Stochastic isogeometric analysis on arbitrary multipatch domains by spline dimensional decomposition[☆]

Ramin Jahanbin[*], Sharif Rahman

College of Engineering, The University of Iowa, Iowa City, IA 52242, USA

Abstract

This paper presents a new stochastic method by integrating spline dimensional decomposition (SDD) of a high-dimensional random function and isogeometric analysis (IGA) on arbitrary multipatch geometries to solve stochastic boundary-value problems from linear elasticity. The method, referred to as SDD-mIGA, involves (1) analysis-suitable T-splines with significant approximating power for geometrical modeling, random field discretization, and stress analysis; (2) Bézier extraction operator for isogeometric mesh refinement; and (3) a novel Fourier-like expansion of a high-dimensional output function in terms of measure-consistent orthonormalized splines. The proposed method can handle arbitrary multipatch domains in IGA and uses standard least-squares regression to efficiently estimate the SDD expansion coefficients for uncertainty quantification applications. Analytical formulae have been derived to calculate the second-moment properties of an SDD-mIGA approximation for a general output random variable of interest. Numerical results, including those obtained for a 54-dimensional, industrial-scale problem, demonstrate that a low-order SDD-mIGA is capable of efficiently delivering accurate probabilistic solutions when compared with the benchmark results from crude Monte Carlo simulation.
© 2022 Elsevier B.V. All rights reserved.

1. Introduction

Many complex mechanical systems exhibit uncertainties in their responses due to the intrinsic uncertainties in the input to the system. For instance, consider a system where some material properties, geometrical characteristics, and external loads are random based on the nature of the manufacturing processes and operational conditions. In such a case, if a stress response is defined as an output function, the response variable is naturally random as well. Uncertainties in the output are generally dependent on those of the input, and must be quantified accurately for meticulous design of complex systems. This process, referred to as uncertainty quantification (UQ), entails effectively propagating uncertainties from input to an output response variable of interest.

[*] Corresponding author.
 E-mail address: ramin-jahanbin@uiowa.edu (R. Jahanbin).

UQ often requires solving stochastic partial differential equations (PDEs) by numerical methods, such as Monte Carlo simulation (MCS) [1] as a direct sampling method, stochastic finite element methods [2], stochastic boundary-element methods [3], stochastic meshfree methods [4,5], non-intrusive sparse-grids methods [6], collocation methods [7], and so forth. Although MCS is popular owing to its simplicity, broad applicability, and natural suitability for parallel computing, it is often subject to slow convergence. For practical engineering problems mandating expensive-to-run computational analysis, MCS is hardly feasible or affordable. For the stochastic collocation methods, the crucial factor is how to construct the set of collocation points appropriately because the choice of the collocation points determines the performance of the method. While the basis splines (B-splines) have been employed to construct the sparse-grid quadrature or interpolation, including those associated with a hierarchical version [8], they are neither orthogonal nor measure-consistent, meaning that the basis functions are not adapted to the probability measure of input random variables. Nonetheless, the sparse grids are highly efficient, although they may require many levels for an accurate integration or interpolation, depending on the problem at hand. However, in the presence of nonsmooth or heavily oscillatory functions, the convergence properties of these aforementioned methods may deteriorate markedly when compared with those obtained for smooth functions. Isogeometric analysis (IGA) [9,10] has been proposed to solve PDEs by exploiting the outstanding approximating power of B-splines and non-uniform rational B-splines (NURBS) that are globally smooth over the physical domain, provided that the mapping between parametric and physical domains is smooth. IGA has the prospect of avoiding the mesh generation phase by using a single geometry representation that is simultaneously suitable for design and analysis and is capable of modeling many common geometries precisely. Over time, IGA has been improved on for water-tight modeling of significantly complex domains by employing T-splines [11,12] in computer-aided design (CAD) and computational mechanics. T-splines deliver the essential feature of local mesh refinement [13] over the regions where the response functions of interest are locally prominent. However, there are linear dependence challenges for T-spline blending functions [14,15]. Therefore, analysis-suitable T-splines, which correspond to analysis-suitable T-meshes have been proposed [16,17]. Analysis-suitable T-meshes have linearly independent underlying basis functions, desirable for computational analysis. Furthermore, hierarchical T-splines [18] and truncated T-splines [19] have been introduced, and IGA element data have been stored by the Bézier extraction operator [20,21] in search of a more accurate and convenient analysis framework. By adopting IGA in UQ, several numerical stochastic IGA (SIGA) methods have been proposed for vibration analysis of functionally graded plates [22], spectral analysis [23], and collocation-based IGA formulations [24]. Moreover, order reduction techniques have been implemented to accelerate MCS [25] and for more efficient SIGA [26].

There are numerous UQ methods applicable in the stochastic analysis of complex engineering systems. Polynomial chaos expansion (PCE) [27] is one of the most prevalent methods, although it lacks accuracy in the presence of non-smooth functions unless the polynomial order is significantly, if not prohibitively, high. The reason for this phenomenon is that PCE utilizes globally supported polynomials. In an effort to enhance the performance of globally supported PCE, domain decomposition techniques, such as multi-element formulation of PCE [28] was introduced. The idea was to decompose the domain based on some variance error threshold. However, in the presence of many subdomains, the multi-element PCE becomes computationally inefficient. Spline chaos expansion (SCE) [29] has been introduced by employing B-splines in the expansion instead of globally supported polynomials, thereby delivering the capability of effectively capturing locally abrupt changes in the response functions. Another challenge in UQ is to efficiently, yet accurately, solve high-dimensional stochastic problems, i.e., where the number of random variables is large and typically more than ten. While in such cases MCS benefits from its convergence rate not relying on the dimension of the problem, it is tremendously expensive and, hence, not practical for solving industrial-scale UQ problems. In addition, because of their tensor-product structure, PCE and SCE both succumb to the curse of dimensionality, where the number of expansion coefficients booms versus the increase in the number of random variables. As a remedy, polynomial dimensional decomposition (PDD) [30] and spline dimensional decomposition (SDD) [31] have been proposed based on the analysis-of-variance (ANOVA) dimensional decomposition (ADD) [32], to alleviate the computational difficulties with PCE and SCE, respectively. Essentially, the PDD and SDD methods are, when properly truncated, efficiently capable of expressing a high-dimensional function in terms of effective low-variate component functions with marginal loss of accuracy. However, PDD, very much like PCE, still employs globally supported polynomials and therefore struggles in the case of non-smooth or heavily oscillatory functions, while SDD addresses concerns with both the curse of dimensionality and ineffective modeling of non-smooth functions. When facing the curse of dimensionality, one advantage of SDD

over traditional stochastic Galerkin, stochastic collocation, and sparse grids methods is its capability of handling any input probability measure, because its underlying basis functions are measure-consistent. Therefore, the authors envision the need to present novel methods predicated on SDD to accurately solve the stochastic problems in linear elasticity. A novel SIGA method [33] has been developed by employing IGA as a non-intrusive deterministic solver of a general stochastic boundary-value problem (BVP) in a stochastic analysis framework involving SDD. The SDD-IGA method has been applied to linear elasticity problems and benefits from the exceptional properties and approximating power of B-splines and NURBS for CAD modeling, random field discretization, stress analysis, and UQ. Furthermore, SDD-IGA has been able to effectively handle smooth and non-smooth functions. However, it has been restricted to single-patch domains in the context of IGA. The motivation of this work is to develop a novel SIGA method, referred to as SDD-mIGA, to solve stochastic BVPs from linear elasticity on arbitrary multipatch domains. This is done by exploiting analysis-suitable T-splines in IGA and orthonormalized measure-consistent B-splines in SDD, within an integrated framework.

In this paper, a novel stochastic method is proposed for solving a general stochastic BVP from linear elasticity on arbitrary multipatch domains by integrating SDD and IGA. This work mainly focuses on practical applications, and readers interested in mathematical aspects and more details of SDD are referred to the companion paper [31].[1] The paper is organized as follows. Section 2 entails basic preliminary definitions and assumptions on input random variables and random field discretization by Karhunen–Loève expansion. IGA on multipatch domains is discussed in Section 3, where the notations for analysis-suitable T-splines are presented and the patch coupling technique used is briefly discussed. In Section 4 the stochastic BVP from linear elasticity is officially defined and a Galerkin, finite dimensional IGA formulation of the problem is developed. Section 5 entails the SDD formulation and standard least squares (SLS) regression method to efficiently estimate the SDD expansion coefficients. Analytical formulae are also derived for the computation of output statistical moments. Three numerical examples, including a 54-dimensional stochastic problem, are presented in Section 6, to illustrate the accuracy and efficiency of the proposed method, before the conclusions are drawn in Section 7. The Bézier extraction operator is briefly presented in the Appendix for the paper to be self-contained.

2. Input random variables and random fields

Let $\mathbb{N} := \{1, 2, \ldots\}$, $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$, $\mathbb{R} := (-\infty, +\infty)$, and $\mathbb{R}^+ := (0, +\infty)$ represent the sets of positive integer, non-negative integer, real, and positive real numbers, respectively. Moreover, let the closed and bounded $\mathcal{D} \subset \mathbb{R}^d$ represent a physical domain of dimension $d = 1, 2,$ or 3.

2.1. Input random variables

Define a closed finite interval $\mathbb{A}^{\{k\}} := [a_k, b_k] \subset \mathbb{R}$ for $N \in \mathbb{N}$ and $k = 1, \ldots, N$, where $a_k, b_k \in \mathbb{R}$ and $b_k > a_k$. An $N-$dimensional bounded stochastic domain is therefore defined using the tensor-product operator " $\times$ " as $\mathbb{A}^N := \times_{k=1}^N \mathbb{A}^{\{k\}} = \times_{k=1}^N [a_k, b_k]$.

Define a complete probability space by $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω, $\mathcal{F}$, and $\mathbb{P} : \mathcal{F} \to [0, 1]$ are sample space, σ-algebra, and probability measure, respectively. Let $\mathcal{B}^N := \mathcal{B}(\mathbb{A}^N)$ represent the Borel σ-algebra on $\mathbb{A}^N \subseteq \mathbb{R}^N$ and consider an $\mathbb{A}^N$-valued input random vector $\mathbf{X} := (X_1, \ldots, X_N)^\mathsf{T} : (\Omega, \mathcal{F}) \to (\mathbb{A}^N, \mathcal{B}^N)$ as the collection of a total of N random variables $\{X_i\}_{i=1}^N$, which carry the information about the input statistical uncertainties. Vector $\mathbf{X}$ is N-dimensional; therefore, the corresponding stochastic problem is commonly referred to be of dimension N. Similarly, define each random variable X_k on its associated abstract marginal probability space $(\Omega^{\{k\}}, \mathcal{F}^{\{k\}}, \mathbb{P}^{\{k\}})$. A joint distribution function of $\mathbf{X}$ is then denoted by $F_{\mathbf{X}}(\mathbf{x}) := \mathbb{P}(\cap_{k=1}^N \{X_k \leq x_k\})$, for which a joint probability density function (PDF), if it exists, is written as $f_{\mathbf{X}}(\mathbf{x}) := \partial^N F_{\mathbf{X}}(\mathbf{x})/\partial x_1 \cdots \partial x_N$. For instance, for a cylinder of random inner radius r_i, outer radius r_o, Young's modulus E, and Poisson's ratio ν under random internal pressure P with associated statistical distributions, there is a total of five random variables ($N = 5$) and $\mathbf{X} = \{r_i, r_o, E, \nu, P\}$.

---

[1] The PDF file to Ref. [31] is available at http://user.engineering.uiowa.edu/~rahman/sdd.pdf.

2.2. Random fields

In many engineering applications, there is spatial variability of a random input parameter, referred to as a random field. In other words, some characteristic parameter of the stochastic problem may not only vary from sample to sample, but also from point to point within each sample. For example, the thickness and Young's modulus of a cylinder can be considered as random fields. In computational analysis, however, since there are infinitely many points in the domain, there is a need for random field discretization, thereby expressing the random field by a finite number of constituent uncorrelated random variables. An illustrious approach for random field discretization is the Karhunen–Loève (K–L) expansion [34], which yields an infinite series expansion of a given random field in terms of uncorrelated random variables and deterministic functions with a spatial argument. The K–L method has an error-minimizing property as well.

2.2.1. Random field discretization

Let $\mathbf{z} \in \mathcal{D}$ denote a spatial point in the physical domain of interest and define a real-valued random field $\alpha(\mathbf{z}, \cdot)$ on $\mathcal{D} \in \mathbb{R}^d$ with *zero* mean and a square-integrable covariance function $\Gamma : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$. By solving the Fredholm integral equation

$$\int_{\mathcal{D}} \Gamma(\mathbf{z}, \mathbf{z}')\phi(\mathbf{z}')d\mathbf{z}' = \lambda\phi(\mathbf{z}), \tag{1}$$

an infinite sequence of eigenpairs $\{\lambda_i, \phi_i(\mathbf{z})\}_{i \in \mathbb{N}}$ is obtained, yielding

$$\alpha(\mathbf{z}, \cdot) \sim \sum_{i=1}^{\infty} \sqrt{\lambda_i}\phi_i(\mathbf{z})X_i, \tag{2}$$

which is convergent in the mean-square sense to the correct limit. Note that in (2), $\{X_i\}_{i \in \mathbb{N}}$ is an infinite sequence of *zero*-mean, standardized, and uncorrelated random variables.

2.2.2. Karhunen-Loève approximation

In practice, the infinite series in (2) must be truncated. Therefore, an N'-term truncation or K–L approximation is written as

$$\alpha_{N'}(\mathbf{z}, \cdot) = \sum_{i=1}^{N'} \sqrt{\lambda_i}\phi_i(\mathbf{z})X_i \tag{3}$$

with the eigenvalues arranged in a descending order in terms of magnitude. Hence, $\alpha_{N'}(\mathbf{z}, \cdot)$ is used in the subsequent stochastic analysis instead of $\alpha(\mathbf{z}, \cdot)$.

Note that truncating (2) at a relatively small N' yields a smaller problem dimension and, by extension, less computational burden in the subsequent stochastic analysis. On the other hand, a low accuracy is generally expected in such a K–L approximation. Essentially, the truncation parameter N' must be chosen judiciously while keeping an eye on the complexity of the subsequent stochastic analysis. The random variables $X_i, \ i = 1, \ldots, N'$ should be added to vector $\mathbf{X}$, described in Section 2.1. In other words, whether or not a random field exists in a stochastic problem, N represents the total number of random variables.

Various methods have been proposed for accurate and efficient numerical solution of (1). IGA methods have proved to be accurate and convenient with Galerkin and collocation formulations for random field discretization on single-patch [35,36] and arbitrary multipatch domains [37]. While the Galerkin formulations are typically more computationally intensive, the collocation-based methods deliver a satisfactory trade-off between accuracy and efficiency when approximating the eigensolutions by B-splines, NURBS, or analysis-suitable T-splines. Moreover, a recent work [38] entails a matrix-free isogeometric Galerkin method using tensor-product splines to further boost the efficiency in random field discretization by the K–L expansion. Readers interested in more details are referred to the aforementioned works.

2.3. Assumptions

The input random vector $\mathbf{X} := (X_1, \ldots, X_N)^{\mathsf{T}} : (\Omega, \mathcal{F}) \to (\mathbb{A}^N, \mathcal{B}^N)$ satisfies all of the following conditions to be implementable in the SDD framework:

(a) Each input random variable $X_k : (\Omega^{\{k\}}, \mathcal{F}^{\{k\}}) \to (\mathbb{A}^{\{k\}}, \mathcal{B}^{\{k\}})$ with a bounded support $\mathbb{A}^{\{k\}} \subset \mathbb{R}$ has absolutely continuous marginal distribution function $F_{X_k}(x_k) := \mathbb{P}(X_k \leq x_k)$ and continuous marginal density function $f_{X_k}(x_k) := \partial F_{X_k}(x_k)/\partial x_k$.

(b) All random variables X_k, $k = 1, \ldots, N$, are statistically independent, but not necessarily identical. In other words, $f_{\mathbf{X}}(\mathbf{x}) = \prod_{k=1}^{N} f_{X_k}(x_k)$.

(c) Given a non-negative integer $p_k \in \mathbb{N}_0$, each input random variable X_k has finite moments of all orders up to $2p_k$. In mathematical form, for all $k = 1, \ldots, N$ and $0 \leq l \leq 2p_k$,

$$\mathbb{E}\left[X_k^l\right] := \int_{\Omega} X_k^l(\omega) d\mathbb{P}(\omega) = \int_{\mathbb{A}^{\{k\}}} x_k^l f_{X_k}(x_k) dx_k < \infty,$$

where $\mathbb{E}$ is referred to as the expectation operator with respect to the probability measure $\mathbb{P}$.

For a given input probability measure, these assumptions ensure the existence of a relevant sequence of orthogonal splines. Furthermore, for the SDD formulation to be valid, the probability measures of input random variables must be constrained to closed and bounded domains. However, the SDD formulation is able to handle the uncertainties stemming from unbounded statistical distributions as well, such as normal or lognormal probability measures, if the associated measures are first properly transformed to others defined on closed and bounded domains. This will be re-visited in the numerical examples section. The discrete distributions and dependent random variables are not considered in this work.

3. Isogeometric analysis

IGA provides a powerful framework for accurate approximation of the solution to PDEs in various fields of engineering [9,10] by exploiting B-splines, NURBS, T-splines, and other smooth basis functions. Essentially, IGA employs identical basis functions for CAD modeling and computational analysis. Although IGA precisely models common geometries, many analysis domains are so complex that they must be modeled in multiple patches. T-splines are common in modeling multipatch geometries due to their ability in water-tight modeling and enabling local mesh refinement. This section briefly introduces T-splines by starting from two-dimensional domains. Readers are referred to related works [12,15,21] for further details.

3.1. T-splines

T-splines were first introduced for water-tight modeling in CAD and for tackling the issues with gaps and overlaps on the patch interfaces [11] before IGA even officially made a debut. Through local mesh refinement, T-junctions are allowed in the T-mesh tiling in the parametric space. In what follows, the paraphernalia of T-splines is described simply for two-dimensional domains, which can be generalized for three-dimensional cases.

Denote by p_1 and $p_2 \in \mathbb{N}_0$ the polynomial orders in the two parametric coordinate directions 1 and 2, respectively. For $n_c \in \mathbb{N}$ number of control points denoted by $\mathbf{c}_i$, $i = 1, \ldots, n_c$ in the physical domain $\mathcal{D}$, there is the same number of anchors, denoted by $\hat{\mathbf{c}}_i$, $i = 1, \ldots, n_c$ in the parametric domain $\hat{\mathcal{D}}$ with a one-to-one correspondence. Two T-meshes for even degree $p_k = 2$, $k = 1, 2$ are indicated in parametric and physical domains in Figs. 1(a) and 1(b), respectively. The anchors are illustrated by black closed circles, and the black lines represent the element edges. The shaded area in Fig. 1(b) demonstrates a nonzero area that covers the physical domain in Fig. 1(a). Let n_e denote the total number of elements, where each element $e = 1, \ldots, n_e$ can be transformed to the parent space $[-1, 1]^2$ for Gauss integration.

For each anchor in a T-mesh, there are d number of local knot vectors associated with the corresponding parametric coordinate directions ξ_k, $k = 1, \ldots, d$. In each direction, a local knot vector is formed by taking an anchor and marching horizontally or vertically. For the T-meshes with an even polynomial degree, marching stops when $(p_k/2 + 1)$ number of edges is crossed on either side. If there are no more edges to cross, the most recently placed knot is repeated. Eventually, there are $(p_k + 2)$ entries in the local knot vectors. In mathematical form, for an anchor i in a two-dimensional T-mesh,

$$\boldsymbol{\xi}_k^i = \{\xi_{k,1}^i, \xi_{k,2}^i, \ldots, \xi_{k,p_k+2}^i\}, \quad k = 1, 2, \text{ and } i = 1, \ldots, n_c,$$

where $\xi_{k,1}^i \leq \xi_{k,2}^i \leq \cdots \leq \xi_{k,p_k+2}^i$. As an example, the two local knot vectors for Anchor A in Fig. 1(b) are $\boldsymbol{\xi}_1^A = \{0.2, 0.4, 1, 1\}$ and $\boldsymbol{\xi}_2^A = \{0, 0.2, 1, 1\}$. For each set of local knot vectors associated with an anchor, the knot
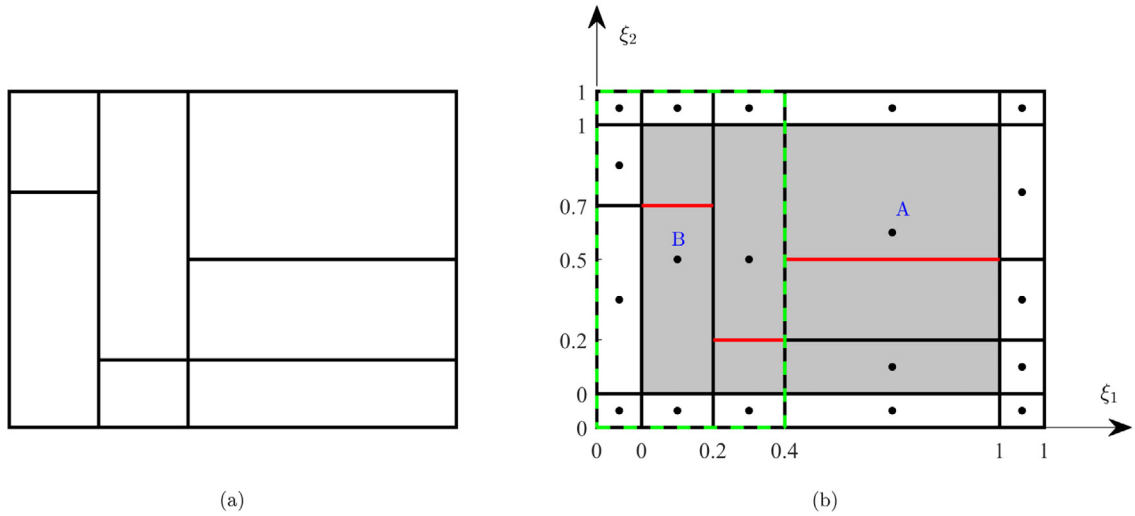
Fig. 1. A T-mesh: (a) physical domain; (b) parametric domain. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

lines are drawn by the tensor-product grid constructed by the knots. If a knot line does not coincide with an existing element edge, a continuity reduction line is formed, as shown in red in Fig. 1(b). Continuity reduction lines break the existing elements into smaller elements [21].

