REPORT SERIES

Joakim Beck, Lorenzo Tamellini, Raúl Tempone

IGA-based Multi-Index Stochastic Collocation for random PDEs on arbitrary domains

18-08

IMATI REPORT Series

Nr. 18-08
October 2018

Managing Editor

Michela Spagnuolo

Editorial Office

Istituto di Matematica Applicata e Tecnologie Informatiche "*E. Magenes*"
Consiglio Nazionale delle Ricerche
Via Ferrata, 5/a
27100 PAVIA (Italy)
Email: reports@imati.cnr.it
http://www.imati.cnr.it

Follow this and additional works at: http://www.imati.cnr.it/reports

IGA-based Multi-Index Stochastic Collocation for random PDEs on arbitrary domains

Joakim Beck, Lorenzo Tamellini, Raúl Tempone

Corresponding author:

Lorenzo Tamellini
Istituto di Matematica Applicata e Tecnologie Informatiche "E. Magenes" (IMATI-CNR)
Consiglio Nazionale delle Ricerche
Via Ferrata 5, 27100, Pavia, Italy
E-mail: tamellini@imati.cnr.it

Joakim Beck
King Abdullah University of Science and Technology (KAUST)
Computer, Electrical and Mathematical Science and Engineering Division (CEMSE)
Thuwal 23955-6900, Saudi Arabia
E-mail: joakim.beck@kaust.edu.sa

Raúl Tempone
King Abdullah University of Science and Technology (KAUST)
Computer, Electrical and Mathematical Science and Engineering Division (CEMSE)
Thuwal 23955-6900, Saudi Arabia
E-mail: raul.tempone@kaust.edu.sa

Abstract.

This paper proposes an extension of the Multi-Index Stochastic Collocation (MISC) method for forward uncertainty quantification (UQ) problems in computational domains of shape other than a square or cube, by exploiting isogeometric analysis (IGA) techniques. Introducing IGA solvers to the MISC algorithm is very natural since they are tensor-based PDE solvers, which are precisely what is required by the MISC machinery. Moreover, the combination-technique formulation of MISC allows the straight-forward reuse of existing implementations of IGA solvers. We present numerical results to showcase the effectiveness of the proposed approach.

_____

[page left intentionally blank]

IGA-based Multi-Index Stochastic Collocation for random PDEs on arbitrary domains

Joakim Beck[a], Lorenzo Tamellini[b,*], Raúl Tempone[a]

[a]*King Abdullah University of Science and Technology (KAUST), Computer, Electrical and Mathematical Science and Engineering Division (CEMSE) Thuwal 23955-6900, Saudi Arabia*
[b]*Consiglio Nazionale delle Ricerche - Istituto di Matematica Applicata e Tecnologie Informatiche "E. Magenes" (CNR-IMATI), Via Ferrata 1, 27100, Pavia, Italy*

Abstract

This paper proposes an extension of the Multi-Index Stochastic Collocation (MISC) method for forward uncertainty quantification (UQ) problems in computational domains of shape other than a square or cube, by exploiting isogeometric analysis (IGA) techniques. Introducing IGA solvers to the MISC algorithm is very natural since they are tensor-based PDE solvers, which are precisely what is required by the MISC machinery. Moreover, the combination-technique formulation of MISC allows the straight-forward reuse of existing implementations of IGA solvers. We present numerical results to showcase the effectiveness of the proposed approach.

Highlights

- Isogeometric solvers used in a MISC framework for forward UQ problems.

- The combination-technique formulation of the method allows straight-forward reuse of legacy IGA solvers.

