler, H.-J. Starkloff, and E. Ullmann (2012). On the convergence of generalized polynomial chaos expansions. *Search Results Web results ESAIM: Mathematical Modelling and Numerical Analysis* 46(02), 317–339.
- Foo, J., X. Wan, and G. E. Karniadakis (2008). The multi-element probabilistic collocation method (ME-PCM): Error analysis and applications. *Journal of Computational Physics* 227(22), 9572 – 9595.
- Friedman, J. H. (1991). Multivariate adaptive regression splines. *Annals of Statistics* 19(1), 1–67.
- Gautschi, W. (2004). *Orthogonal polynomials: computation and approximation*. Oxford University Press.
- Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*. MIT Press.
- Guo, X. and D. Dias (2020). Kriging based reliability and sensitivity analysis—application to the stability of an earth dam. *Computers and Geotechnics* 120, 103411.
- Harenberg, D., S. Marelli, B. Sudret, and V. Winschel (2019, January). Uncertainty quantification and global sensitivity analysis for economic models. *Quantitative Economics* 10, 1–41.
- Hrinda, G. (2010). Snap-through instability patterns in truss structures. In *51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 18th AIAA/ASME/AHS Adaptive Structures Conference 12th*, pp. 2611.
- Kersaudy, P., B. Sudret, N. Varsier, O. Picon, and J. Wiart (2015). A new surrogate modeling technique combining Kriging and polynomial chaos expansions – Application to uncertainty analysis in computational dosimetry. *Journal of Computational Physics* 286, 103–117.
- Lataniotis, C., S. Marelli, and B. Sudret (2020). Extending classical surrogate modeling to high dimensions through supervised dimensionality reduction: a data-driven approach. *International Journal for Uncertainty Quantification* 10(1), 55–82.
- Le Gratiet, L., S. Marelli, and B. Sudret (2016). *Metamodel-based sensitivity analysis: polynomial chaos expansions and Gaussian processes*, Chapter 38, Handbook on Uncertainty Quantification (Ghanem, R. and Higdon, D. and Owhadi, H. (Eds.), pp. 1289–1325. Springer.

- Lowry, C. A., W. H. Woodall, C. W. Champ, and S. E. Rigdon (1992). A multivariate exponentially weighted moving average control chart. *Technometrics* 34(1), 46–53.
- Lüthen, N., S. Marelli, and B. Sudret (2020). Sparse polynomial chaos expansions: Literature survey and benchmark. *arXiv preprint arXiv:2002.01290*. submitted to: SIAM journal for Uncertainty Quantification.
- Maître, O. L., H. Najm, R. Ghanem, and O. Knio (2004). Multi-resolution analysis of Wiener-type uncertainty propagation schemes. *Journal of Computational Physics* 197(2), 502 – 531.
- Marelli, S. and B. Sudret (2014). UQLab: A framework for uncertainty quantification in Matlab. In *Vulnerability, Uncertainty, and Risk (Proc. 2nd Int. Conf. on Vulnerability, Risk Analysis and Management (ICVRAM2014), Liverpool, United Kingdom)*, pp. 2554–2563.
- Marelli, S. and B. Sudret (2018). An active-learning algorithm that combines sparse polynomial chaos expansions and bootstrap for structural reliability analysis. *Structural Safety* 75, 67–74.
- Marelli, S. and B. Sudret (2019). UQLab user manual – Polynomial chaos expansions. Technical report, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland. Report UQLab-V1.3-104.
- Moustapha, M. and B. Sudret (2019). A two-stage surrogate modelling approach for the approximation of models with non-smooth outputs. In M. Papadrakakis, V. Papadopoulos, and G. Stefanou (Eds.), *3rd ECCOMAS thematic conference on Uncertainty Quantification in Computational Sciences and Engineering (UNCECOMP 2019), Crete, Greece*.
- Moustapha, M., B. Sudret, J.-M. Bourinet, and B. Guillaume (2018). Comparative study of Kriging and support vector regression for structural engineering applications. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering* 4(2).
- Nagel, J. and B. Sudret (2016). Spectral likelihood expansions for Bayesian inference. *Journal of Computational Physics* 309, 267–294.
- Nelder, J. A. and R. W. M. Wedderburn (1972). Generalized linear models. *Journal of the Royal Statistical Society. Series A (General)* 135(3), 370–384.
- Pang, G., L. Lu, and G. E. Karniadakis (2019). fPINNs: Fractional physics-informed neural networks. *SIAM Journal on Scientific Computing* 41(4), A2603–A2626.
- Radaideh, M. I. and T. Kozlowski (2020). Analyzing nuclear reactor simulation data and uncertainty with the group method of data handling. *Nuclear Engineering and Technology* 52(2), 287–295.