For each and every anchor i, $i = 1, \ldots, n_c$ there is one associated T-spline blending function. T-splines are constructed recursively, starting from order *zero* as

$$U^{i,k}_{i_k,0,\boldsymbol{\xi}_k}(\xi_k) = \begin{cases} 1, & \xi^i_{k,i_k} \leq \xi_k < \xi^i_{k,i_k+1}, \\ 0, & \text{otherwise}, \end{cases} \quad i_k = 1, \ldots, p_k + 1,$$

and for $p_k \geq 1$, a T-spline blending function is constructed by the Cox–de Boor formula [39] as

$$U^{i,k}_{i_k,p_k,\boldsymbol{\xi}_k}(\xi_k) = \frac{\xi_k - \xi^i_{k,i_k}}{\xi^i_{k,i_k+p_k} - \xi^i_{k,i_k}} U^{i,k}_{i_k,p_k-1,\boldsymbol{\xi}_k}(\xi_k) + \frac{\xi^i_{k,i_k+p_k+1} - \xi_k}{\xi^i_{k,i_k+p_k+1} - \xi^i_{k,i_k+1}} U^{i,k}_{i_k+1,p_k-1,\boldsymbol{\xi}_k}(\xi_k).$$

Eventually, a single univariate T-spline function denoted by $U^{i,k}_{1,p_k,\boldsymbol{\xi}_k}(\xi_k)$, which is associated with anchor i, parametric direction k, polynomial degree p_k, and local knot vector $\boldsymbol{\xi}_k$, is obtained.

The multivariate T-spline functions are constructed by the product operator as

$$\mathrm{U}^i_{\mathbf{p},\boldsymbol{\Xi}^i}(\boldsymbol{\xi}) := \prod_{k=1}^d U^{i,k}_{1,p_k,\boldsymbol{\xi}_k}(\xi_k), \; i = 1, \ldots, n_c,$$

where $\mathbf{p} := (p_1, \ldots, p_d) \in \mathbb{N}_0^d$, $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_d)$, and $\boldsymbol{\Xi}^i := (\boldsymbol{\xi}^i_1, \ldots, \boldsymbol{\xi}^i_d)$. The function $\mathrm{U}^i_{\mathbf{p},\boldsymbol{\Xi}^i}(\boldsymbol{\xi})$ is, in general, only nonzero over a chunk of the domain of analysis. For instance, the green rectangle in Fig. 1(b) illustrates region $[0, 0.4] \times [0, 1]$, over which the multivariate T-spline associated with Anchor B is nonzero. Furthermore, T-spline blending functions, in contrast to NURBS, are formed for each anchor i, $i = 1, \ldots, n_c$ independently, which ensures much more flexibility in mesh refinement. As a drawback, however, the T-splines defined so far are not necessarily linearly independent — hence the name "blending function". The linear dependence of these functions causes instability in the computational analyses, to say the least [15], which would not have been an issue if NURBS were employed. Therefore, only analysis-suitable T-splines are acceptable to proceed with in the numerical computations. This will be re-visited in Section 3.3.

3.2. Rational T-splines

In IGA, the control points representing an object in the physical domain, each have an associated weight $w_i \in \mathbb{R}^+$, $i = 1, \ldots, n_c$. For CAD modeling and computational analysis on more complex geometries, such as circles, cylinders, and spheres, the T-splines are rationalized as

$$\mathrm{T}^i_{\mathbf{p}, \boldsymbol{\Xi}^i}(\boldsymbol{\xi}) := \frac{w_i \mathrm{U}^i_{\mathbf{p}, \boldsymbol{\Xi}^i}(\boldsymbol{\xi})}{\displaystyle\sum_{j=1}^{n_c} w_j \mathrm{U}^j_{\mathbf{p}, \boldsymbol{\Xi}^j}(\boldsymbol{\xi})}, \tag{4}$$

where $\mathrm{T}^i_{\mathbf{p}, \boldsymbol{\Xi}^i} : \hat{\mathcal{D}} \to \mathbb{R}$ is defined as a multivariate rational T-spline. The rational T-splines, simply referred to as T-splines from this point forth, satisfy the partition of unity property, which is essential in finite element methods. Hence, a line ($d = 1$), surface ($d = 2$), or volume ($d = 3$) can be expressed by the linear projection $\tilde{\mathrm{H}}(\boldsymbol{\xi})$, defined as

$$\tilde{\mathrm{H}}(\boldsymbol{\xi}) := \sum_{i=1}^{n_c} \mathrm{T}^i_{\mathbf{p}, \boldsymbol{\Xi}^i}(\boldsymbol{\xi}) \mathbf{c}_i.$$

Recall that $\mathbf{c}_i$ is a control point in the physical domain and $\boldsymbol{\xi}$ represents a point in the parametric space. Assuming that the mapping is invertible,

$$\bar{\mathrm{T}}^i_{\mathbf{p}, \boldsymbol{\Xi}^i}(\mathbf{z}) := \mathrm{T}^i_{\mathbf{p}, \boldsymbol{\Xi}^i}(\boldsymbol{\xi}) \circ \tilde{\mathrm{H}}^{-1} \tag{5}$$

is defined as the multivariate T-spline blending functions corresponding to anchor i, $i = 1, \ldots, n_c$ in the physical space. Here, $\mathbf{z}$ refers to a point in the physical domain.

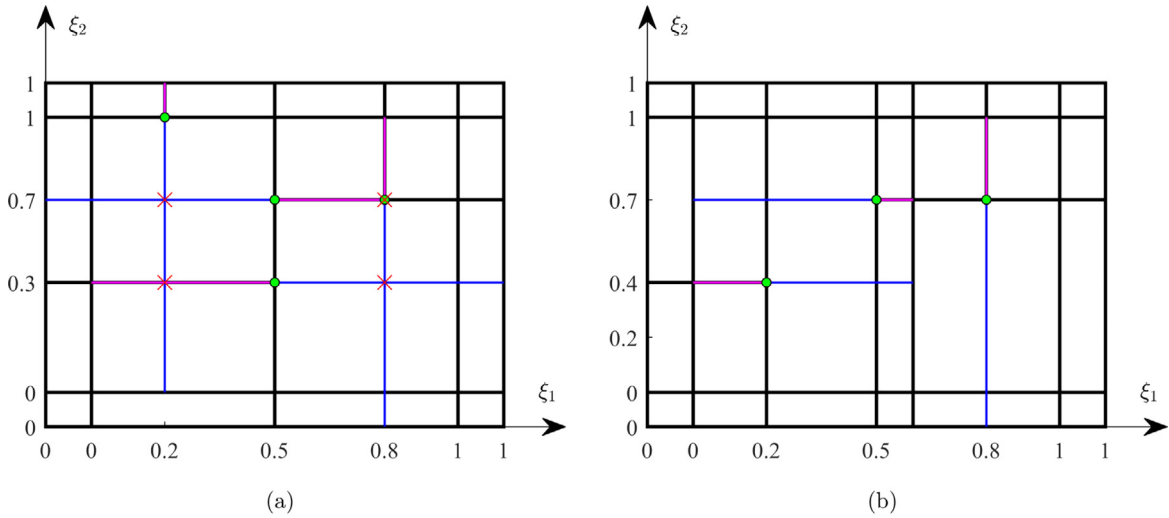## 3.3. Analysis-suitable T-meshes

As discussed in Section 3.1, T-splines deliver flexibility to the discretization process by introducing T-junctions. This is mainly because of the nature of point-based splines [11], which stems from using local knot vectors. However, Buffa et al. [14] were the first to realize that not every T-mesh is suitable for analysis because the underlying T-splines may not necessarily be linearly independent. This, as a downside of using T-splines, continued to be studied until Li et al. [15] developed mathematical and topological ways to understand this phenomenon and assess whether a given T-mesh is analysis-suitable or not.

The topological approach of evaluating the analysis-suitability of a T-mesh is very straightforward. Define a T-node as a T-junction inside the parametric domain, and for every T-junction in the T-mesh, draw T-node extension lines, as indicated in Fig. 2, as an example. The T-nodes are indicated by closed green circles. Here, $p_k = 2$, $k = 1, 2$. There are two types of extension lines, namely face extension lines and edge extension lines [16]. The former are drawn in the direction where there is no edge, and the latter are emitted in the opposite direction. In Fig. 2, the two types of lines are demonstrated in blue and magenta, respectively. Simply, a T-mesh is called analysis-suitable if no T-node extensions intersect. Figs. 2(a) and 2(b) are examples of non-analysis-suitable and analysis-suitable T-meshes, respectively. The intersections are marked with red crosses. Note that the analysis-suitable T-splines are those associated with an analysis-suitable T-mesh [15].

By adopting the topological approach, drawing an analysis-suitable T-mesh may become tedious, as one may have to remove and add edges repeatedly and check to avoid T-node extension intersections. Having said so, hierarchical T-splines [18] and adaptive refinement methods [40] have been developed to avoid such complexities in a rather more systematic manner.

3.4. T-mesh refinement with Bézier extraction operator

The Bézier extraction operator, discussed in the Appendix, provides a simple means for local mesh refinement. Let $\mathcal{T}_0$ and $\mathcal{T}_R$ represent the coarse mesh and refined mesh, respectively. In mathematical form, $\mathcal{T}_0 \subseteq \mathcal{T}_R$. Moreover, denote by $\mathbf{U}_0(\boldsymbol{\xi})$ and $\mathbf{U}_R(\boldsymbol{\xi})$ the T-splines corresponding to $\mathcal{T}_0$ and $\mathcal{T}_R$, respectively, as defined in (A.2), with $n_{c,0}$ and $n_{c,R}$ being the total number of control points (or T-splines) associated with $\mathcal{T}_0$ and $\mathcal{T}_R$, respectively. The idea is

Fig. 2. Two T-mesh examples: (a) non-analysis-suitable T-mesh; (b) analysis-suitable T-mesh. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

that the representation of a T-spline object should remain the same after refinement. Therefore, express the T-splines of the coarse mesh in terms of those of the refined mesh by

$$\mathbf{U}_0(\boldsymbol{\xi}) = \mathbf{T}\mathbf{U}_R(\boldsymbol{\xi}). \tag{6}$$

The transformation matrix $\mathbf{T}$ is of dimension $n_{c,0} \times n_{c,R}$ and is referred to as the refinement matrix. Hence, let $\mathbf{C}_0$ and $\mathbf{C}_R$ be the Bézier extraction operators associated with the meshes $\mathcal{T}_0$ and $\mathcal{T}_R$, respectively, and by substituting (A.2) in (6), write

$$\mathbf{C}_0 = \mathbf{T}\mathbf{C}_R. \tag{7}$$

By matrix operations, (7) can be formulated for each anchor i, $i = 1, \ldots, n_{c,0}$ as

$$\mathbf{C}_0^i = \mathbf{C}_R{}^\mathsf{T}\mathbf{T}^i, \tag{8}$$

with $\mathbf{T}^i$ being the ith submatrix in $\mathbf{T}$ as

$$\mathbf{T} = [\mathbf{T}^1, \ldots, \mathbf{T}^{n_{c,0}}]^\mathsf{T}.$$

By solving (8) for $\mathbf{T}^i$, $i = 1, \ldots, n_{c,0}$, the refinement matrix $\mathbf{T}$ is obtained and the linear transformation in (6) is at hand.

The weighted control points for $\mathcal{T}_R$ are defined as $\mathbf{c}_R^w := [\mathbf{c}_{R,w}^1 \ \cdots \ \mathbf{c}_{R,w}^{n_{c,R}}]^\mathsf{T}$ and can be computed using the weighted control points of the coarse mesh $\mathbf{c}_0^w := [\mathbf{c}_{0,w}^1 \ \cdots \ \mathbf{c}_{0,w}^{n_{c,0}}]^\mathsf{T}$ by the projection [21]

$$\mathbf{c}_R^w = \mathbf{T}^\mathsf{T}\mathbf{c}_0^w,$$

where

$$\mathbf{c}_{0,w}^i := \{w_i c_1^i, \ldots, w_i c_d^i\}, \ i = 1, \ldots, n_{c,0},$$

and c_d^i is the dth coordinate of the ith control point in the physical space $\mathcal{D}$, and w_i is its associated weight.

Note that in order to be able to analytically calculate the refinement matrix $\mathbf{T}$, one must be able to write the T-splines of the coarse mesh in terms of those of the refined mesh. In technical language, the two sets of T-spline functions must be nested [21]. Otherwise, there is no unique solution for $\mathbf{T}^i$ in (8). In this work, the Bézier extraction operator is employed for T-mesh refinement, the basis functions of the coarse and refined mesh are nested, and all the T-meshes are analysis-suitable.
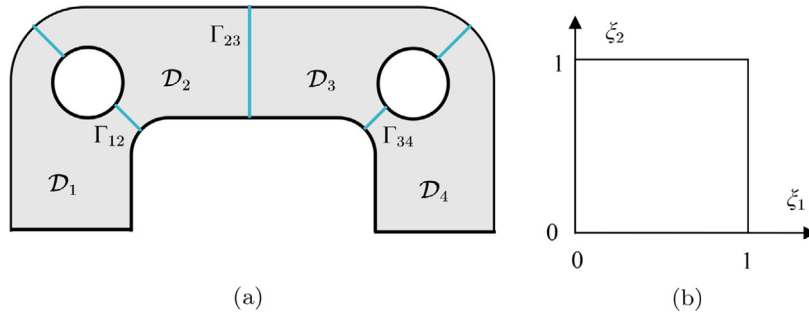
Fig. 3. A decomposition of a two-dimensional geometry: (a) four-patch domain in the physical space; (b) parametric domain for mapping to an arbitrary patch. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.5. Patch-coupling methods

For many common, relatively simple domains, there is no need for multiple patches, and a single T-mesh suffices for representing the entire geometry. In some cases, however, the mapping of a domain from parametric to physical space is so complex that it might make the numerical computations inaccurate. For CAD modeling and computational analysis on arbitrary complex domains, subdividing the domain into smaller patches is, therefore, desirable for convenience and accuracy.

Note that, in this work, a multipatch domain is specifically a region that is partitioned into a finite number of single patches. As defined previously, denote by $\mathcal{D} \subset \mathbb{R}^d$ a closed bounded domain of a physical object and let $n_p \in \mathbb{N}$ be the total number of patches with open bounded domains $\mathcal{D}_q \subseteq \mathcal{D}$, $q = 1, \ldots, n_p$, such that

$$\bar{\mathcal{D}} = \mathcal{D} = \bigcup_{q=1}^{n_p} \bar{\mathcal{D}}_q,$$

where the overline represents set closure. Moreover, consider two distinct adjacent patches q and q', $q \neq q'$ with associated domains $\mathcal{D}_q$ and $\mathcal{D}_{q'}$. The two patches may overlap only at their interface $\Gamma_{qq'}$ (say), that is,

$$\bar{\mathcal{D}}_q \cap \bar{\mathcal{D}}_{q'} = \Gamma_{qq'} \neq \emptyset.$$

Fig. 3(a) illustrates a four-patch geometry with domains $\mathcal{D}_1$ through $\mathcal{D}_4$ and three interfaces Γ_{12}, Γ_{23}, and Γ_{34}, which are shown by cyan lines. Each patch is an image of the rectangular domain in Fig. 3(b) to the physical domain. The solutions of interest can be mapped to the physical domain by initiating the numerical analyses from the rectangular domains in the parametric space.

Clearly, multiple patches are required to model many complex geometries, and the arrangement of the patches in the domain may be of importance, depending on the problem. Considering how the patches are arranged, the formulation of multipatch IGA may then become challenging. That is why research is ongoing to develop powerful coupling methods [41]. However, regardless of the patch shapes including unstructured and highly structured conforming patches, the C⁰-continuity constraints are straightforward to impose in IGA to couple them. This is accomplished as follows.

Let Figs. 4(a) and 4(b) demonstrate two analysis-suitable T-meshes, which represent two distinct adjacent patches, named as Patch 1 and Patch 2, respectively. Bazilevs et al. [12] described how such two patches are coupled in such a way to yield C⁰-continuity at the patch interface. Fig. 4(c) illustrates the coupled T-meshes, where C⁰-continuity constraints are imposed in the region shown by a cyan rectangle. In this region, there essentially must be common edges and anchors in the two adjacent T-meshes, which are demonstrated by black lines and closed black circles, respectively. Note that the resulting T-mesh in Fig. 4(c) must be double-checked for analysis-suitability according to Section 3.3. Moreover, the knots have been normalized to zero to one after coupling, which is optional. Due to the flexible nature of T-splines, T-junctions can be incorporated in such a way as to effectively minimize the number of elements in the subsequent computational analysis, and still yield proper local mesh refinement over particular regions of the domain. Fig. 4(d) depicts the associated T-mesh tiling in the physical domain, where the patch interface with the C⁰-continuity is shown by a cyan line.
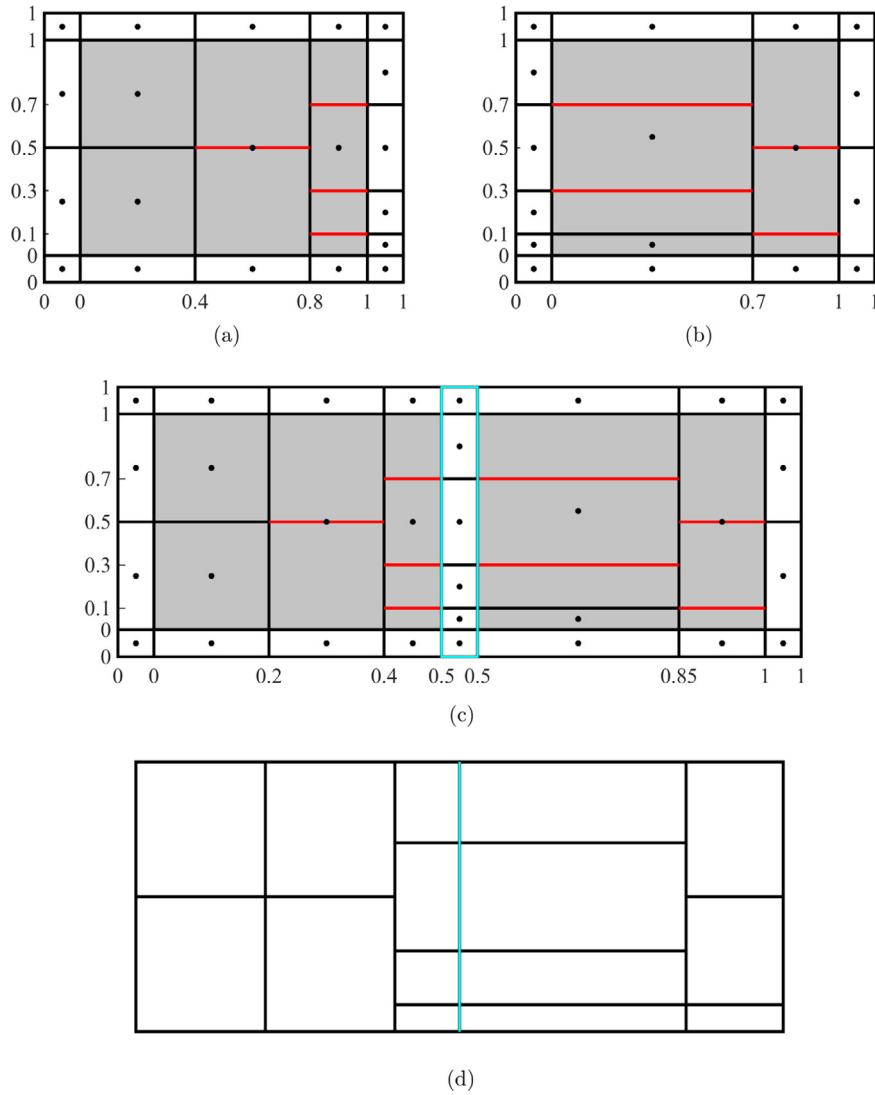
Fig. 4. A T-mesh for a two-patch domain: (a) T-mesh for Patch 1; (b) T-mesh for an adjacent Patch 2; (c) patch coupling with C^0-continuity; (d) T-mesh tiling of the physical domain. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

For many practical applications, the C^0-continuity is adequate and may be preferable due its simplicity of implementation, although it clearly is not the best option in terms of accuracy. In this work, only C^0-continuity conditions are considered on all patch interfaces.