Keywords: Isogeometric analysis, Uncertainty Quantification, Sparse Grids, Stochastic Collocation methods, multilevel methods, combination-technique

1. Introduction

Uncertainty quantification (UQ) has received a considerable amount of attention in recent years and is by now considered an essential tool in the domain of computational science and engineering [1, 2, 3]. However, performing UQ analyses remains a significant computational challenge, since these analyses typically require solving a computational model repeatedly for different values of the uncertain variables in the model. In this paper, we consider in particular models described by elliptic PDEs whose solution is denoted by $\mathbf{u}$. Two general "meta-strategies" (complementary to each other) are by now recognized in the UQ community as key to reducing the computational cost and making UQ analyses feasible: a) dimension-adaptivity, i.e., investing the majority of the computational cost in approximating the dependence of $\mathbf{u}$ on the random variables whose variability has the largest impact on $\mathbf{u}$ itself; and b) a multi-level approach, in which most of the variability of $\mathbf{u}$ is explored by using "low-fidelity" approximations of the PDE (e.g., coarse meshes and simplified-physics models), and resorting to "high-fidelity" approximations only sparingly. Dimension-adaptivity was the first "meta-strategy" to be introduced in the UQ community and has been extensively

---

*Corresponding author
Email addresses: `joakim.beck@kaust.edu.sa` (Joakim Beck), `tamellini@imati.cnr.it` (Lorenzo Tamellini), `raul.tempone@kaust.edu.sa` (Raúl Tempone)

discussed in the literature, e.g. in [4, 5, 6, 7], while the multi-level approach was developed more recently; see, e.g., [8, 9, 10].

The Multi-Index Stochastic Collocation (MISC) method for random PDEs was first introduced in [11, 12] and represents an attempt to combine both strategies. MISC uses an ad-hoc algorithm to simultaneously choose both the best sequence of computational meshes and the random variables whose impact should be more carefully approximated. It is closely related to the sparse-grids technique introduced to solve high-dimensional PDEs [13, 14, 15, 16] and to compute high-dimensional integrals/interpolants [17, 18], and indeed it can be seen as a combination of the two methodologies. The main idea behind the MISC algorithm is to write the approximation operator as a linear combination of many less-refined approximations, in the spirit of a Richardson extrapolation [19]. In particular, a "profit" is assigned to each possible component, and eventually only those components with the largest profit are included in the computation. Another possible approach would be to balance the errors of the physical and stochastic discretizations, as proposed in [20, 21, 22].

Crucially, MISC needs both a physical solver and a sampler in the stochastic domain with a tensor structure. The tensor construction in the stochastic domain can be obtained by e.g. tensorizing standard Lagrangian (interpolant) polynomials, which would also be the starting point of the classical sparse-grids collocation method for UQ [23, 24]. When it comes to the physical solver, previous works on MISC [11, 12] only considered square domains over structured grids, which induce a tensor structure in the solver; possible strategies to apply MISC to non-square domains, on which having a tensorized solver might be non-trivial, were only briefly mentioned in [12].

In this paper, we extend [11, 12] and consider MISC on arbitrarily shaped domains, by employing isogeometric analysis (IGA) [25, 26]. IGA is an alternative to the standard finite element analysis that uses the basis functions employed by CAD softwares to represent the computational domain (typically, B-splines or non uniform rational B-splines (NURBS)) as basis for the approximation of the PDE solution as well. IGA has several interesting features and has therefore received a growing interest from researchers and practitioners in computational science. For example, IGA can work with exact domain representation and the meshing process is simplified in some situations; B-splines/NURBS of arbitrary polynomial degree and regularity can be generated in a very easy way and show, in certain cases, superior error vs. degrees-of-freedom ratio with respect to standard finite element bases. We refer the reader to [27] for an in-depth discussion on IGA. Crucially, multivariate B-splines and NURBS are built by tensorization of their univariate counterparts, which makes IGA solvers particularly suitable for use with MISC.

It is important to remark that in the end the MISC algorithm itself simply prescribes to solve a number of standard uncoupled PDEs, each of them corresponding to a different realization of the random parameters, on physical meshes with different resolutions (possibly anisotropic, i.e., more refined along some physical directions). Therefore, any available IGA software can be readily re-used. We also point out that in the case where there are no random variables, this procedure corresponds to the sparse IGA method for solving PDEs discussed in [28]. Finally, we mention that while IGA is a convenient choice to extend MISC to problems on non-square domains, it is not the only possible choice. Other choices, such as finite differences, finite volumes, $\mathbb{Q}_k$ finite elements, might be envisaged; the comparison of these methods is outside the scope of this work.