- Raissi, M., P. Perdikaris, and G. Karniadakis (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* 378, 686 – 707.
- Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian processes for machine learning* (Internet ed.). Adaptive computation and machine learning. Cambridge, Massachusetts: MIT Press.
- Reinsch, C. H. (1967). Smoothing by spline functions. *Numerische mathematik* 10(3), 177–183.
- Rosenblatt, M. (1952). Remarks on a multivariate transformation. *Annals of Mathematical Statistics* 23(3), 470–472.
- Roustant, O., F. Barthe, and B. Iooss (2017). Poincaré inequalities on intervals – application to sensitivity analysis. *Electronic journal of statistics* 11(2), 3081–3119.
- Santner, T. J., B. J. Williams, and W. I. Notz (2003). *The Design and Analysis of Computer Experiments*. Springer, New York.
- Schöbi, R., B. Sudret, and J. Wiart (2015). Polynomial-chaos-based Kriging. *International Journal for Uncertainty Quantification* 5(2), 171–193.
- Serna, A. and C. Bucher (2009). Advanced surrogate models for multidisciplinary design optimization. *8th Weimar Optimization and Stochastic Days*.
- Shields, M. D. (2018). Adaptive monte carlo analysis for strongly nonlinear stochastic systems. *Reliability Engineering & System Safety* 175, 207–224.
- Slot, R. M., J. D. Sørensen, B. Sudret, L. Sørensen, and M. L. Thøgersen (2020). Surrogate model uncertainty in wind turbine reliability assessment. *Renewable Energy* 151, 1150–1162.
- Sobol’, I. M. (1993). Sensitivity estimates for nonlinear mathematical models. *Mathematical and Computer Modelling* 1, 407–414.
- Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering and System Safety* 93, 964–979.
- Torre, E., S. Marelli, P. Embrechts, and B. Sudret (2019a, July). Data-driven polynomial chaos expansion for machine learning regression. *Journal of Computational Physics* 388, 601–623.
- Torre, E., S. Marelli, P. Embrechts, and B. Sudret (2019b). A general framework for data-driven uncertainty quantification under complex input dependencies using vine copulas. *Probabilistic Engineering Mechanics* 55, 1–16.

- Vapnik, V. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.
- Wagner, P.-R., S. Marelli, and B. Sudret (2020). Bayesian model calibration with stochastic spectral embedding. *arXiv preprint arXiv:2005.07380*. submitted to: Journal of Computational Physics.
- Wan, X. and G. E. Karniadakis (2006). Multi-element generalized polynomial chaos for arbitrary probability measures. *SIAM Journal on Scientific Computing* 28(3), 901–928.
- Wang, H., Z. Yan, M. Shahidehpour, X. Xu, and Q. Zhou (2020). Quantitative evaluations of uncertainties in multivariate operations of microgrids. *IEEE Transactions on Smart Grid*.
- Xiu, D. and G. E. Karniadakis (2002). The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing* 24(2), 619–644.
- Zhou, Y. (1998). *Adaptive importance sampling for integration*. Ph. D. thesis, Stanford University.