4. Stochastic boundary-value problem

This section entails official definition of the stochastic BVP from linear elasticity. In this problem, the domain $\mathcal{D} \subset \mathbb{R}^d$ represents the material under equilibrium. Generally, there is statistical variability in the loads, geometry, and material properties, while the goal is to quantify the uncertainties in some response variables of interest.

4.1. Stochastic PDE

The information about statistical variability of the input is carried by the random vector $\mathbf{X}$, as defined in Section 2.1. In order to solve the stochastic PDE, the goal is to find the displacement vector $\mathbf{u}(\mathbf{z}; \mathbf{X})$ and stress

vector $\boldsymbol{\sigma}(\mathbf{z}; \mathbf{X})$, which satisfy

$$
\begin{aligned}
\nabla \cdot \boldsymbol{\sigma}(\mathbf{z}; \mathbf{X}) + \mathbf{b}(\mathbf{z}; \mathbf{X}) &= \mathbf{0} && \text{in } \mathcal{D}(\mathbf{X}) \subset \mathbb{R}^d, \\
\boldsymbol{\sigma}(\mathbf{z}; \mathbf{X}) \cdot \mathbf{n}(\mathbf{z}; \mathbf{X}) &= \bar{\mathbf{t}}(\mathbf{z}; \mathbf{X}) && \text{on } \Gamma_t(\mathbf{X}) \subset \partial\mathcal{D}(\mathbf{X}), \\
\mathbf{u}(\mathbf{z}; \mathbf{X}) &= \bar{\mathbf{u}}(\mathbf{z}; \mathbf{X}) && \text{on } \Gamma_u(\mathbf{X}) \subset \partial\mathcal{D}(\mathbf{X}),
\end{aligned}
\tag{9}
$$

at point $\mathbf{z} = (z_1, \ldots, z_d) \in \mathcal{D}(\mathbf{X}) \subset \mathbb{R}^d$ such that

$$
\Gamma_t(\mathbf{X}) \cup \Gamma_u(\mathbf{X}) \cup \Gamma_0(\mathbf{X}) = \partial\mathcal{D}(\mathbf{X}), \;\; \Gamma_t(\mathbf{X}) \cap \Gamma_u(\mathbf{X}) = \emptyset,
$$

where $\Gamma_0(\mathbf{X})$ is the free boundary. In (9), $\nabla := (\partial/\partial z_1, \ldots, \partial/\partial z_d)$ contains the gradients, $\mathbf{b}(\mathbf{z}; \mathbf{X})$ is the body force, $\mathbf{n}(\mathbf{z}; \mathbf{X})$ is an unit outward normal, $\mathbf{t}(\mathbf{z}; \mathbf{X})$ is the prescribed traction, and $\bar{\mathbf{u}}(\mathbf{z}; \mathbf{X})$ is the prescribed displacement.

The stress vector $\boldsymbol{\sigma}(\mathbf{z}; \mathbf{X})$ is computed by $\mathbf{D}(\mathbf{z}; \mathbf{X}) : \boldsymbol{\epsilon}(\mathbf{z}; \mathbf{X})$, where $\mathbf{D}(\mathbf{z}; \mathbf{X})$ is the elasticity tensor and $\boldsymbol{\epsilon}(\mathbf{z}; \mathbf{X}) := (1/2)(\nabla + \nabla^\top)\mathbf{u}(\mathbf{z}; \mathbf{X})$ is the strain vector. Furthermore, ":" symbolizes the tensor contraction operator.

The remainder of Section 4 concisely summarizes the method for solving the BVP defined in (9). Readers in search of more details are directed to the authors' previous work [33].

4.2. Finite dimensional formulation

The finite dimensional formulation of the stochastic BVP starts from converting the problem to its weak form. Let $L^2(\mathcal{D}(\mathbf{X}))$ represent the collection of all square-integrable functions $\mathbf{u} : \mathcal{D} \times \Omega \to \mathbb{R}^d$. Define the vector space

$$
\mathcal{U} := \left\{ \mathbf{u}(\mathbf{z}; \mathbf{X}) : \mathbf{u}(\mathbf{z}; \mathbf{X}) \in H^1(\mathcal{D}(\boldsymbol{\mathcal{X}})), \mathbf{u}(\mathbf{z}; \mathbf{X})\,|_{\Gamma_u} = \bar{\mathbf{u}}(\mathbf{z}; \mathbf{X}) \right\},
$$

where the solution of the weak form lives. Here, $H^1(\mathcal{D}(\boldsymbol{\mathcal{X}}))$ is the Sobolev space. The vector space of weighting functions is also defined as

$$
\mathcal{W} := \left\{ \mathbf{w}(\mathbf{z}; \mathbf{X}) : \mathbf{w}(\mathbf{z}; \mathbf{X}) \in H^1(\mathcal{D}(\boldsymbol{\mathcal{X}})), \mathbf{w}(\mathbf{z}; \mathbf{X})\,|_{\Gamma_u} = 0 \right\}.
$$

Therefore, the weak form is written as [10]

$$
\mathcal{A}(\mathbf{u}(\mathbf{z}; \mathbf{X}), \mathbf{w}(\mathbf{z}; \mathbf{X})) = \mathcal{L}(\mathbf{w}(\mathbf{z}; \mathbf{X})),
\tag{10}
$$

where

$$
\mathcal{A}(\mathbf{u}(\mathbf{z}; \mathbf{X}), \mathbf{w}(\mathbf{z}; \mathbf{X})) = \int_{\mathcal{D}(\boldsymbol{\mathcal{X}})} \boldsymbol{\epsilon}^\top(\mathbf{u}(\mathbf{z}; \mathbf{X}))\mathbf{D}(\mathbf{z}; \mathbf{X})\boldsymbol{\epsilon}(\mathbf{w}(\mathbf{z}; \mathbf{X}))\, d\mathbf{z}
$$

and

$$
\mathcal{L}(\mathbf{w}(\mathbf{z}; \mathbf{X})) = \int_{\mathcal{D}(\mathbf{X})} \mathbf{w}^\top(\mathbf{z}; \mathbf{X})\mathbf{b}(\mathbf{z}; \mathbf{X})\, d\mathbf{z} + \int_{\Gamma_t(\mathbf{X})} \mathbf{w}^\top(\mathbf{z}; \mathbf{X})\, \bar{\mathbf{t}}(\mathbf{z}; \mathbf{X})\, d\Gamma_t.
$$

The weak form in (10) and the strong form in (9) are analytically equivalent [42], if a set of regulatory conditions is satisfied.

4.2.1. Galerkin discretization

The Galerkin method uses the vector subspaces $\mathcal{U}^h \subset \mathcal{U}$ and $\mathcal{W}^h \subset \mathcal{W}$ as approximation spaces and expresses a solution of interest in terms of a linear combination of a set of linearly independent basis functions residing in $\mathcal{U}^h$ and $\mathcal{W}^h$. Typically, identical types of basis functions are used to form $\mathcal{U}^h$ and $\mathcal{W}^h$. Consequently, for all $\mathbf{w}^h \subset \mathcal{W}^h$,

$$
\mathcal{A}(\mathbf{v}^h(\mathbf{z}; \mathbf{X}), \mathbf{w}^h(\mathbf{z}; \mathbf{X})) = \mathcal{L}(\mathbf{w}^h(\mathbf{z}; \mathbf{X})) - \mathcal{A}(\mathbf{w}^h(\mathbf{z}; \mathbf{X}), \bar{\mathbf{u}}^h(\mathbf{z}; \mathbf{X})),
\tag{11}
$$

assuming that for a given function $\bar{\mathbf{u}}^h(\mathbf{z}; \mathbf{X}) \in \mathcal{U}^h$ and $\bar{\mathbf{u}}^h(\mathbf{z}; \mathbf{X})\,|_{\Gamma_u} = \bar{\mathbf{u}}(\mathbf{z}; \mathbf{X})$, for every $\mathbf{u}^h(\mathbf{z}; \mathbf{X}) \in \mathcal{U}^h$, there exists a unique $\mathbf{v}^h(\mathbf{z}; \mathbf{X}) \in \mathcal{W}^h$ such that

$$
\mathbf{u}^h(\mathbf{z}; \mathbf{X}) = \mathbf{v}^h(\mathbf{z}; \mathbf{X}) + \bar{\mathbf{u}}^h(\mathbf{z}; \mathbf{X}).
$$

The Galerkin approach can be employed to solve stochastic PDEs by finite element methods [43].

4.2.2. Matrix formulation by T-splines

In this work, T-splines are employed for both geometrical modeling and stress analysis. For matrix formulation of the BVP by (11), the same sets of T-spline basis functions are employed from $\mathcal{U}^h$ and $\mathcal{W}^h$, which have been defined in (4) and (5) associated with the parametric and physical spaces, respectively.

The matrix formulation of the stochastic BVP in its weak form yields

$$\mathbf{K}(\mathbf{X})\mathbf{d}(\mathbf{X}) = \mathbf{F}(\mathbf{X}),$$

where

$$
\begin{aligned}
K_{ij}(\mathbf{X}) &= \mathcal{A}\Big((\bar{\mathrm{T}}_{\mathbf{p},\boldsymbol{\Xi}^{k_1(i,d)}}^{k_1(i,d)}(\mathbf{z}); \mathbf{X}), (\bar{\mathrm{T}}_{\mathbf{p},\boldsymbol{\Xi}^{k_2(j,d)}}^{k_2(j,d)}(\mathbf{z}); \mathbf{X})\Big), \\
F_i(\mathbf{X}) &= \mathcal{L}\Big((\bar{\mathrm{T}}_{\mathbf{p},\boldsymbol{\Xi}^{k(i,d)}}^{k(i,d)}(\mathbf{z}); \mathbf{X})\Big).
\end{aligned}
\tag{12}
$$

In (12), $K_{ij}(\mathbf{X})$ and $F_i(\mathbf{X})$ refer to the (i, j)th and F_ith entry of $\mathbf{K}(\mathbf{X})$ and $\mathbf{F}(\mathbf{X})$, respectively, which are typically random. Moreover, k_1, k_2, and k are the control points indices, which are related to i or j and the problem dimension d. Eventually, solve for $\mathbf{d}(\mathbf{X})$ by

$$\mathbf{d}(\mathbf{X}) = \mathbf{K}^{-1}(\mathbf{X})\mathbf{F}(\mathbf{X})$$

and proceed with constructing the approximate displacement solution $\mathbf{u}^h(\mathbf{z}; \mathbf{X}) \in \mathcal{U}^h$, as done in conventional finite element methods [42].

Note that the displacements $\mathbf{u}^h(\mathbf{z}; \mathbf{X})$ are random because the stiffness matrix $\mathbf{K}(\mathbf{X})$ and force vector $\mathbf{F}(\mathbf{X})$ are generally random. Thus, the random stress response is calculated by

$$\boldsymbol{\sigma}^h(\mathbf{z}; \mathbf{X}) = \mathbf{D}(\mathbf{z}; \mathbf{X})\boldsymbol{\epsilon}^h(\mathbf{z}; \mathbf{X}),$$

where $\boldsymbol{\epsilon}^h(\mathbf{z}; \mathbf{X}) := (1/2)(\nabla + \nabla^\top)\mathbf{u}^h(\mathbf{z}; \mathbf{X})$ is defined as the approximate strain tensor. Furthermore, the elasticity tensor $\mathbf{D}(\mathbf{z}; \mathbf{X})$ is generally random as it carries the information about the material properties. In this work, $\mathbf{u}^h(\mathbf{z}; \mathbf{X})$ and $\boldsymbol{\sigma}^h(\mathbf{z}; \mathbf{X})$ are the main response variables of interest.

5. Spline dimensional decomposition

This section encompasses the SDD method for uncertainty quantification with deterministic IGA solver for arbitrary multipatch domains. While the details have been suppressed for brevity, readers interested in specifics and mathematical aspects of SDD are directed to the authors' previous work [31].

5.1. Standard B-splines

Denote by $n_k \in \mathbb{N}$ and $p_k \in \mathbb{N}_0$ the number of spline basis functions and polynomial order, respectively, in coordinate direction $k \in \mathbb{N}$. A knot vector $\boldsymbol{\xi}_k$ is defined as a non-decreasing sequence of real numbers as

$$
\begin{aligned}
\boldsymbol{\xi}_k &:= \{\xi_{k,i_k}\}_{i_k=1}^{n_k+p_k+1} = \{a_k = \xi_{k,1}, \xi_{k,2}, \dots, \xi_{k,n_k+p_k+1} = b_k\}, \\
&\qquad \xi_{k,1} \le \xi_{k,2} \le \cdots \le \xi_{k,n_k+p_k+1}
\end{aligned}
$$

on a bounded interval $[a_k, b_k] \subset \mathbb{R}$ with $a_k < b_k$ and $n_k > p_k \ge 0$. Note that ξ_{k,i_k} refers to the i_kth knot in coordinate direction k, where $i_k = 1, 2, \dots, n_k + p_k + 1$. The knots may or may not be equally spaced, although this work only considers equal knot spacing.

Denote by r_k the total number of distinct knots in a knot vector and re-write the knot vector as

$$
\boldsymbol{\xi}_k = (a_k = \underbrace{\zeta_{k,1}, \dots, \zeta_{k,1}}_{m_{k,1} \text{ times}}, \underbrace{\zeta_{k,2}, \dots, \zeta_{k,2}}_{m_{k,2} \text{ times}}, \dots, \underbrace{\zeta_{k,r_k}, \dots, \zeta_{k,r_k}}_{m_{k,r_k} \text{ times}} = b_k),
$$

where ζ_{k,j_k}, $j_k = 1, \dots, r_k$ is a distinct knot and m_{k,j_k}, $j_k = 1, \dots, r_k$ is its associated multiplicity. Therefore, for every $\zeta_{k,j_k} > \zeta_{k,j_k-1}$, denote by $[\zeta_{k,j_k-1}, \zeta_{k,j_k}] \subset \mathbb{R}$ a subinterval of the bounded domain $[a_k, b_k]$. A knot vector is called open if its first and last knots appear $(p_k + 1)$ times [44]. The knot vectors considered in this work are open, and all the knots on the interior region (a_k, b_k) have multiplicity of *one*.

Univariate B-splines are constructed in a recursive manner. The *zero*-order basis functions are computed by

$$B^k_{i_k,0,\boldsymbol{\xi}_k}(x_k) = \begin{cases} 1, & \xi_{k,i_k} \leq x_k < \xi_{k,i_k+1}, \\ 0, & \text{otherwise}, \end{cases}$$

and the higher-order functions are built by the Cox–de Boor formula [39] as

$$B^k_{i_k,p_k,\boldsymbol{\xi}_k}(x_k) = \frac{x_k - \xi_{k,i_k}}{\xi_{k,i_k+p_k} - \xi_{k,i_k}} B^k_{i_k,p_k-1,\boldsymbol{\xi}_k}(x_k) + \frac{\xi_{k,i_k+p_k+1} - x_k}{\xi_{k,i_k+p_k+1} - \xi_{k,i_k+1}} B^k_{i_k+1,p_k-1,\boldsymbol{\xi}_k}(x_k),$$

where $i_k = 1, \ldots, n_k$, $x_k \in [a_k, b_k]$ is the function argument, and $0/0$ is set as *zero*.

The B-splines are non-negative, locally supported on the subintervals, linearly independent, and committed to partition of unity [9]. As a result, they form a basis over $[a_k, b_k]$. A B-spline function is C^∞-continuous everywhere except at the distinct knots ζ_{k,i_k}, where it is $C^{p_k-m_{k,i_k}}$-continuous, with the condition $1 \leq m_{k,i_k} < p_k + 1$.

5.2. Measure-consistent orthonormalized B-splines

The orthogonalization of B-splines has been previously elaborated on for regression formulations [45]. For UQ applications, however, having orthogonal basis functions is beneficial, because it helps with more stable computational analysis and elegant formulation of the method [31,33].

5.2.1. Univariate orthonormalized B-splines

In this section, a linear transformation, originally proposed in the authors' previous work [31], is presented as follows to construct measure-consistent orthonormalized B-splines associated with any probability distribution of interest.

(1) Given a set of B-splines for degree p_k and knot vector $\boldsymbol{\xi}_k$ in coordinate direction k, create an auxiliary set by replacing, arbitrarily, the first basis function with 1. Arrange the elements of the set into an auxiliary n_k-dimensional vector

$$\mathbf{P}_k(x_k) := (1, B^k_{2,p_k,\boldsymbol{\xi}_k}(x_k), \ldots, B^k_{n_k,p_k,\boldsymbol{\xi}_k}(x_k))^T,$$

which are still linearly independent [31].

(2) Form the spline moment matrix $\mathbf{G}_k$ of dimension $n_k \times n_k$ as

$$\mathbf{G}_k := \mathbb{E}[\mathbf{P}_k(X_k)\mathbf{P}_k^T(X_k)],$$

where $\mathbb{E}$ is the expectation operator. Moreover, $\mathbf{G}_k$ is symmetric and positive-definite by definition [31]. Thus, a non-singular $n_k \times n_k$ whitening matrix $\mathbf{W}_k$ exists such that

$$\mathbf{W}_k^T \mathbf{W}_k = \mathbf{G}_k^{-1}.$$

(3) Apply a whitening transformation $\mathbf{W}_k$ on the auxiliary set of B-splines $\mathbf{P}_k(x_k)$ to obtain

$$\boldsymbol{\psi}_k(x_k) = \mathbf{W}_k \mathbf{P}_k(x_k),$$

which has the univariate orthonormalized basis functions $\psi^k_{i_k,p_k,\boldsymbol{\xi}_k}(x_k)$, $i_k = 1, \ldots, n_k$, $k = 1, \ldots, N$. Recall that N is the number of random variables. There are many options for $\mathbf{W}_k$. However, a straightforward choice is to apply the Cholesky factorization on the spline moment matrix as

$$\mathbf{G}_k = \mathbf{Q}_k \mathbf{Q}_k^\mathsf{T},$$

and obtain the whitening matrix as

$$\mathbf{W}_k = \mathbf{Q}_k^{-1}.$$

Since the input random variables are independent, multivariate B-splines in N variables are constructed by tensor product of measure-consistent univariate orthonormalized B-spline ingredients. However, since the tensor-product structure can be costly for high-dimensional problems, the authors advocate building the multivariate functions in a dimensionwise manner.

5.2.2. Dimensionwise multivariate orthonormalized B-splines

For a subset $\emptyset \neq u = \{k_1, \ldots, k_{|u|}\} \subseteq \{1, \ldots, N\}$, define $\mathbf{X}_u := (X_{k_1}, \ldots, X_{k_{|u|}})^\mathsf{T}$ on the abstract probability space $(\Omega^u, \mathcal{F}^u, \mathbb{P}^u)$, where Ω^u is the sample space, $\mathcal{F}^u$ is a σ-algebra on Ω^u, and $\mathbb{P}^u$ is a probability measure. As $\mathbf{X}$ comprises independent random variables, the PDF of $\mathbf{X}_u$ is

$$f_{\mathbf{X}_u}(\mathbf{x}_u) = \prod_{k \in u} f_{X_k}(x_k) = \prod_{l=1}^{|u|} f_{X_{k_l}}(x_{k_l}), \ \ \mathbf{x}_u := (x_{k_1}, \ldots, x_{k_{|u|}})^\mathsf{T}.$$

Furthermore, denote by $\mathbf{i}_u := (i_{k_1}, \ldots, i_{k_{|u|}}) \in \mathbb{N}^{|u|}$, $\mathbf{n}_u := (n_{k_1}, \ldots, n_{k_{|u|}}) \in \mathbb{N}^{|u|}$, and $\mathbf{p}_u := (p_{k_1}, \ldots, p_{k_{|u|}}) \in \mathbb{N}_0^{|u|}$, three multi-indices for the knot indices, numbers of basis functions, and degrees of B-splines, respectively. Define an index set

$$\mathcal{I}_{u,\mathbf{n}_u} := \left\{ \mathbf{i}_u = (i_{k_1}, \ldots, i_{k_{|u|}}) : 1 \leq i_{k_l} \leq n_{k_l}, \ l = 1, \ldots, |u| \right\} \subset \mathbb{N}^{|u|}$$

with the cardinality

$$|\mathcal{I}_{u,\mathbf{n}_u}| = \prod_{k \in u} n_k.$$

For the coordinate direction k_l, denote by r_{k_l} the number of distinct knots in the knot vector $\boldsymbol{\xi}_{k_l}$. Hence,

$$I_{k_l} = r_{k_l} - 1,$$

is the number of subintervals.

The multivariate B-splines in $\mathbf{x}_u = (x_{k_1}, \ldots, x_{k_{|u|}})$ consistent with the probability measure $f_{\mathbf{X}_u}(\mathbf{x}_u) d\mathbf{x}_u$ are constructed as

$$\Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{x}_u) = \prod_{k \in u} \psi^k_{i_k, p_k, \boldsymbol{\xi}_k}(x_k) = \prod_{l=1}^{|u|} \psi^k_{i_{k_l}, p_{k_l}, \boldsymbol{\xi}_{k_l}}(x_{k_l}), \ \ \mathbf{i}_u = (i_{k_1}, \ldots, i_{k_{|u|}}) \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}, \tag{13}$$

where

$$\bar{\mathcal{I}}_{u,\mathbf{n}_u} := \left\{ \mathbf{i}_u = (i_{k_1}, \ldots, i_{k_{|u|}}) : 2 \leq i_{k_l} \leq n_{k_l}, \ l = 1, \ldots, |u| \right\} \subset (\mathbb{N} \setminus \{1\})^{|u|},$$

is defined as a reduced index set with the cardinality

$$|\bar{\mathcal{I}}_{u,\mathbf{n}_u}| := \prod_{k \in u} (n_k - 1). \tag{14}$$

The function arguments in (13) are substituted with the input random variables $X_1, \ldots, X_N$ to yield multivariate B-splines $\Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{X}_u)$ as functions of variables with uncertainty. Remove the first element of $\psi_k(X_k)$, which is the constant term, by limiting the index i_{k_l} associated with x_{k_l} to run from 2 to n_{k_l}. Hence,

For given $\emptyset \neq u, v \subseteq \{1, \ldots, N\}$, $\mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}$, and $\mathbf{j}_v \in \bar{\mathcal{I}}_{v,\mathbf{n}_v}$, the first and second statistical moments of the multivariate orthonormalized B-splines are [31]

$$\mathbb{E}\left[\Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{X}_u)\right] = 0 \tag{15}$$

and

$$\mathbb{E}\left[\Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{X}_u) \Psi^v_{\mathbf{j}_v, \mathbf{p}_v, \boldsymbol{\Xi}_v}(\mathbf{X}_v)\right] = \begin{cases} 1, & u = v \text{ and } \mathbf{i}_u = \mathbf{j}_v, \\ 0, & \text{otherwise}, \end{cases} \tag{16}$$

respectively. These second-moment statistical properties are important for developing the SDD method.

5.3. SDD approximation

For a set of measure-consistent multivariate B-splines $\{\Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{X}_u) : \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}\}$, the size of the set is shown in (14), which depends on the number of basis functions n_{k_l}, and by extension, the length of the knot vector $\boldsymbol{\xi}_{k_l}$ and order p_{k_l}. As the refinement process, consider a fixed $\mathbf{p}_u$ while increasing the length of $\boldsymbol{\xi}_{k_l}$ in all $|u|$ coordinate directions, where the largest subinterval size, say h_{k_l}, is monotonically reduced. In the limit, when $n_{k_l} \to \infty$,

$k_l = 1, \ldots, |u|$, denote by $\boldsymbol{\xi}_{k_l,\infty}$ and $\boldsymbol{\Xi}_{u,\infty} = (\boldsymbol{\xi}_{1,\infty}, \ldots, \boldsymbol{\xi}_{|u|,\infty})$ the associated knot vector of infinite length in the k_lth coordinate direction and the collection of $|u|$ such knot vectors, respectively. Write the multi-index set

$$\bar{\mathcal{I}}_{u,\infty} := \left\{ \mathbf{i}_u = (i_{k_1}, \ldots, i_{k_{|u|}}) : 2 \le i_{k_l} < \infty, \ l = 1, \ldots, |u| \right\}$$

to call the basis functions corresponding to $\boldsymbol{\Xi}_{u,\infty} = (\boldsymbol{\xi}_{1,\infty}, \ldots, \boldsymbol{\xi}_{|u|,\infty})$.

According to the standard Hilbert space theory, one can write the Fourier spline expansion [31] as

$$y(\mathbf{X}) \sim y_\emptyset + \sum_{\emptyset \ne u \subseteq \{1,\ldots,N\}} \sum_{\mathbf{i}_u \in \bar{\mathcal{I}}_{u,\infty}} C^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_{u,\infty}} \Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_{u,\infty}}(\mathbf{X}_u), \tag{17}$$

where

$$y_\emptyset := \int_{\mathbb{A}^N} y(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \tag{18}$$

and

$$C^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_{u,\infty}} := \int_{\mathbb{A}^N} y(\mathbf{x}) \Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_{u,\infty}}(\mathbf{x}_u) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}, \ \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\infty} \tag{19}$$

are various expansion coefficients.

The expansion in (17) is a Fourier-like expansion and is referred to as SDD. Evidently, the SDD of any square-integrable random variable $y(\mathbf{X})$ is a dimensionwise orthogonal projection onto the spline space spanning the set of associated measure-consistent multivariate orthonormalized B-splines [31]. The SDD formulation is valid for any square-integrable random function $y(\mathbf{X})$ and is not specific to linear elasticity problems.

In practice, the knot vectors are not of infinite dimension and n_{k_l}, $k_l = 1, \ldots, |u|$ is a finite number. In this case, a truncated set $\{\Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{X}_u) : \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}\}$ is used to approximate $y(\mathbf{X})$, and an SDD approximation is written as

$$y_{\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X}) := y_\emptyset + \sum_{\emptyset \ne u \subseteq \{1,\ldots,N\}} \sum_{\mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}} C^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u} \Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{X}_u), \tag{20}$$

where the expansion coefficients are defined as

$$C^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u} := \int_{\mathbb{A}^N} y(\mathbf{x}) \Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{x}_u) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}, \quad \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}, \tag{21}$$

while the computation of $y_\emptyset$ still follows (18).

Since SDD benefits from a dimensional, hierarchical structure, many higher-variate interaction terms typically contribute only marginally to the function value and therefore can be safely ignored. A convenient approach to ignore higher-variate terms in SDD is to retain the terms including at most $1 \le S \le N$ variables. Therefore, S is applied as an important truncation parameter to keep only the terms with the degrees of interaction less than or equal to S. In mathematical form, an S-variate SDD approximation reads

$$y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X}) := y_\emptyset + \sum_{\substack{\emptyset \ne u \subseteq \{1,\ldots,N\} \\ 1 \le |u| \le S}} \sum_{\mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}} C^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u} \Psi^u_{\mathbf{i}_u, \mathbf{p}_u, \boldsymbol{\Xi}_u}(\mathbf{X}_u), \tag{22}$$

which is called a truncated SDD approximation. When N is large and $S \ll N$, as it is common in real-life applications, the number of expansion coefficients drops significantly, which yields substantial savings of computational effort. This will be discussed further in Section 5.4.

For any square-integrable function $y(\mathbf{X})$, the sequence of SDD approximations $\{y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})\}_{1 \le S \le N, \ \mathbf{h} > \mathbf{0}}$ converges to $y(\mathbf{X})$ in mean-square [31], that is,

$$\lim_{S \to N, \ \mathbf{h} \to \mathbf{0}} \mathbb{E}\left[\left| y(\mathbf{X}) - y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X}) \right|^2 \right] = 0,$$

where $\mathbf{h}$ represents the vector of largest subinterval sizes associated with the coordinate directions.

The SDD approximations in (20) and (22) converge in probability and in distribution. Readers interested in formal proofs are directed to the prior theoretical work [31].

5.4. Computational effort

In an SDD approximation $y_{\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ of a random function $y(\mathbf{X})$, there is a total of

$$L_{\mathbf{p},\boldsymbol{\Xi}} = 1 + \sum_{\emptyset \neq u \subseteq \{1,\ldots,N\}} \prod_{k \in u} (n_k - 1) = \prod_{k=1}^{N} n_k$$

expansion coefficients. This clearly means that the SDD method succumbs to the curse of dimensionality, implying that if all the terms in (20) are retained, the number of expansion coefficients surges exponentially versus N. In such case, the SDD method is identical to SCE [29].

On the other hand, a truncated SDD method introduced in (22) comprises of

$$L_{S,\mathbf{p},\boldsymbol{\Xi}} = 1 + \sum_{\substack{\emptyset \neq u \subseteq \{1,\ldots,N\} \\ 1 \leq |u| \leq S}} \prod_{k \in u} (n_k - 1) \leq \prod_{k=1}^{N} n_k = L_{\mathbf{p},\boldsymbol{\Xi}} \tag{23}$$

expansion coefficients. As mentioned before, for high-dimensional stochastic problems, if $S \ll N$, for instance, univariate ($S = 1$) or bivariate ($S = 2$) SDD approximations lead to tremendously efficient estimations of the random function $y(\mathbf{X})$, as the numbers of expansion coefficients decline drastically. As an example, set $N = 20$, $n_k = 5$, and $S = 1$ or 2. By retaining all the terms in the SDD approximation, the system is equivalent to SCE, and there are more than 9×10^{13} expansion coefficients, whereas the univariate and bivariate SDD approximations include only 81 and 3121 terms, respectively. This is because of the polynomial computational complexity of truncated SDD with respect to N versus the exponential computational complexity in the case of SCE. This comparison between the computations efforts demonstrates the huge impact of SDD truncation by taking advantage of the dimensionwise decomposition of $y(\mathbf{X})$, thereby alleviating the curse of dimensionality to the extent possible, while in many applications, the loss of accuracy by setting $S = 1$ or 2 is negligible.

Furthermore, $y_{1,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ or $y_{2,2,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ should not be viewed as first- and second-order approximations because S only limits the degree of interaction between the random variables, but not their order. As a result, heavily non-linear functions can still be effectively modeled by low-variate B-spline component functions. This will be re-visited in the Numerical Examples section.

How to choose a suitable combination of p, I, and S for a given problem remains an open question. As this process is problem-dependent, an effective SDD method is picked based on either existing insight about the regularity of the random function, prior experience from employing other UQ methods, or trial-and-error if the response function is completely unknown. However, numerical schemes can be developed to pick the B-spline orders and knot vectors in an adaptive framework that comprises of an error measure, a computational budget, and an optimization algorithm [46].

5.5. Output statistics and other properties

In Section 5.3 an inexpensive model $y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ was developed as a surrogate of the original random function $y(\mathbf{X})$, which is generally expensive to evaluate. Therefore, investigating the statistical moments of $y(\mathbf{X})$ through those of $y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ is tremendously beneficial, provided that the expansion coefficients of $y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ are accurately calculated.

By applying the expectation operators on $y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ and using (15) and (16), the mean and variance of $y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ are formulated as [36]

$$\mathbb{E}\left[y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})\right] = y_{\emptyset} = \mathbb{E}\left[y(\mathbf{X})\right] \tag{24}$$

and

$$\text{var}\left[y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})\right] = \sum_{\substack{\emptyset \neq u \subseteq \{1,\ldots,N\} \\ 1 \leq |u| \leq S}} \sum_{\mathbf{i}_u \in \tilde{\mathcal{I}}_{u,\mathbf{n}_u}} C^{u2}_{\mathbf{i}_u,\mathbf{p}_u,\boldsymbol{\Xi}_u} \leq \text{var}\left[y(\mathbf{X})\right], \tag{25}$$

respectively. As observed, the mean is equivalent to $y_{\emptyset}$, which is independent of S, $\mathbf{p}$, and $\boldsymbol{\Xi}$. More importantly, if the integration in (18) is computed exactly, the SDD approximation always yields the exact mean.

The analytical equations (24) and (25) clearly state that the accuracy to which the second-moment statistical characteristics of a random response are computed is highly dependent on the accuracy of the coefficient calculation. While there are several methods to estimate the expansion coefficients efficiently, the cumulative distribution function (CDF) and the PDF of $y(\mathbf{X})$, if they exist, can also be estimated by economically re-sampling the surrogate model $y_{S,\mathbf{p},\varXi}(\mathbf{X})$. This will be further discussed in the Numerical Examples section.

5.6. Calculation of expansion coefficients

The expansion coefficients in SDD are naturally computed by invoking their respective definitions in (18) and (21). This is generally done by numerical computations, which can be expensive, especially in the case of high-dimensional problems. As a remedy, a dimension-reduction method has been proposed with B-spline basis functions [33] to tremendously decrease the dimension of the integrals involved, thereby drastically boosting the computational expediency. However, there are various methods to estimate the expansion coefficients.

The SLS regression method provides a way to efficiently solve linear systems of equations. In this work, the SLS method is employed to estimate the SDD expansion coefficients, thereby avoiding the costly numerical integration in (18) and (19). This is accomplished in the context of single-index description of SDD as follows.

5.6.1. A single-index variant of SDD

Consider the collection of measure-consistent orthonormalized multivariate B-splines

$$\{\Psi_{\mathbf{i}_u,\mathbf{p}_u,\varXi_u}^{u}(\mathbf{X}_u) : 1 \le |u| \le S, \ \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}\}, \tag{26}$$

as previously defined in Section 5.3, which consists of $L_{S,\mathbf{p},\varXi}$ basis functions. Without loss of generality and consistent with the orthonormalization procedure discussed in Section 5.2, write $\Psi_1(\mathbf{X}) = 1$, and with an arbitrary order of choice, arrange the elements of the set in (26) as

$$\left\{\Psi_{\mathbf{i}_u,\mathbf{p}_u,\varXi_u}^{u}(\mathbf{X}_u) : 1 \le |u| \le S, \ \mathbf{i}_u \in \bar{\mathcal{I}}_{u,\mathbf{n}_u}\right\} = \left\{\Psi_2(\mathbf{X}), \dots, \Psi_{L_{S,\mathbf{p},\varXi}}(\mathbf{X})\right\}, \quad \Psi_1(\mathbf{X}) = 1,$$

such that $\Psi_i(\mathbf{X})$, $i = 1, \dots, L_{S,\mathbf{p},\varXi}$ represents the ith basis function in a truncated SDD expansion. For each basis function, there exists an associated expansion coefficient $C_i \in \mathbb{R}$, $i = 1, \dots, L_{S,\mathbf{p},\varXi}$. Hence, re-write the truncated SDD approximation as

$$y_{S,\mathbf{p},\varXi}(\mathbf{X}) := \sum_{i=1}^{L_{S,\mathbf{p},\varXi}} C_i \Psi_i(\mathbf{X}). \tag{27}$$

The result obtained from $y_{S,\mathbf{p},\varXi}(\mathbf{X})$ in (27) is identical to that computed from (22).

5.6.2. Least-squares regression for SDD coefficient estimation

By employing the SLS regression method, the approximate SDD expansion coefficients of $y_{S,\mathbf{p},\varXi}(\mathbf{X})$ are computed through minimizing

$$\mathbb{E}\left[y(\mathbf{X}) - \sum_{i=1}^{L_{S,\mathbf{p},\varXi}} C_i \Psi_i(\mathbf{X})\right]^2.$$

In UQ applications, the probability distributions of the input random variables in $\mathbf{X} = \{X_i, \dots, X_N\}$ are defined and a random response function $y : \mathbb{A}^N \to \mathbb{R}$ is under study with unknown uncertainty. Consider a set of input–output data set $\{\mathbf{x}^{(l)}, y(\mathbf{x}^{(l)})\}_{l=1}^{L}$ of size $L \in \mathbb{N}$. Each element of this data set is generally obtained from an expensive finite element analysis or IGA of a complex system. The IGA formulation, for instance, solves a PDE, as described in Section 4, for a fixed input vector $\mathbf{x}^{(l)}$ comprising samples drawn from associated probability distributions of the random input variables. As a result, the corresponding response of the deterministic IGA solver is also recorded as $y(\mathbf{x}^{(l)})$, and the process is repeated $L \in \mathbb{N}$ times. There are various methods, such as the crude MCS, quasi MCS, and Latin hypercube sampling methods, for drawing samples from the probability distributions to form $\mathbf{x}^{(l)}$.

Having obtained the data set $\{\mathbf{x}^{(l)}, y(\mathbf{x}^{(l)})\}_{l=1}^{L}$, the SLS regression requires minimizing the mean-square error [47]

$$e_{S,\mathbf{p},\varXi} := \frac{1}{L} \sum_{l=1}^{L} \left[y(\mathbf{x}^{(l)}) - \sum_{i=1}^{L_{S,\mathbf{p},\varXi}} C_i \Psi_i(\mathbf{x}^{(l)})\right]^2,$$

which is done by estimating the expansion coefficients through

$$\hat{\mathbf{c}} := \left(\hat{C}_1, \ldots, \hat{C}_{L_{S,\mathbf{p},\Xi}} \right)^{\mathsf{T}} = \left(\mathbf{A}^{\mathsf{T}} \mathbf{A} \right)^{-1} \mathbf{A}^{\mathsf{T}} \mathbf{b} \tag{28}$$

as an estimation of $\mathbf{c} := (C_1, \ldots, C_{L_{S,\mathbf{p},\Xi}})^{\mathsf{T}}$, where

$$\mathbf{A} := \begin{bmatrix} \Psi_1(\mathbf{x}^{(1)}) & \cdots & \Psi_{L_{S,\mathbf{p},\Xi}}(\mathbf{x}^{(1)}) \\ \vdots & \ddots & \vdots \\ \Psi_1(\mathbf{x}^{(L)}) & \cdots & \Psi_{L_{S,\mathbf{p},\Xi}}(\mathbf{x}^{(L)}) \end{bmatrix} \text{ and } \mathbf{b} := \left(y(\mathbf{x}^{(1)}), \ldots, y(\mathbf{x}^{(L)}) \right)^{\mathsf{T}}.$$

Note that matrix $\mathbf{A}$ of dimension $L \times L_{S,\mathbf{p},\Xi}$ carries the information about the basis functions used in the SLS method, which are evaluated at $\{\mathbf{x}^{(l)}\}_{l=1}^{L}$. Moreover, the L-dimensional column vector $\mathbf{b}$ has the responses reported by the IGA deterministic solver, previously discussed in Section 4.2.2.

Furthermore, the square matrix $\mathbf{A}^{\mathsf{T}}\mathbf{A}$ of dimension $L_{S,\mathbf{p},\Xi} \times L_{S,\mathbf{p},\Xi}$, which is referred to as the information matrix, plays a crucial role in the SLS method. The accuracy of the regression method hinges on the condition number of the information matrix, which itself depends on how many samples are used in the data set and how they are picked for analysis. In other words, matrix $\mathbf{A}^{\mathsf{T}}\mathbf{A}$ should not be singular and had better be well-conditioned. While there are rules of thumb stating that two or three times the number of coefficients $L_{S,\mathbf{p},\Xi}$ is typically sufficient for the number of samples L, the selection of L heavily depends on the application and may yield either inaccurate, if attainable, results or unnecessarily expensive computations if L is too small or too large, respectively. Having said this, a necessary condition for the SLS method reads $L > L_{S,\mathbf{p},\Xi}$.

There are measures to pick the samples for regression more cleverly. For example, the leave-one-out error has proven to do very well with estimating the mean-square error [48] and cross validation. Moreover, the least angle regression [49] is another popular method used in stochastic finite elements by regression [50] and other UQ applications. Readers interested in more details are directed to the aforementioned works.

5.7. Integration of SDD and multipatch IGA

In Section 4, a stochastic PDE was defined to solve stochastic linear elasticity problems. The displacements and stresses are generally not available in closed form. Hence, a matrix formulation was proposed to solve for $\mathbf{u}^h(\mathbf{z}; \mathbf{X})$ and $\boldsymbol{\sigma}^h(\mathbf{z}; \mathbf{X})$ numerically by using an IGA deterministic solver, which exploits analysis-suitable T-splines for modeling and analysis of arbitrary multipatch domains. Henceforth, the random response function of interest $y(\mathbf{X})$ defined in Section 5.3, is re-defined as

$$y(\mathbf{X}) := y(\mathbf{u}^h(\mathbf{z}; \mathbf{X}); \boldsymbol{\sigma}^h(\mathbf{z}; \mathbf{X})),$$

where the response is approximated by multipatch IGA. Therefore, the SDD method in (27) and the multipatch IGA solver are integrated to yield an SDD-multipatch IGA (SDD-mIGA) method.

Fig. 5 illustrates a flow chart for the proposed SDD-mIGA method. The algorithm reads some input data for the random vector $\mathbf{X}$, the number of samples for SLS regression L, and the IGA-related data, such as the geometry, loads, and material properties. The IGA responses are stored in the response vector $\mathbf{b}$. Hence, the SDD algorithm is carried out in a non-intrusive manner, as the expansion coefficients are estimated efficiently by applying (28). Eventually, the function $y_{S,\Xi,\mathbf{p}}$ is written by (27) as a surrogate of the original function $y(\mathbf{X})$.

Note that although the proposed method utilizes SLS regression, various numerical techniques can, indeed, be applied to estimate the SDD expansion coefficients.

Moreover, from this point forth, the SDD-mIGA method proposed for UQ in practical engineering problems will simply be called SDD for brevity.

6. Numerical examples

The numerical examples section starts with the second-moment statistical analysis of nonsmooth and high-dimensional stochastic mathematical functions not related to IGA and custom-tailored to generally evaluate the efficacy of SDD for UQ. Although this work focuses mainly on the applications of SDD, the authors aim to scrutinize its accuracy in Example 1 by making proper comparisons with other common UQ methods, such as PCE and sparse grids. Examples 2 and 3 involve a two-dimensional connecting rod and a three-dimensional gear,
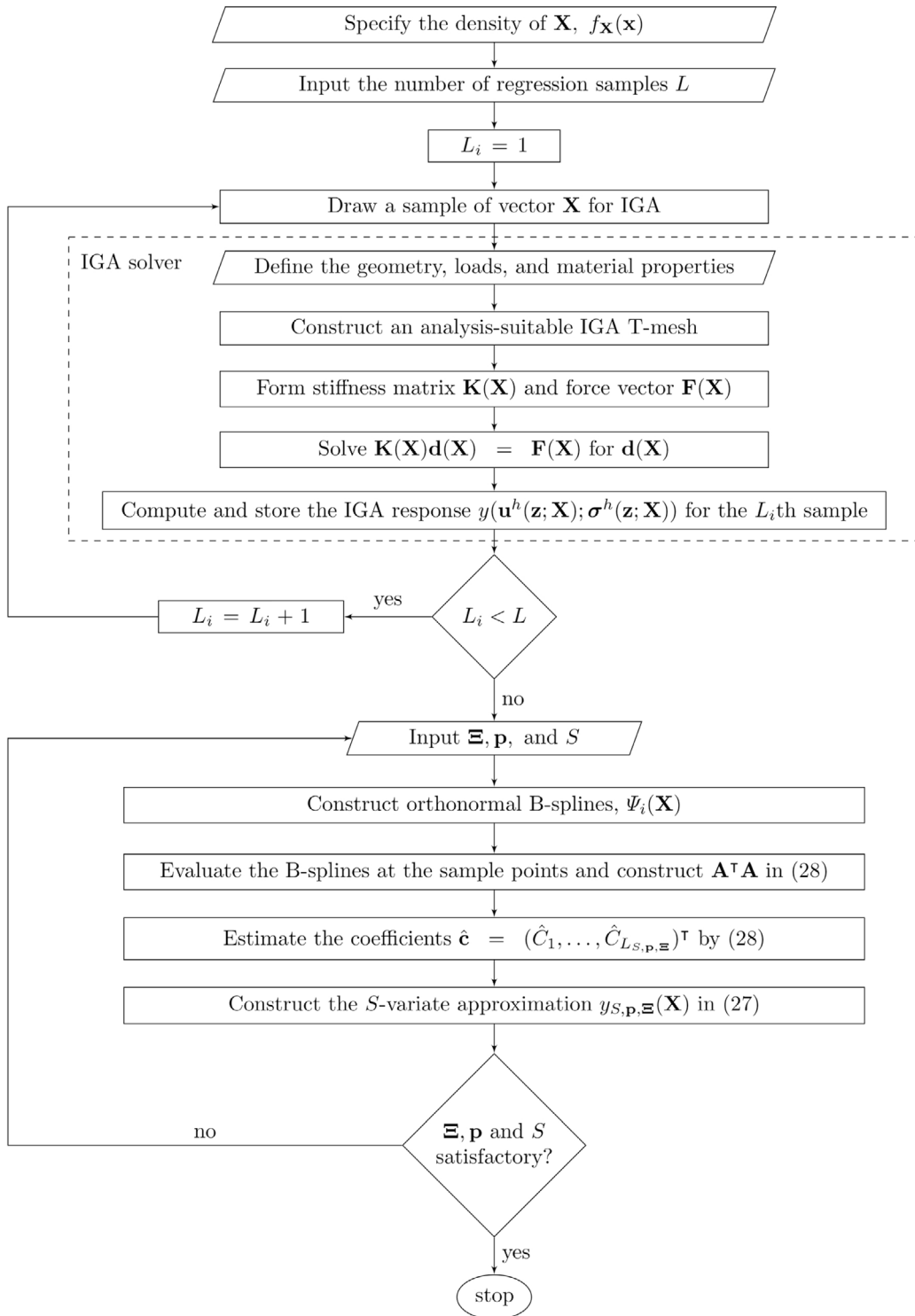
Specify the density of $\mathbf{X}$, $f_{\mathbf{X}}(\mathbf{x})$

Input the number of regression samples L

$L_i = 1$

Draw a sample of vector $\mathbf{X}$ for IGA

IGA solver

Define the geometry, loads, and material properties

Construct an analysis-suitable IGA T-mesh

Form stiffness matrix $\mathbf{K}(\mathbf{X})$ and force vector $\mathbf{F}(\mathbf{X})$

Solve $\mathbf{K}(\mathbf{X})\mathbf{d}(\mathbf{X}) = \mathbf{F}(\mathbf{X})$ for $\mathbf{d}(\mathbf{X})$

Compute and store the IGA response $y(\mathbf{u}^h(\mathbf{z};\mathbf{X}); \boldsymbol{\sigma}^h(\mathbf{z};\mathbf{X}))$ for the L_ith sample

$L_i = L_i + 1$

yes

$L_i < L$

no

Input $\boldsymbol{\Xi}, \mathbf{p},$ and S

Construct orthonormal B-splines, $\Psi_i(\mathbf{X})$

Evaluate the B-splines at the sample points and construct $\mathbf{A}^{\mathsf{T}}\mathbf{A}$ in (28)

Estimate the coefficients $\hat{\mathbf{c}} = (\hat{C}_1, \ldots, \hat{C}_{L_{S,\mathbf{p},\boldsymbol{\Xi}}})^{\mathsf{T}}$ by (28)

Construct the S-variate approximation $y_{S,\mathbf{p},\boldsymbol{\Xi}}(\mathbf{X})$ in (27)

no

$\boldsymbol{\Xi}, \mathbf{p}$ and S satisfactory?

yes

stop

Fig. 5. A flow chart for the proposed SDD-mIGA method.

respectively, from the real world to study the impact of the proposed SDD method in conjunction with multipatch IGA. Essentially, solving practical engineering problems involving complex geometries has been the main focus of this work.

In Examples 2 and 3, the geometries were modeled precisely by multipatch IGA and the problems were solved considering fixed IGA finite element models. In other words, the IGA meshes were not refined, and the impact of mesh refinement was not studied in this work. However, the mesh densities were deemed satisfactory in terms of the accuracy provided and the computation burden imposed to the analyses. In both examples, the IGA mesh and its corresponding control points were obtained by dint of the refinement matrix computed via the Bézier extraction operator. This was accomplished by acting on a base coarse T-mesh $\mathcal{T}_0$, as discussed in the Appendix. Moreover, since the T-splines technology has still not been commercialized in a user-friendly software package, all the analyses were conducted by developing Matlab [51] codes. Moreover, MCS was employed in the examples to provide benchmark solutions.

In all examples, the B-spline order p and knot vector $\boldsymbol{\xi}$ used for the SDD calculations were identical in all coordinate directions of the stochastic domain. As a result, all subscripts k from p_k, $\boldsymbol{\xi}_k$, and I_k were dropped accordingly. Furthermore, all the knot vectors used by SDD had equally spaced interior knots. All the analyses in this work were carried out on a standard desktop computer.

6.1. Example 1: Mathematical functions

The first numerical example comprises of two problems with the goal of accurately estimating the second-moment statistical properties of analytical random functions. In Example 1-1, low-dimensional and high-dimensional nonsmooth and discontinuous stochastic functions are studied, where SDD is compared with other popular UQ methods, namely, PCE and sparse grids to rigorously conduct a convergence study and show the impact of SDD in handling harsh functions in terms of regularity. In addition, Example 1–2 entails UQ analysis of a relatively smooth function only by SDD, where the objective is to demonstrate the impact of the SDD truncation parameter S on the resulting SDD accuracy by keeping at most S-variate terms in the spline expansion, where $S \leq N$.

6.1.1. Example 1-1: Nonsmooth and discontinuous functions

Consider the nonsmooth, bivariate ($N = 2$) mathematical function

$$y_1(X_1, X_2) = f_1(X_1) + f_2(X_2) + 0.05 f_1(X_1) f_2(X_2),$$

where

$$f_i(X_i) = \begin{cases} \exp(5X_i), & \text{if } X_i < 0, \\ \exp(-5X_i), & \text{if } X_i \geq 0, \end{cases} \quad i = 1, 2,$$

and the discontinuous N-variate function

$$y_2(\mathbf{X}) = \prod_{i=1}^{N} g_i(X_i),$$

where

$$g_i(X_i) = \begin{cases} 1, & \text{if } X_i \leq 0, \\ 0.5 \exp(X_i), & \text{if } X_i > 0, \end{cases} \quad i = 1, 2, \ldots, N. \tag{29}$$

Define all $X_i, i = 1, \ldots, N$ as independent, identical random variables with uniform probability distribution on $[-1, 1]$. Here, $y_1(X_1, X_2)$ is clearly a bivariate function, whereas $y_2(\mathbf{X})$ is studied for two cases: (1) $N = 2$ (bivariate) and (2) $N = 10$ (decavariate) to cover the two classes of low-dimensional and high-dimensional stochastic functions to the extent possible. Hence, there will be three sets of numerical results in total.

Fig. 6 depicts the two-dimensional ($N = 2$) functions. The objective of this problem is to estimate the variance of the defined stochastic functions. Bearing in mind that the variance of a random variable is generally more computationally challenging to estimate than its mean, the focus of this example has been allocated to variance investigations.
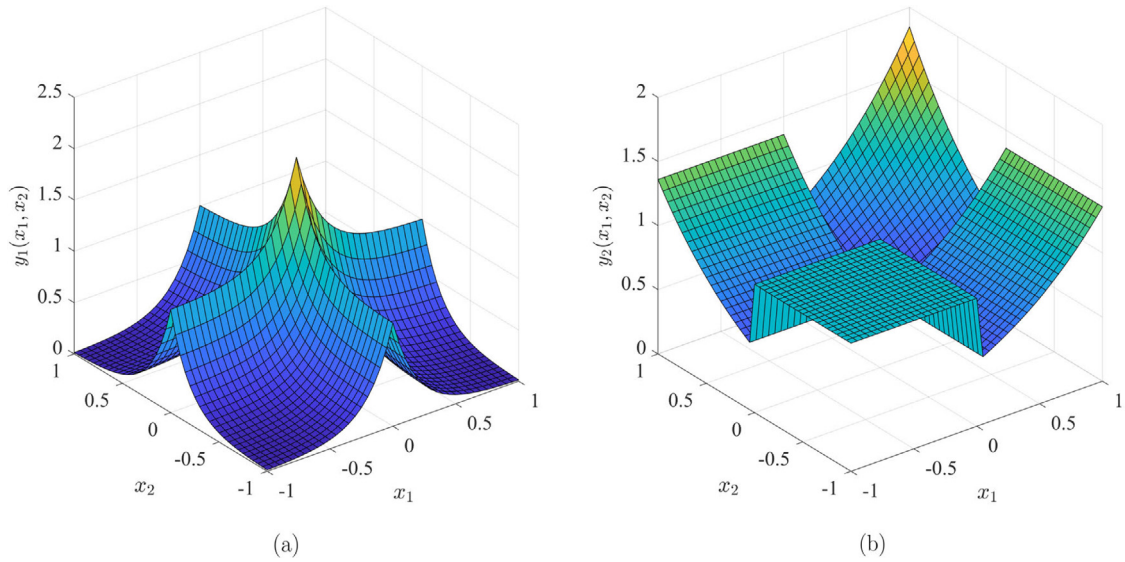
Fig. 6. Mathematical two-dimensional functions in Example 1-1: (a) nonsmooth function $y_1(x_1, x_2)$; (b) discontinuous function $y_2(x_1, x_2)$.

For $N = 2$, the variances of $y_1(X_1, X_2)$ and $y_2(X_1, X_2)$ were calculated exactly as 1.2349×10^{-1} and 6.2098×10^{-2}, respectively. The variances were then estimated by sparse grids with the Clenshaw–Curtis quadrature rule [52,53], tensor-product PCE [27], and bivariate, linear ($S = 2$, $p = 1$) and bivariate, quadratic ($S = 2$, $p = 2$) SDD methods, where all the PCE and SDD coefficients were computed exactly. For $N = 10$, only the discontinuous function $y_2(X_1, \ldots, X_{10})$ was studied as it was expected to be computationally more difficult to encounter than the nonsmooth function. In this case, as the ten-dimensional exact integration for calculating the function expectations was computationally burdensome, the reference variance was estimated by crude MCS with 10^9 samples as 1.1395×10^{-1}. For $N = 10$, PCE was ruled out due to its hefty computational cost. Three SDD methods were therefore employed: 1) univariate, linear ($S = 1$, $p = 1$); 2) bivariate, linear ($S = 2$, $p = 1$); and 3) trivariate, linear ($S = 3$, $p = 1$), where each SDD coefficient was estimated by using crude MCS with 10^7 samples and the definition presented in (21). In order to capture the discontinuity or nonsmoothness, the knots at $x_i = 0$, $i = 1, \ldots, N$, were repeated properly to harness the capabilities of SDD. Eventually, the approximate variances from these methods were obtained, and their relative errors with respect to their reference values were calculated.

Fig. 7 illustrates how the variance errors decay versus the number of basis functions for various methods. Figs. 7(a) and 7(b) correspond to the low-dimensional ($N = 2$) functions $y_1(X_1, X_2)$ and $y_2(X_1, X_2)$, respectively. As observed, both sparse grids and PCE struggle to provide results as accurate as those obtained by the SDD methods of order one or two. Furthermore, the superiority of SDD is more evident in Fig. 7(b) for the discontinuous function. The convergence is also more steep for quadratic ($p = 2$) SDD in comparison with linear ($p = 1$) SDD for both nonsmooth and discontinuous functions. Note that as $S = N = 2$, the spline expansions are exact in terms of the level of interaction between the random variables. In addition, no improvement is delivered by the PCE methods of odd order m in comparison with those of order $m - 1$, since the original functions are both even, as depicted in Fig. 6. According to Fig. 7(c), the high-dimensional stochastic function $y_2(X_1, \ldots, X_{10})$ is very challenging for UQ, as the sparse grids method requires prohibitively many levels to estimate its variance. SDD, however, is efficiently convergent with respect to the increase in the number of subintervals and the truncation parameter S by using only linear ($p = 1$) basis functions. Obviously, for a given number of basis functions, the SDD methods are substantially more accurate than sparse grids. Recall that there are errors in the SDD coefficients and the reference variance estimated by crude MCS. Moreover, each SDD curve reaches a plateau because $S < N$ in this case, as opposed to the cases observed in Figs. 7(a) and 7(b), where there was a monotonic decrease in the errors because S was set equal to N. Nevertheless, although selecting an SDD method with a higher B-spline order, a denser knot vector, or a larger truncation parameter S would have reduced the amount of error further in this problem, reaching a trade-off between accuracy and efficiency for large-scale problems is crucial. This is because the more the number
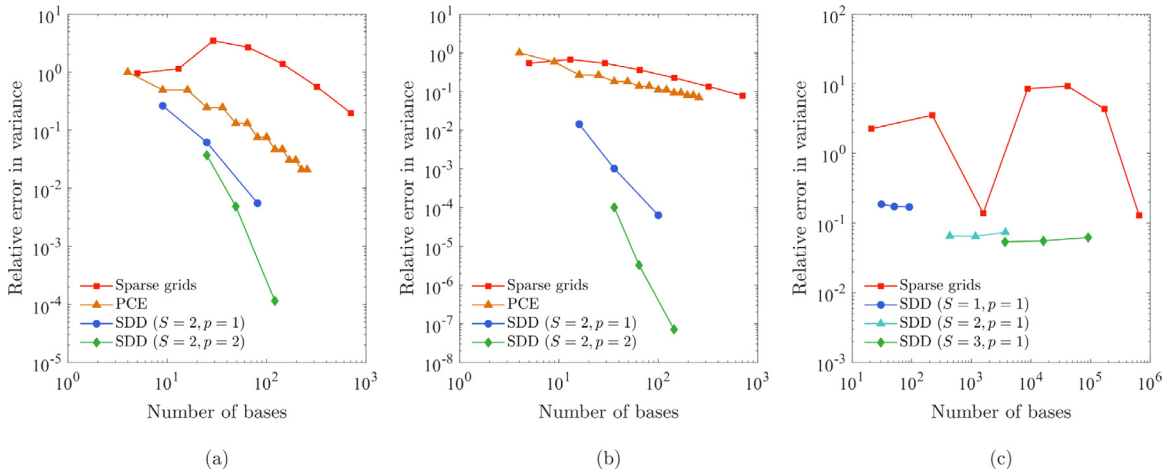
Fig. 7. Variance error analysis for the mathematical functions with different numbers of random variables in Example 1-1: (a) $y_1(X_1, X_2)$; (b) $y_2(X_1, X_2)$; and (c) $y_2(X_1, X_2, \ldots, X_{10})$.

of subintervals or the higher the B-spline order, the more daunting the coefficient estimation process becomes, due to the increase in the number of basis functions.

It is noteworthy that in Fig. 7, the number of bases is associated with the number of quadrature points for sparse grids integration, the maximum polynomial order for PCE, or the number of subintervals for a given B-spline order in SDD. For instance, for $N = 2$, sparse grids has at most seven levels, PCE has an order of at most fifteen, and there are two, four, or eight subintervals in the SDD calculations. Overall, Example 1–1 shows the superiority of SDD over the popular PCE and sparse grids methods in terms of the accuracy, while handling low- or high-dimensional stochastic functions that are harsh in terms of regularity.

Note that a fundamental requirement of most UQ methods, including sparse-grid methods, is that the output function of interest is square-integrable with respect to the probability measure of input random variables. Bearing in mind that sparse grids do not generally perform well when encountering discontinuous functions, as commented by Barthelmann et al. [54], the proposed SDD method performed much better for such functions by employing measure-consistent B-splines, provided that the knot vectors are judiciously chosen. Conveniently, the basis functions of SDD are orthogonal and measure-consistent, which are the principal reasons why SDD performs better than sparse grids, even when compactly-supported B-splines can be used in both approximations. Moreover, the separable variable structure chosen in the first function in this example is merely for restricting the degree of interaction in SDD to be at most two ($S \leq 2$), which is also the same as the dimension ($N = 2$) of the UQ problem. The weaker two-variable interaction selected does not penalize the sparse grid method, as the foundational idea of sparse grids can be traced to the referential dimensional decomposition (RDD) [32] of a high-dimensional function. By contrast, SDD is rooted in ADD, which is generally superior to RDD [32]. Therefore, contrasting the results of SDD and sparse grids predicated on RDD for a function with a separable variable structure is informative in this example.

6.1.2. Example 1-2: Smooth function

Consider an elementary transformation function

$$y(\mathbf{X}) = \sqrt{1 + \mathbf{X}^\mathsf{T}\mathbf{X}/2},$$

where $\mathbf{X} = \{X_1, \ldots, X_N\}$ is a random vector comprising independent, identical random variables X_i, $i = 1, \ldots, N$, defined on $[-1, 1]$ with a uniform probability measure. The objective is to evaluate the variance of $y(\mathbf{X})$ for the cases of $N = 1, 2, \ldots, 6$.

To solve the problem by SDD, N was varied from 1 to 6, and S was varied from 1 to N considering a fixed quadratic ($p = 2$) SDD method with four subintervals ($I = 4$) in its knot vector with uniformly spaced simple knots. According to the theory of SDD, S cannot be larger than N. For all cases, the reference variances were estimated by Gauss–Legendre numerical integration with 16 Gauss points in each coordinate direction i, $i = 1, \ldots, N$.

Table 1
Relative variance errors computed by various SDD methods in Example 1–2.

S	N					
	1	2	3	4	5	6
1	4.9807×10^{-6}	1.3779×10^{-3}	2.2648×10^{-3}	2.8327×10^{-3}	3.1887×10^{-3}	3.4027×10^{-3}
2		3.3902×10^{-6}	1.6025×10^{-5}	3.0776×10^{-5}	4.3366×10^{-5}	5.2735×10^{-5}
3			2.1928×10^{-6}	1.5757×10^{-6}	1.3268×10^{-6}	1.2484×10^{-6}
4				1.2836×10^{-6}	5.9719×10^{-7}	7.8650×10^{-8}
5					5.8795×10^{-7}	5.2506×10^{-8}
6						5.2128×10^{-8}

Table 2
Numbers of basis functions in various SDD methods in Example 1–2.

| S	N					
	1	2	3	4	5	6
1	6	11	16	21	26	31
2		36	91	171	276	406
3			216	671	1526	2906
4				1296	4651	12281
5					7776	31031
6						46656

Moreover, the SDD expansion coefficients were approximately obtained by using a three-point Gauss–Legendre numerical quadrature rule on each knot vector subinterval.

Table 1 lists the relative variance errors computed by various SDD methods. For a given N, the variance estimates converge with the increase in S, but the errors are *not* zero when $S = N$, since the original function is not a spline or polynomial. Moreover, the SDD methods enjoy an impressive accuracy by employing only quadratic ($p = 2$) B-splines, while having such low-order basis functions makes the subsequent numerical integration process so straightforward that a three-point quadrature rule estimates the expansion coefficients accurately. Despite having low-order basis functions in the SDD methods, their relative errors listed in Table 1 are tremendously small, especially for S as small as 2. The implication is that a bivariate approximation of the original function practically suffices to provide accurate results, even when $N = 6$.

Table 2 lists the numbers of basis functions in various SDD methods used in Example 1–2. Comparing Tables 1 and 2 delivers a big picture about SDD. A take-away message is that increasing the truncation parameter S in SDD may not be needed, as it may unnecessarily increase the number of basis functions that not only contribute marginally to the expansion, but also make the computations drastically intensive. For $N = 6$ as an example, an error of 5.2128×10^{-8} is recorded having 46656 basis functions when $S = 6$. However, a reasonably small error of 5.2735×10^{-5} is obtained using a bivariate ($S = 2$) approximation with only 406 basis functions, which is tremendously cheaper to achieve. Example 1–2 clarifies why to start approaching a stochastic problem with $S = 1$ or 2 in SDD is a proper initiative. This will be re-visited in Examples 2 and 3. The proposed SDD method is able to efficiently and accurately handle high-dimensional problems by expressing them in terms of linear combinations of low-variate, low-order components. This feature is vital when facing large-scale stochastic problems, such as the one presented in Example 3.

6.2. Example 2: Stress analysis of a two-dimensional connecting rod ($N = 8$)

Consider a two-dimensional connecting rod, as shown in Fig. 8(a) in a stochastic linear elasticity problem. As the boundary conditions, the connecting rod is fixed on its right end and undergoes a prescribed displacement D_L on its left end. The material is considered isotropic. There are eight random variables ($N = 8$) in the problem. In particular, the lengths d_1, d_2, and d_3, the radii r_L and r_R, the prescribed displacement D_L, the Young's modulus E, and the Poisson's ratio v are all random. Essentially, $\mathbf{X} = \{d_1, d_2, d_3, r_L, r_R, D_L, E, v\}$. The statistical details of
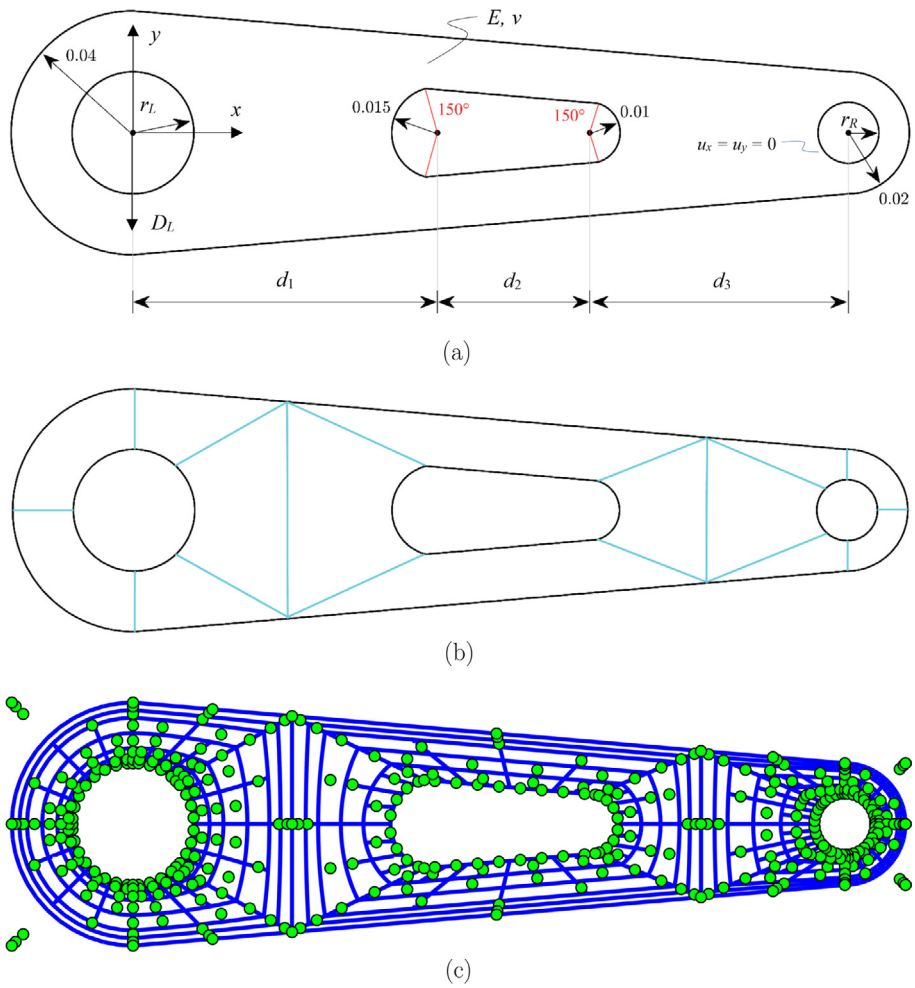
Fig. 8. A two-dimensional connecting rod in Example 2: (a) the geometry, boundary conditions, and material properties; (b) the 14-patch domain; (c) the IGA finite element model with analysis-suitable T-splines and control points. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Statistical properties of the random input variables in Example 2.

Random variable	Type	Bounds	Mean	Standard deviation	Units
d_1	uniform	[0.09, 0.11]	0.1	0.0058	m
d_2	uniform	[0.045, 0.055]	0.05	0.0029	m
d_3	uniform	[0.0765, 0.0935]	0.085	0.0049	m
r_L	uniform	[0.019, 0.021]	0.02	5.7735×10^{-4}	m
r_R	uniform	[0.0095, 0.0105]	0.01	2.8868×10^{-4}	m
D_L	uniform	$[7.5 \times 10^{-5}, 1.25 \times 10^{-4}]$	1×10^{-4}	1.4434×10^{-5}	m
E	uniform	[165.6, 248.4]	207	23.90	GPa
ν	uniform	[0.255, 0.345]	0.3	0.026	–

the independent random variables are listed in Table 3. The random variables in this problem all follow uniform probability distribution, with the units, bounds, mean values, and standard deviation values defined.

To model the geometry in IGA, the domain was decomposed into 14 patches, as indicated in Fig. 8(b), where the patch interfaces are illustrated by cyan lines. Furthermore, the patches were coupled in a C^0-continuous manner, as discussed in Section 3.5. Fig. 8(c) demonstrates the IGA finite element model with quadratic ($p_k = 2, \ k = 1, 2$)

Table 4
Numbers of basis functions and regression samples for the SDD methods in Example 2.

SDD method	Number of basis functions	Number of IGA samples for regression
Univariate SDD ($p = 1$, $I = 2$)	17	68
Univariate SDD ($p = 1$, $I = 4$)	33	132
Univariate SDD ($p = 2$, $I = 4$)	41	164
Bivariate SDD ($p = 1$, $I = 2$)	129	516
Bivariate SDD ($p = 1$, $I = 4$)	481	1924
Bivariate SDD ($p = 2$, $I = 4$)	741	2964

analysis-suitable T-spline, which includes 270 elements and 423 control points. The control points are shown by closed green circles.

To induce stresses in the domain, the left end of the connecting rod moves downward by amount D_L, which is random, as stated previously. One may assume that this displacement is, in reality, applied by using a pin in contact with the left end. In the finite element IGA model, however, the displacement D_L was applied to all the control points that control the left end circle. Similarly, the boundary conditions $u_x = u_y = 0$ were applied to the right end by constraining the displacements of the control points of the right end circle in both x and y directions.

For UQ of the displacements and stresses, the problem was solved by crude MCS with 20,000 samples (IGA) and univariate ($S = 1$) and bivariate ($S = 2$) truncations of the SDD ($p = 1$, $I = 2$), SDD ($p = 1$, $I = 4$), and SDD ($p = 2$, $I = 4$) methods with the B-spline order p and the number of subintervals I equal in all coordinate directions of the stochastic domain. Moreover, the interior knots were evenly spaced in the knot vectors. The expansion coefficients of the SDD methods were computed by the SLS method via (28) with the number of regression samples being four times the number of basis functions used in the SDD methods. Recall that in an SDD method, the number of basis functions is equivalent to the number of expansion coefficients, which is computed by invoking (23). Table 4 lists the numbers of basis functions and regression samples for the SDD methods used in this example. Moreover, the standard deviations of the output variables of interest were computed by using the analytical formula in (25).

6.2.1. Second-moment statistical analysis

Figs. 9 and 10 depict the standard deviation contour plots of root-mean-squared (RMS) displacement and von Mises stress in the connecting rod, obtained by MCS and SDD methods. As the geometry changes randomly, the results have been visualized on the mean input geometry. Evidently, the SDD methods have accurately estimated the results, as there is an outstanding match between the contour plots associated with the MCS and SDD methods. The differences in these plots are indistinguishable to the naked eye, considering that the SDD methods have been able to characterize the second-moment statistical properties of the responses by using far fewer numbers of IGA than the 20,000 used by MCS. In addition, the match between the results clearly shows that the SLS method has accomplished accurate estimation of the SDD expansion coefficients.

Note that although the contour plots related to the RMS displacement are smooth, those of the von Mises stress are not. This is because the solution space in IGA finite element formulation includes T-spline basis functions that are smooth within the patches, but are only C^0-continuous at the patch interfaces. Therefore, the first derivatives of the T-splines, which are connected to the stresses, are C^{-1}-continuous, or discontinuous, on the patch interfaces. Therefore, improvements in interpatch continuity of stresses merit further study.

In UQ applications, standard deviation is generally more challenging to accurately compute than mean, and the higher the statistical moment, the more daunting the estimation becomes. Therefore, with the standard deviations being splendidly computed by the SDD methods, the results corresponding to the mean values have been suppressed for brevity. That being noted, the mean values of maximum RMS displacement and maximum von Mises stress were roughly 1.1×10^{-4} m and 60 MPa, respectively. Based on Figs. 9 and 10, this shows a coefficient of variation (COV) of roughly 15% in the responses.

6.2.2. Cumulative distribution function

The CDF includes all statistical moments of a random variable and is, therefore, computationally demanding to estimate by UQ methods. Typically, the tail region of the CDF is of significant importance for probability of failure
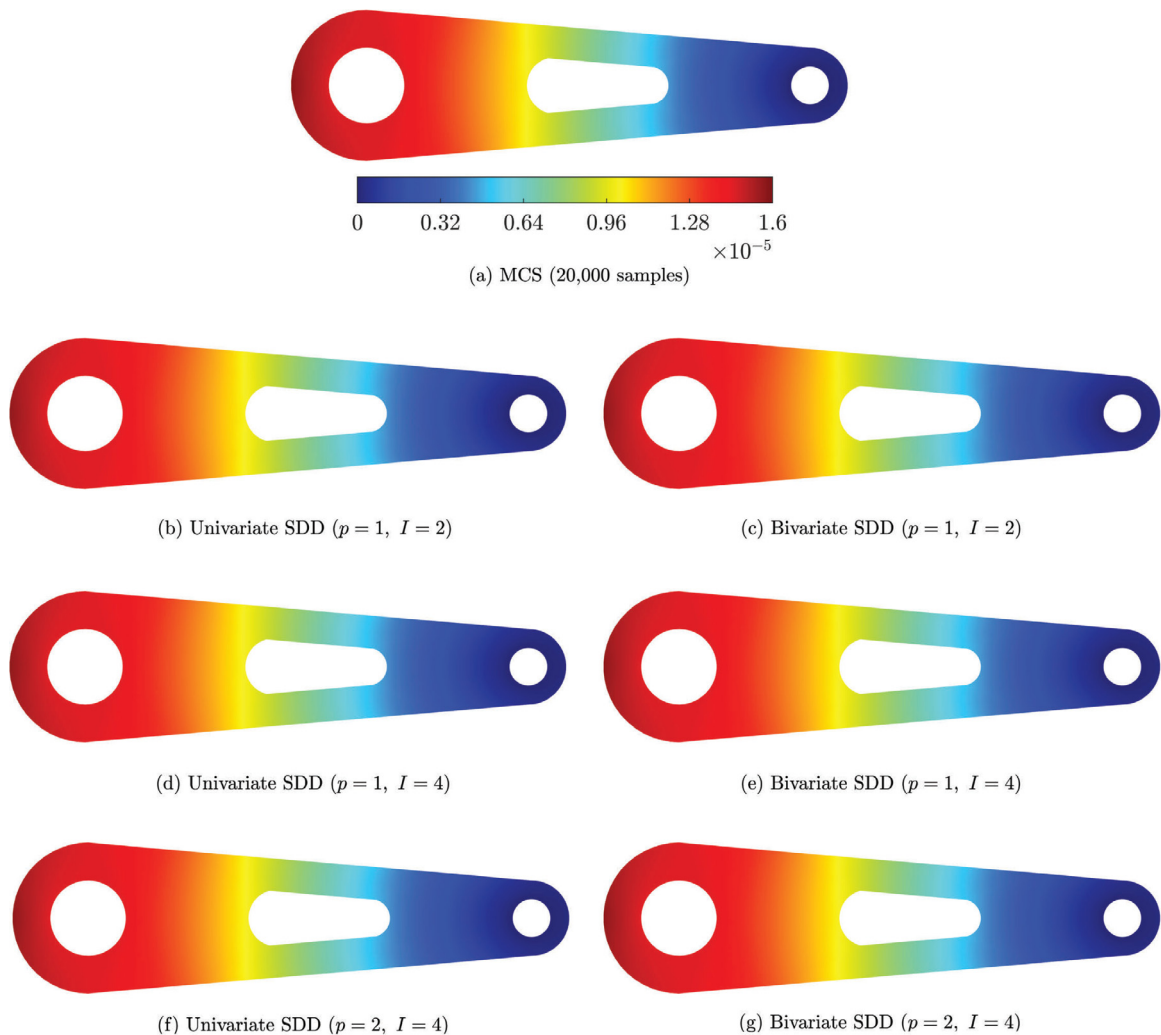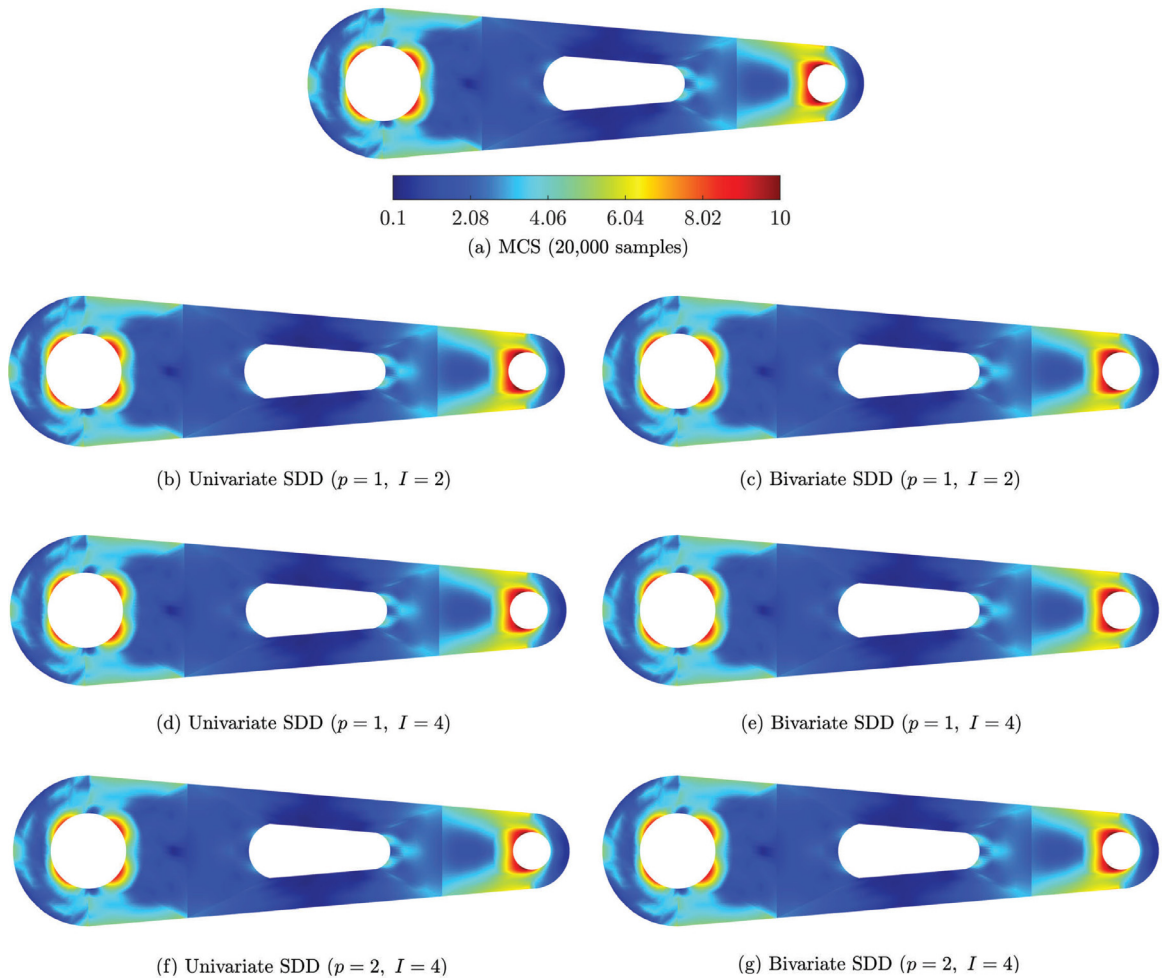
Fig. 9. Standard deviation contour plots of the RMS displacement in m by various methods visualized on the mean input geometry in Example 2.

analysis, probabilistic design purposes, and so forth. Therefore, in many engineering stochastic problems, the CDF of a random output is investigated.

Denote by σ_c the von Mises stress located at the node where the von Mises stress is maximum at mean input. Based on IGA considering mean input, σ_c occurs around the smaller hole on the right-hand side of the connecting rod. However, as the geometric parameters are modeled as random variables, the physical domain of the connecting rod and hence the location of the critical point are random. As a result, σ_c is not only a function of input random variables but also possibly discontinuous. Fig. 11 illustrates the tail region of the CDF of σ_c, estimated by various methods. Note that in this figure, "U" and "B" symbolize the univariate and bivariate SDD methods, respectively. While the accuracy of the SDD estimations improves with the increase in the B-spline order p or the number of subintervals I in both univariate and bivariate SDD approximations, there is a wide gap between the univariate and bivariate cases. This reveals the need for bivariate SDD truncation in the CDF analysis, although for second-moment statistical analysis, univariate SDD approximation sufficed, according to Figs. 9 and 10. Moreover, the dimensionwise expansion of the random function $\sigma_c(\mathbf{X})$ according to (22) has proven to have a significant impact in accurately and efficiently solving this stochastic problem, where a bivariate truncation at $S = 2$ can simply capture the major contribution of the terms in the expansion. This observation is consistent with the results reported in Example 1–2 for a simple mathematical function.

Fig. 10. Standard deviation contour plots of the von Mises stress in MPa by various methods visualized on the mean input geometry in Example 2.

6.3. Example 3: Fatigue durability analysis of a three-dimensional gear ($N = 54$)

The third numerical example entails solving a 54-dimensional, industrial-type, stochastic problem of a three-dimensional mechanical gear, where the uncertainties emanate from a random field and several random variables. As mentioned before, many UQ methods struggle, to say the least, when solving high-dimensional stochastic problems, while methods like PCE and SCE are ruled out due to their tremendous computational burden. As a result, the goal in this example is to efficiency and accurately approximate the second-moment statistical properties of the stresses and fatigue durability of the mechanical gear by the proposed SDD method.

6.3.1. Problem definition

Consider a three-dimensional mechanical Bevel Gear, simply referred to as gear in this example, as shown in Fig. 12. This type of gear is cone-shaped and typically transmits power between two intersecting axles in many engineering devices and heavy duty machinery. Fig. 12(a) illustrates the front view of the gear with some geometrical dimensions shown and the loaded region depicted with a red rectangle. In Fig. 12(b), the loaded region is magnified, and the distributed surface loads are indicated with their directions noted with respect to the $+x$ direction. Fig. 12(c) demonstrates the rear view of the gear, with more dimensions and displacement boundary conditions $u_x = u_y = u_z = 0$ all around the interior surface of the gear, where the gear is fixed to a shaft by means
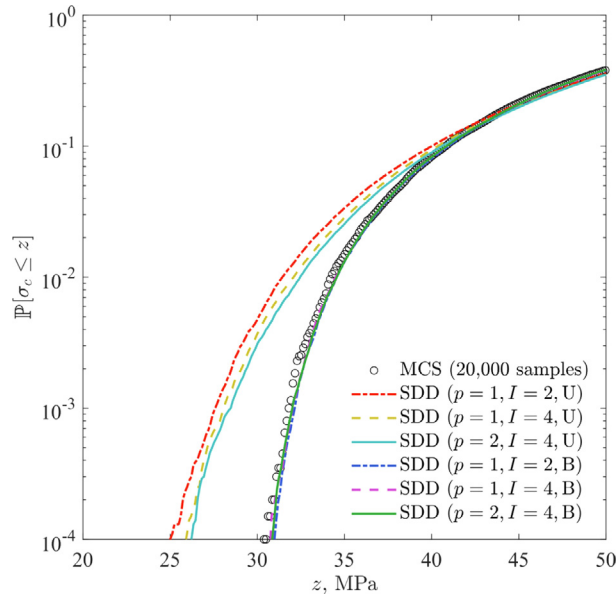
Fig. 11. The tail region of the CDF of σ_c, estimated by various methods in Example 2.

of a key. Last but not least, the side view of the gear is shown in Fig. 12(d) with two more geometrical dimensions needed to completely define the geometry.

The gear in this example is made of cast steel. For material properties, set the Poisson's ratio as $\nu = 1/3$, and let a homogeneous lognormal random field $E(\mathbf{z}; \cdot)$ as

$$E(\mathbf{z}; \cdot) = C_\alpha \exp[\alpha(\mathbf{z}; \cdot)] \tag{30}$$

with the mean $\mu_E = 30 \times 10^6$ psi and COV of $\nu_E = 0.1$ define the Young's modulus of the material. In (30), $\mathbf{z} \in \mathcal{D}$ represents a physical point. Moreover,

$$C_\alpha = \frac{\mu_E}{\sqrt{1 + \nu_E^2}}$$

and $\alpha(\mathbf{z}; \cdot)$ is a homogeneous Gaussian random field with *zero* mean and exponential covariance function

$$\Gamma(\mathbf{z}, \mathbf{z}') = \sigma^2 \exp\left(-\frac{\|\mathbf{z} - \mathbf{z}'\|}{bL}\right), \quad \mathbf{z}, \mathbf{z}' \in \mathcal{D} \subset \mathbb{R}^3,$$

where $\sigma^2 = 0.01$, $b = 0.2$, and $L = 2$ in.

The mechanical gears typically withstand cyclic loading and contact forces, which lead to the fatigue failure in their teeth. In this work, contact effects are ignored. For fatigue simulation, however, let the loads shown in Fig. 12(b) vary between *zero* and their specified value in a cyclic manner and with a constant amplitude. Let the four parameters – (1) fatigue strength coefficient σ_f', (2) fatigue strength exponent b, (3) fatigue ductility coefficient ε_f', and (4) fatigue ductility exponent c – specify the fatigue properties of the cast steel material. Table 5 lists the statistical properties of the fatigue parameters defined. Note that, with some simplification, these parameters have been considered to be mutually independent. They follow lognormal probability distributions and are defined on an unbounded domain $[0, +\infty)$.

With the fatigue parameters defined, implement the strain-life method for fatigue durability analysis [55], which uses the Coffin–Manson–Morrow equation to evaluate the fatigue crack initiation life N_f by solving

$$\frac{\Delta\varepsilon}{2} = \frac{\sigma_f' - \sigma_m}{E}(2N_f)^b + \varepsilon_f'(2N_f)^c$$

for N_f, where $\Delta\varepsilon$, σ_m, and E are the equivalent strain range, equivalent mean stress, and Young's modulus at a given point. Here, the stresses and strains must first be evaluated analytically or by numerical methods. For a
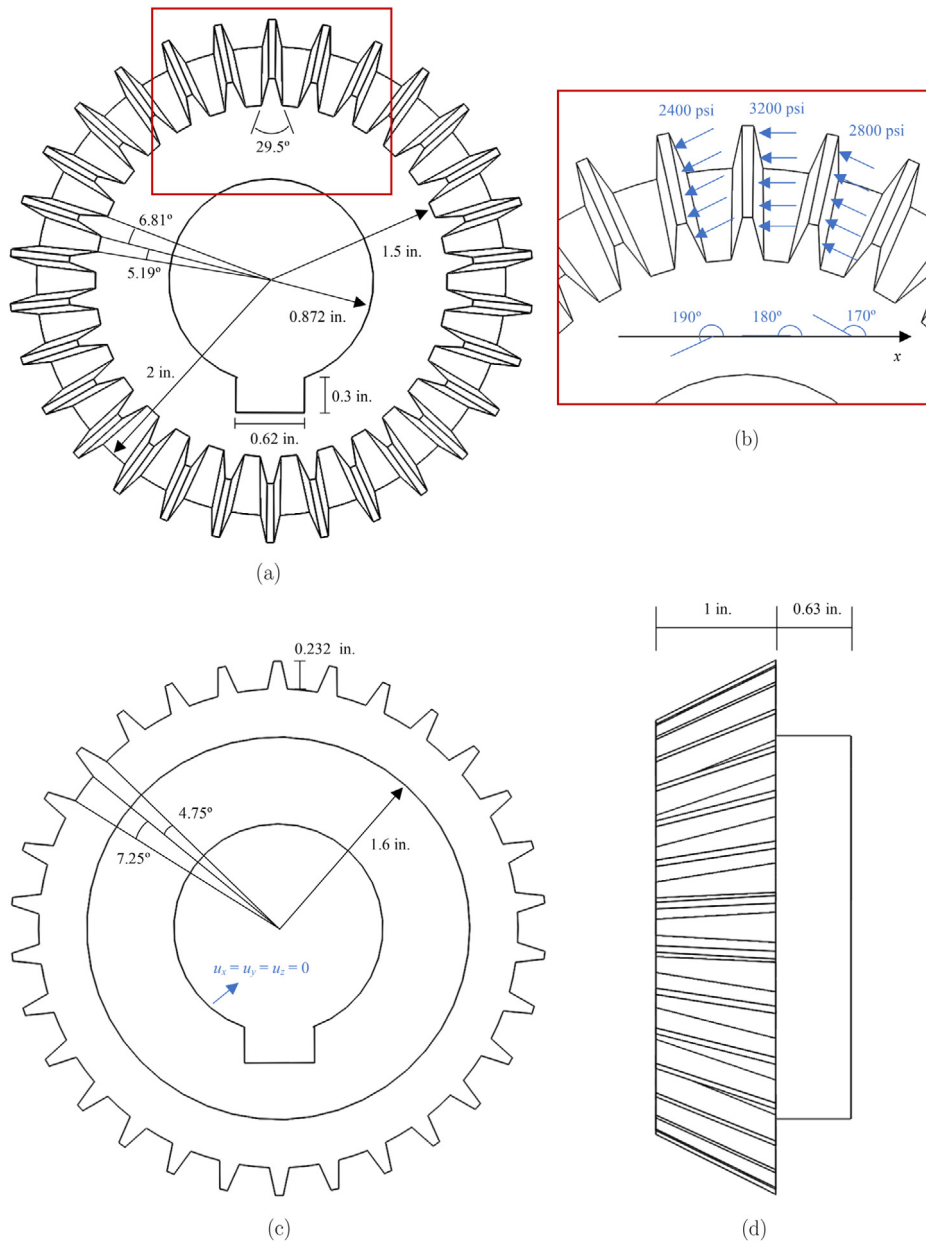
Fig. 12. Gear geometry, dimensions, loads, and boundary conditions in Example 3: (a) front view with the loaded region depicted by a rectangle; (b) magnified view of the loaded region with the direction of the distributed surface loads shown with respect to the $+x$ direction; (c) rear view and displacement boundary conditions; and (d) side view.

standard fatigue analysis, the largest principal stresses and strains are typically used as equivalent stress and strain measures, respectively, but how to calculate σ_m and $\Delta\varepsilon$ by using the Ramberg–Osgood parameters [55] is general knowledge in solid mechanics and is, therefore, omitted from this paper.

6.3.2. Computational modeling

As the gear model has a complex geometry, it is most convenient to be decomposed into patches. Therefore, a total of 150 patches were conformed with C^0-continuity over the patch boundaries to model the whole domain by a T-mesh. Fig. 13(a) illustrates two three-dimensional views of the model geometry, precisely constructed in multipatch

Table 5
Statistical properties of the fatigue parameters in Example 3.

Parameter	Type	Bounds	Mean	Standard deviation	Units
σ'_f	Lognormal	$[0, +\infty)$	193190	28979	psi
b	Lognormal	$[0, +\infty)$	-0.1085	0.01085	–
ε'_f	Lognormal	$[0, +\infty)$	0.375	0.05625	–
c	Lognormal	$[0, +\infty)$	-0.6354	0.06354	–
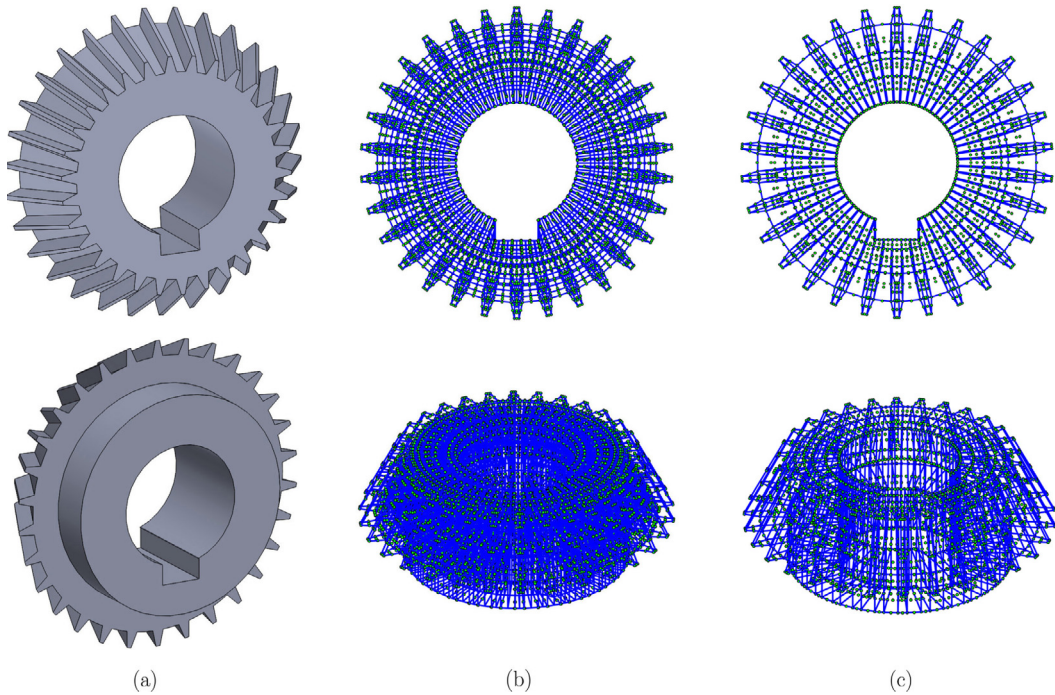
Fig. 13. Gear model in Example 3: (a) two three-dimensional views; (b) three-dimensional and top views of the final IGA finite element model $\mathcal{T}_R$ with analysis-suitable T-splines and control points; and (c) three-dimensional and top views of the IGA base mesh $\mathcal{T}_0$ and its control points. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

IGA by using three-dimensional, analysis-suitable, quadratic ($p_k = 2$, $k = 1, 2, 3$) T-splines. The three-dimensional and top views of the IGA finite element model are demonstrated in Fig. 13(b) and include 1950 elements and 8680 control points. The control points are illustrated by closed green circles. The final IGA mesh $\mathcal{T}_R$ in Fig. 13(b) and its control points were obtained by using the Bézier extraction operator and the base mesh $\mathcal{T}_0$ depicted in Fig. 13(c), as discussed in Section 3.4. The mesh density was deemed adequate for analysis and was not refined further.

The stochastic problem involves a random field and four random variables, as defined previously. The K–L expansion method, described in Section 2.2.2, was employed for random field discretization. Note that for a low correlation length parameter b in the covariance function of a random field, a large number of terms is suggested to be retained in the K–L expansion to accurately discretize the original random field. In this problem, as $b = 0.2$ was considered relatively small, the K–L expansion was truncated at $N' = 50$. Moreover, the eigensolutions $\{\lambda_i, \phi_i(\mathbf{z})\}_{i=1,\dots,50}$, needed to write (3), were numerically computed from (1) by using the collocation IGA method recently proposed for arbitrary multipatch domains [37].

Eventually, there were 50 standard Gaussian random variables stemming from the random field discretization, plus four additional random variables to describe the fatigue behavior of cast steel. Hence, there was a total of 54 ($N = 54$) independent random variables in this large-scale problem, which were all defined on unbounded domains.

For the UQ analysis, the problem was solved by crude MCS with 10,000 samples (IGA) to obtain a benchmark solution. In addition, three univariate ($S = 1$) SDD methods were used, including SDD ($p = 1$, $I = 2$), SDD

($p = 1$, $I = 4$), and SDD ($p = 2$, $I = 4$) methods, which involved 109, 217, and 271 basis functions, respectively, in their expansions. Recall that p and I refer to the B-spline order and the number of subintervals, respectively, which were set equal in all coordinate directions of the stochastic domain. Again, the interior knots were evenly spaced in the knot vectors. The SLS regression method, discussed in Section 5.6.2, was exploited to estimate the expansion coefficients practically by setting the number of samples as four times the number of basis functions for each SDD method. Consequently, 436, 868, and 1084 samples (IGA) were needed to construct the SDD ($p = 1$, $I = 2$), SDD ($p = 1$, $I = 4$), and SDD ($p = 2$, $I = 4$) surrogate models, respectively.

Note that SDD, by theory, requires the random variables to be defined on bounded domains. Therefore, all random variables were properly transformed to a truncated normal probability measure on $[-4, 4]$ with $\sigma = 1$. As there are many options for target probability measure for transformation, it is recommended to choose the one that yields a relatively mild, as opposed to heavily nonlinear, mapping from the original to the target distribution. From prior experience [33], the truncated normal distribution was deemed a good choice as such a target probability measure on a bounded domain.

6.3.3. Results

In this section, the second-moment statistical properties of the stress and fatigue life are investigated by the proposed methods over the whole gear domain. Prior to the UQ analysis, the random field was discretized, for which the results have been omitted for brevity. Readers interested in the K–L expansion by the isogeometric collocation method on arbitrary multipatch domains are referred to the authors' previous work [37]. Thus, the Young's modulus E in (29) would be evaluated via the K–L approximation of the random field at each point of the physical domain.

Figs. 14 and 15 demonstrate the mean and standard deviation contour plots, respectively, of the largest principal stress, estimated by various methods. Both figures consistently show that the low-variate, low-order SDD methods are able to accurately estimate the mean and standard deviation of the largest principal stress by using, at most, roughly one tenth of the samples needed for estimation by crude MCS. Any difference between the contour plots is indiscernible to the naked eye.

In Fig. 14, some negative largest principal stresses are set as *one* to be able to have the logarithmic scale. Moreover, the largest principal stress varies around 9×10^4 psi, which is beyond the typical yield strength of cast steel. Hence, the case of fatigue failure is relevant.

Figs. 16 and 17 depict the mean and standard deviation contour plots, respectively, of the logarithm of the fatigue life, estimated by various methods. Here, as the fatigue life varies roughly in the range of 10^7 to 10^{50}, the logarithm of the fatigue life is plotted in logarithmic scale to have a better visual over the results. Based on the numerical results, the fatigue durability of the gear is approximately 10^8 cycles, particularly, in the loaded region with a standard deviation of roughly 10^7 cycles, yielding a COV of about 10%. Again, the results obtained by the MCS and low-order SDD methods match very well, while the SDD methods are tremendously more economical. This reveals the great impact of the dimensionwise decomposition of the random output of interest, in terms of low-order measure-consistent orthonormalized B-splines, in effectively handling a 54-dimensional stochastic problem. Furthermore, the SLS regression method proves reliable in accurately estimating the SDD expansion coefficients.

Overall, the numerical examples indicate the efficacy of the proposed SDD method in conjunction with IGA on arbitrary multipatch domains, where a high-dimensional, complex function of input random variables is approximated by a set of low-variate component functions of B-splines in a systematic manner.

While the user-friendly implementation of IGA with unstructured three-dimensional T-splines in a commercial package is to be expected in the near future, the proposed SDD-mIGA method has proved to be highly desirable, especially for high-dimensional stochastic problems. The authors, therefore, envision the need to incorporate such a powerful UQ method in the common commercial packages for probabilistic engineering design and analysis. Furthermore, this paper mainly focused on solving the stochastic BVPs in linear elasticity by applying the proposed SDD-mIGA method. Having said so, approaching other types of engineering problems would be an interesting topic to pursue in the future, especially when dealing with large-scale stochastic problems.

7. Conclusion

This work debuted a new stochastic method, referred to as SDD-mIGA, to solve linear elasticity stochastic BVPs on arbitrary multipatch domains. The method followed the standard Galerkin formulation of the problem by using the analysis-suitable T-splines with outstanding approximating power, where the patches were coupled
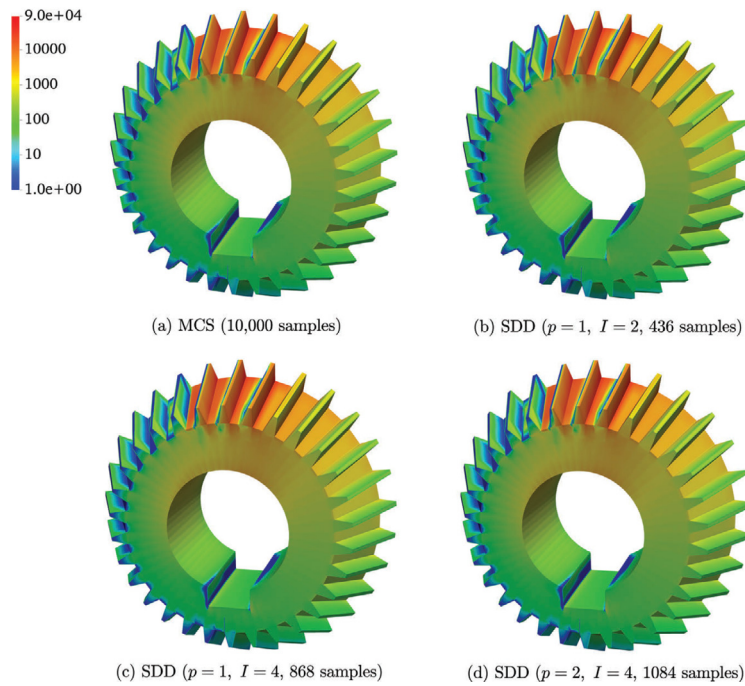
(a) MCS (10,000 samples)   (b) SDD ($p = 1$, $I = 2$, 436 samples)

(c) SDD ($p = 1$, $I = 4$, 868 samples)   (d) SDD ($p = 2$, $I = 4$, 1084 samples)

Fig. 14. Mean contour plots for the largest principal stress in psi, estimated by various methods in Example 3.

(a) MCS (10,000 samples)   (b) SDD ($p = 1$, $I = 2$, 436 samples)

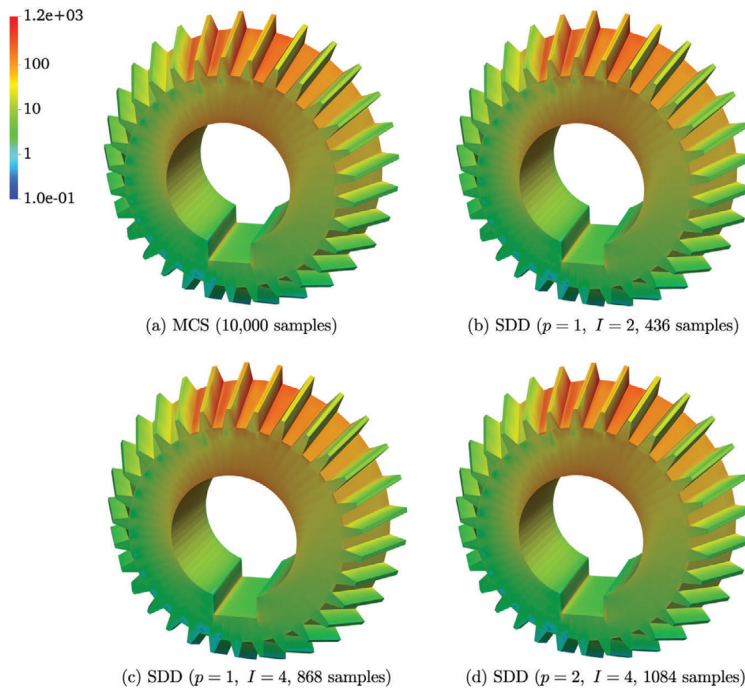(c) SDD ($p = 1$, $I = 4$, 868 samples)   (d) SDD ($p = 2$, $I = 4$, 1084 samples)

Fig. 15. Standard deviation contour plots for the largest principal stress in psi, estimated by various methods in Example 3.

properly and in a water-tight manner. Moreover, the Bézier extraction operator was utilized to efficiently refine the T-meshes used in the multipatch isogeometric analysis from a coarse mesh. Last but not least, the SDD method was integrated with the multipatch IGA framework for UQ purposes by expressing a high-dimensional, complex random
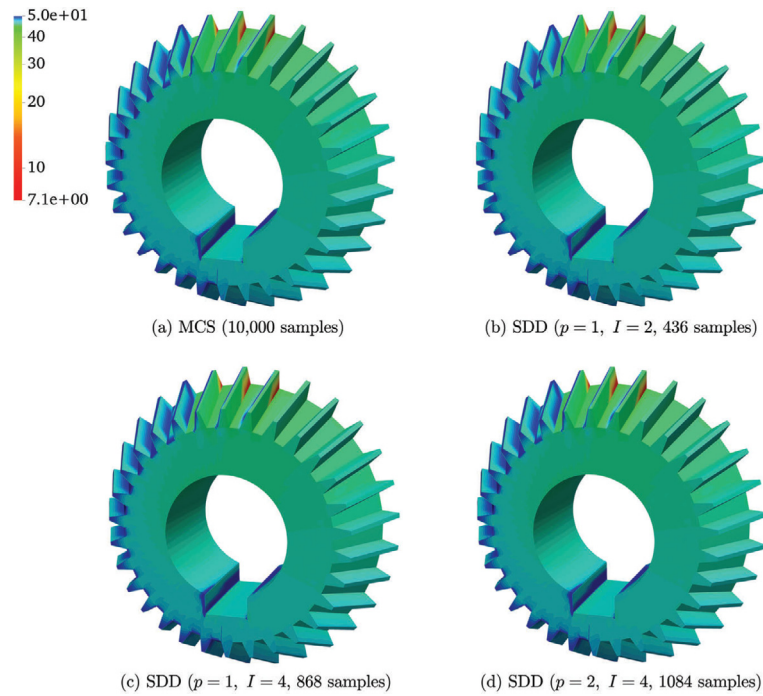
(a) MCS (10,000 samples)

(b) SDD ($p = 1$, $I = 2$, 436 samples)

(c) SDD ($p = 1$, $I = 4$, 868 samples)

(d) SDD ($p = 2$, $I = 4$, 1084 samples)

Fig. 16. Mean contour plots for logarithm of the fatigue life in cycles, estimated by various methods in Example 3.

(a) MCS (10,000 samples)

(b) SDD ($p = 1$, $I = 2$, 436 samples)

(c) SDD ($p = 1$, $I = 4$, 868 samples)

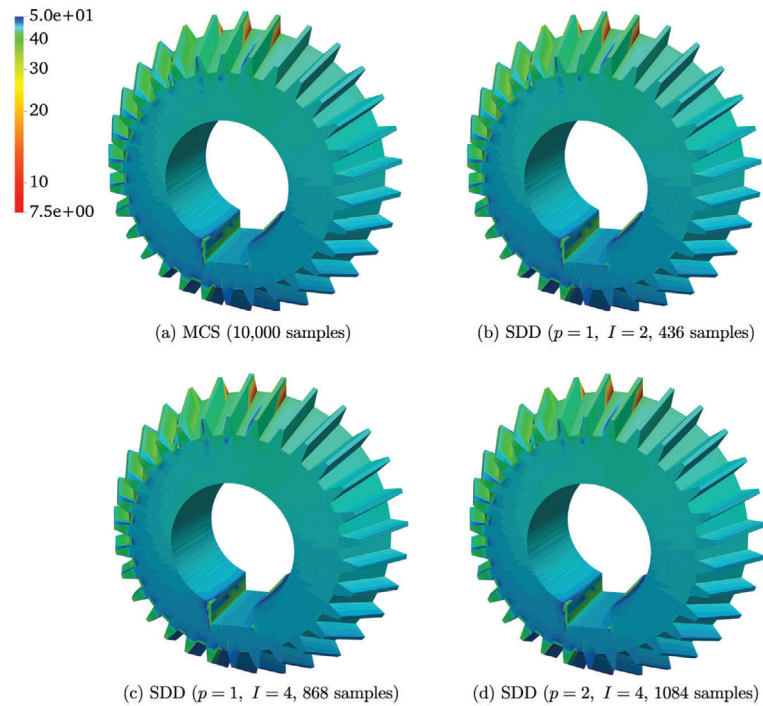(d) SDD ($p = 2$, $I = 4$, 1084 samples)

Fig. 17. Standard deviation contour plots for logarithm of the fatigue life in cycles, estimated by various methods in Example 3.

response in terms of a Fourier-like expansion with measure-consistent orthonormalized B-splines as its ingredients. The dimensionwise expansion of the function led to a polynomial computational complexity, thereby alleviating

the curse of dimensionality to a significant extent. The second-moment statistical properties were evaluated via the analytical formulae derived. The SDD-mIGA method featured common B-spline and T-spline ingredients in geometrical modeling, random field discretization, stress analysis, and uncertainty quantification, thereby harnessing the approximating power of such basis functions. Moreover, the method used the SLS regression method to cheaply but accurately estimate the expansion coefficients.

Three numerical examples were presented to investigate the efficacy of the proposed method in solving stochastic BVPs in linear elasticity, including a 54-dimensional problem of a mechanical gear. The numerical analyses revealed that a low-variate, low-order SDD-mIGA approximation effectively estimates the probabilistic solution not only at an affordable computational cost, but at a desirable one.

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.

Appendix. Bézier extraction operator

The Bézier extraction makes access to IGA data tremendously more convenient and has been developed for T-splines of arbitrary degree [20]. The purpose here is to enable standard assembly of IGA data for simple evaluation routines [21]. In so doing, T-spline functions are written in terms of a linear combination of Bernstein polynomials, which obviously form a basis on $[-1, 1]$. Let p_k, $k = 1, \ldots, d$, denote the polynomial orders of a given T-mesh. The univariate Bernstein polynomials are defined as

$$\tilde{V}_k^j(\eta_k) = \frac{1}{2^{p_k}} \binom{p_k}{j-1} (1 - \eta_k)^{p_k - (j-1)} (1 + \eta_k)^{j-1}, \quad j = 1, \ldots, p_k + 1,$$

where $\eta_k \in [-1, 1]$. Map η_k to ξ_k through $\xi_k : [-1, 1] \rightarrow [0, 1]$, $k = 1, \ldots, d$ and denote by $V_k^{j,e}(\xi_k)$ the Bernstein polynomial of degree $(j - 1)$ in coordinate direction k on element e, $e = 1, \ldots, n_e$. In a d-dimensional problem, there are

$$n_f = \prod_{k=1}^{d} (p_k + 1)$$

d-variate Bernstein polynomials that are nonzero on element e, constructed by tensor product. Form an n_f-dimensional set as

$$\{V_j^e(\boldsymbol{\xi})\}_{j=1,\ldots,n_f} := \left\{ \prod_{k=1}^{d} V_k^{j_k,e}(\xi_k) \right\}_{j_k=1,\ldots,p_k+1, \ k=1,\ldots,d}, \tag{A.1}$$

where $j_k = 1, \ldots, p_k + 1$, $k = 1, \ldots, d$ and $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_d)$ refers to the parametric space discussed in Section 3.1. Arrange the elements of the set in (A.1) in a column vector $\mathbf{V}^e(\boldsymbol{\xi})$ as

$$\mathbf{V}^e(\boldsymbol{\xi}) = \left\{ \begin{array}{c} V_1^e(\boldsymbol{\xi}) \\ \vdots \\ V_{n_f}^e(\boldsymbol{\xi}) \end{array} \right\}.$$

Each and every multivariate T-spline $U^i_{\mathbf{p}, \boldsymbol{\Xi}^i}(\boldsymbol{\xi})$ associated with anchor i, $i = 1, \ldots, n_c$ in the parametric space $\hat{\mathcal{D}}$ can now be written as

$$U^{i,e}_{\mathbf{p}, \boldsymbol{\Xi}^i}(\boldsymbol{\xi}) = \mathbf{C}^{i,e\top} \mathbf{V}^e(\boldsymbol{\xi}), \quad i = 1 = 1, \ldots, n_c, \ e = 1, \ldots, n_e,$$

which essentially is formulated in an element-wise manner. Here, $\mathbf{C}^{i,e}$ is an n_f-dimensional column vector of the expansion coefficients, which can be uniquely and exactly determined. Due to the local support behavior of T-splines, vector $\mathbf{C}^{i,e}$ is generally expected to be sparse.

Up to this point, $\mathbf{C}^{i,e}$ is associated with anchor i and element e. Loop over the elements to obtain an $n_f \times n_e$ matrix as

$$\mathbf{C}^i = \begin{bmatrix} \mathbf{C}^{i,1}, & \cdots & , \mathbf{C}^{i,n_e} \end{bmatrix} \text{ and } \mathbf{V}(\boldsymbol{\xi}) = \begin{bmatrix} \mathbf{V}^1(\boldsymbol{\xi}), & \cdots & , \mathbf{V}^{n_e}(\boldsymbol{\xi}) \end{bmatrix},$$

and denote by

$$\mathbf{U}^i(\boldsymbol{\xi}) = \mathbf{C}^i \mathbf{V}^{\mathsf{T}}(\boldsymbol{\xi})$$

the matrix of multivariate T-splines corresponding to anchor i, which is of dimension $n_f \times n_f$. Make another loop, this time over all anchors, to get

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}^1, & \cdots & , \mathbf{C}^{n_c} \end{bmatrix},$$

and write the key formula for Bézier extraction operation as

$$\mathbf{U}(\boldsymbol{\xi}) = \mathbf{C} \mathbf{V}_t^{\mathsf{T}}(\boldsymbol{\xi}), \tag{A.2}$$

where $\mathbf{C}$ is the Bézier extraction operator matrix of dimension $n_f \times n_c n_e$ and

$$\mathbf{U}(\boldsymbol{\xi}) = \begin{bmatrix} \mathbf{U}^i(\boldsymbol{\xi}), \dots, \mathbf{U}^{n^c}(\boldsymbol{\xi}) \end{bmatrix} \text{ and } \mathbf{V}_t(\boldsymbol{\xi}) = \underbrace{\begin{bmatrix} \mathbf{V}(\boldsymbol{\xi}), \dots, \mathbf{V}(\boldsymbol{\xi}) \end{bmatrix}}_{n_c \text{ times}}$$

of dimension $n_f \times n_c n_f$ and $n_f \times n_c n_e$, respectively.

For more details about the Bézier extraction operator and how it can be conveniently used to evaluate the T-meshes in terms of the types of linear dependence and partition of unity, readers are directed to the works by Scott et al. and May et al. [20,21], respectively.

References

[1] M. Papadrakakis, V. Papadopoulos, Robust and efficient methods for stochastic finite element analysis using Monte Carlo simulation, Comput. Methods Appl. Mech. Engrg. 134 (1996) 325–340.

[2] R.G. Ghanem, P.D. Spanos, Stochastic Finite Elements: A Spectral Approach, Springer, Berlin, 1991.

[3] I. Kaljeviĉ, S. Saigal, Stochastic boundary elements in elastostatics, Comput. Methods Appl. Mech. Engrg. 109 (1993) 259–280.

[4] S. Rahman, B.N. Rao, A perturbation method for stochastic meshless analysis in elastostatics, Internat. J. Numer. Methods Engrg. 50 (2001) 1969–1991.

[5] S. Rahman, H. Xu, A meshless method for computational stochastic mechanics, Int. J. Comput. Methods Eng. Sci. Mech. 6 (2005) 41–58.

[6] F. Xiong, S. Greene, W. Chen, A new sparse grid based method for uncertainty propagation, Struct. Multidiscip. Optim. 41 (2010) 335–349.

[7] M. Motamed, F. Nobile, A stochastic collocation method for the second order wave equation with a discontinuous random speed, Numer. Math. 123 (2013) 493–536.

[8] M. Rehme, D. Pflüger, Stochastic collocation with hierarchical extended B-splines on sparse grids, in: Approximation Theory XVI, Nashville, 2019, 2019.

[9] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Comput. Methods Appl. Mech. Engrg. 194 (2005) 4135–4195.

[10] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs, Isogeometric Analysis: Toward Integration of CAD and FEA, John Wiley & Sons, 2009.

[11] T.W. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCs, ACM Trans. Graph. 22 (3) (2003) 477–484.

[12] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, T.W. Sederberg, Isogeometric analysis using T-splines, Comput. Methods Appl. Mech. Engrg. 199 (2010) 229–263.

[13] M.R. Dörfel, B. Jüttler, B. Simeon, Adaptive isogeometric analysis by local h-refinement with T-splines, Comput. Methods Appl. Mech. Engrg. 199 (2010) 264–275.

[14] A. Buffa, D. Cho, G. Sangalli, Linear independence of the T-spline blending functions associated with some particular T-meshes, Comput. Methods Appl. Mech. Engrg. 199 (2010) 1437–1445.

[15] X. Li, J. Zheng, T.W. Sederberg, T.J.R. Hughes, M.A. Scott, On linear independence of T-spline blending functions, Comput. Aided Geom. Design 29 (2012) 63–76.

[16] M.A. Scott, X. Li, T.W. Sederberg, H.T.J. R., Local refinement of analysis-suitable T-splines, Comput. Methods Appl. Mech. Engrg. 213–216 (2012) 206–222.

[17] L. da Veiga, A. Buffa, G. Sangalli, R. Vázquez, Analysis-suitable T-splines of arbitrary degree: definition, linear independence, and approximation properties, Math. Models Methods Appl. Sci. 23 (11) (2013) 1979–2003.

[18] E.J. Evans, M.A. Scott, X. Li, D.C. Thomas, Hierarchical T-splines: Analysis-suitability, bèzier extraction, and application as an adaptive basis for isogeometric analysis, Comput. Methods Appl. Mech. Engrg. 284 (2015) 1–20.

[19] X. Wei, Y. Zhang, L. Liu, T.J.R. Hughes, Truncated T-splines: Fundamentals and methods, Comput. Methods Appl. Mech. Engrg. 316 (2017) 349–372.

[20] M.A. Scott, M.J. Borden, C.V. Verhoosel, T.W. Sederberg, T.J.R. Hughes, Isogeometric finite element data structures based on bèzier extraction of T-splines, Internat. J. Numer. Methods Engrg. 88 (2011) 126–156.

[21] S. May, J. Vignollet, R. de Borst, The role of the bèzier extraction operator for T-splines of arbitrary degree: linear dependencies, partition of unity property, nesting behaviour and local refinement, Internat. J. Numer. Methods Engrg. 103 (2015) 547–581.

[22] T.D. Hien, H. Noh, Stochastic isogeometric analysis of free vibration of functionally graded plates considering material randomness, Comput. Methods Appl. Mech. Engrg. 318 (2017) 845–863.

[23] K. Li, W. Gao, D. Wu, C. Song, T. Chen, Spectral stochastic isogeometric analysis of linear elasticity, Comput. Methods Appl. Mech. Engrg. 332 (2018) 157–190.

[24] J. Beck, L. Tamellini, R. Tempone, IGA-based multi-index stochastic collocation for random PDEs on arbitrary domains, Comput. Methods Appl. Mech. Engrg. 351 (2019) 330–350.

[25] C. Ding, R.R. Deokar, Y. Ding, G. Li, X. Cui, K.K. Tamma, S.P.A. Bordas, Model order reduction accelerated Monte Carlo stochastic isogeometric method for the analysis of structures with high-dimensional and independent material uncertainties, Comput. Methods Appl. Mech. Engrg. 349 (2019) 266–284.

[26] Z. Liu, M. Yang, J. Cheng, J. Tan, A new stochastic isogeometric analysis method based on reduced basis vectors for engineering structures with random field uncertainties, Appl. Math. Model. 89 (2021) 966–990.

[27] N. Wiener, The homogeneous chaos, Amer. J. Math. 60 (4) (1938) 897–936.

[28] X. Wan, G. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, J. Comput. Phys. 209 (2005) 617–642.

[29] S. Rahman, A spline chaos expansion, SIAM/ASA J. Uncertain. Quantif. 9 (2020) 27–57.

[30] S. Rahman, A polynomial dimensional decomposition for stochastic computing, Internat. J. Numer. Methods Engrg. 76 (2008) 2091–2116.

[31] S. Rahman, R. Jahanbin, A spline dimensional decomposition for uncertainty quantification in high dimensions, SIAM/ASA J. Uncertain. Quantif. (2022) (in Press).

[32] S. Rahman, Approximation errors in truncated dimensional decompositions, Math. Comp. 83 (290) (2014) 2799–2819.

[33] R. Jahanbin, S. Rahman, Stochastic isogeometric analysis in linear elasticity, Comput. Methods Appl. Mech. Engrg. 364 (2020) 112928.

[34] M. Loève, Probability Theory, Vol. II, Berlin, Heidelberg, New York, Springer, 1977.

[35] S. Rahman, A Galerkin isogeometric method for karhunen-loève approximation of random fields, Comput. Methods Appl. Mech. Engrg. 338 (2018) 533–561.

[36] R. Jahanbin, S. Rahman, An isogeometric collocation method for efficient random field discretization, Int. J. Numer. Methods Eng. 117 (3) (2019) 344–369.

[37] R. Jahanbin, S. Rahman, Isogeometric methods for karhunen-loève representation of random fields on arbitrary multi-patch domains, Int. J. Uncertain. Quantif. 11 (3) (2021) 27–57.

[38] M.L. Mika, T.J.R. Hughes, D. Schillinger, P. Wriggers, R.R. Hiemstra, A matrix-free isogeometric Galerkin method for karhunen-loève approximation of random fields using tensor product splines, tensor contraction and interpolation based quadrature, 2021, arXiv:2011. 13861.

[39] C. De Boor, On calculation with B-splines, J. Approx. Theory 6 (1972) 50–62.

[40] L. Chen, R. de Borst, Adaptive refinement of hierarchical T-splines, Comput. Methods Appl. Mech. Engrg. 337 (2018) 220–245.

[41] B. Urick, B. Marussig, E. Cohen, R.H. Crawford, T.J.R. Hughes, R.F. Riesenfeld, Watertight boolean operations: A framework for creating CAD-compatible gap-free editable solid models, Comput. Aided Des. 115 (2019) 147–160.

[42] T.J.R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Dover Publications, 2000.

[43] M.K. Deb, I.M. Babuška, J.T. Oden, Solution of stochastic partial differential equations using Galerkin finite element techniques, Comput. Methods Appl. Mech. Engrg. 190 (2001) 6359–6372.

[44] L.A. Piegl, W. Tiller, The NURBS Book, second ed., Berlin, Springer-Verlag, 1997.

[45] J. Mason, G. Rodriguez, S. Seatzu, Orthogonal splines based on B-splines – with applications to least squares, smoothing and regularisation problems, Numer. Algorithms 5 (1993) 25–40.

[46] S. Dixler, R. Jahanbin, S. Rahman, Uncertainty quantification by optimal spline dimensional decomposition, Internat. J. Numer. Methods Engrg. (2021) http://dx.doi.org/10.1002/nme.6778.

[47] G. Blatman, B. Sudret, Sparse polynomial chaos expansions and adaptive stochastic finite elements using a regression approach, C.R. Mech. 336 (2008) 518–523.

[48] A. Molinaro, R. Simon, R. Pfeiffer, Prediction error estimation: a comparison of resampling methods, Bioinformatics 21 (2005) 3301–3307.

[49] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, J. Comput. Phys. 230 (2011) 2345–2367.

[50] M. Berveiller, B. Sudret, M. Lemaire, Stochastic finite element: a non intrusive approach by regression, Eur. J. Comput. Mech. 15 (2006) 81–92.

[51] MATLAB, Version 2019b, The MathWorks Inc. Natick, MA, 2019.

[52] S. Smolyak, Quadrature and interpolation formulas for tensor products of certain classes of functions, Doklade Akademii Nauk SSSR 148 (1963) 1042–1045.

[53] T. Gerstner, M. Griebel, Numerical integration using sparse grids, Numer. Algorithms 18 (1998) 209–232.

[54] V. Barthelmann, E. Novak, K. Ritter, High dimensional polynomial interpolation on sparse grids, Adv. Comput. Math. 12 (2000) 273–288.

[55] R. Stephens, A. Fatemi, R. Stephens, H. Fuchs, Metal Fatigue in Engineering, second ed., New York, NY, John Wiley & Sons, 2001.