The rest of this paper is organized as follows. We introduce the general UQ framework for elliptic PDEs with random coefficients in Section 2, and discuss IGA solvers in Section 3. We present the MISC algorithm in Section 4 and showcase the results obtained with MISC on some numerical examples in Section 5. Finally, we draw conclusions in Section 6.

Throughout the manuscript, we make extensive use of multi-indices, i.e., vectors with integer components. To this end, we recall some useful definitions and notations:

- Given $\mathbf{i}, \mathbf{j} \in \mathbb{N}^K$, $\mathbf{i} \le \mathbf{j}$ means that $i_k \le j_k$ for $k = 1, \ldots, K$;

- $\boldsymbol{e}_i$ is the i-th canonical multi-index, i.e., $(\boldsymbol{e}_i)_k = 1$ if $i = k$, and zero otherwise;

- **1** is a multi-index whose components are all equal to 1;

- Given a function $f : \mathbb{R} \to \mathbb{R}$ and a multi-index $\mathbf{i}$, $\mathbf{f(i)}$ denotes the multi-index $[f(i_1), f(i_2), \ldots, f(i_K)]$;

- A multi-index set $\Lambda \subset \mathbb{N}^K$ is said to be downward closed if

$$\forall \mathbf{i} \in \Lambda, \quad \mathbf{i} - \mathbf{e}_j \in \Lambda \text{ for every } j = 1, \ldots, K \text{ such that } \mathbf{i}_j > 1. \tag{1}$$

- The margin of a multi-index set Λ, $\mathrm{Mar}(\Lambda)$, is the set of multi-indices that can be reached "within one step" from Λ,

$$\mathrm{Mar}(\Lambda) = \{\mathbf{i} \in \mathbb{N}^K \text{ s.t. } \mathbf{i} = \mathbf{j} + \mathbf{e}_k \text{ for some } \mathbf{j} \in \Lambda \text{ and some } k \in \{1, \ldots, K\}\}; \tag{2}$$

- The reduced margin of a multi-index set Λ, $\mathrm{Red}(\Lambda)$, is the subset of $\mathrm{Mar}(\Lambda)$ composed by indices from which "every backward step" will take inside Λ,

$$\mathrm{Red}(\Lambda) = \{\mathbf{i} \in \mathbb{N}^K \text{ s.t. } \mathbf{i} - \mathbf{e}_k \in \Lambda \text{ for every } k \in \{1, \ldots, K\} \text{s.t. } i_k > 1\}. \tag{3}$$

2. Problem definition

Let $\mathcal{B}$ be a compact domain in $\mathbb{R}^d$, $d = 2, 3$, that represents the "physical domain" of the problem. In addition, let $\mathbf{y} = [y_1, y_2, \ldots, y_N]$ be an N-dimensional random vector, whose components are mutually independent random variables with support $\Gamma_n \subset \mathbb{R}$ and probability density function $\rho_n(y_n)$. Thus, $\mathbf{y} \in \Gamma = \Gamma_1 \times \Gamma_2 \cdots \Gamma_N$, and Γ represents the "stochastic domain" of the problem; since y_n are mutually independent, $\rho(\mathbf{y}) = \prod_{n=1}^N \rho_n(y_n)$ is a probability density function on Γ. Throughout this work, we often refer to y_n as "stochastic directions", which is a short-hand for "directions of the stochastic domain".

We consider the following problem: Find $\mathbf{u} : \mathcal{B} \times \Gamma \to \mathbb{R}^m$ such that for ρ-almost every $\mathbf{y} \in \Gamma$,

$$\begin{cases} \mathcal{L}(\mathbf{u}; \mathbf{x}, \mathbf{y}) = \mathcal{F}(\mathbf{x}) & \mathbf{x} \in \mathcal{B}, \\ \mathbf{u}(\mathbf{x}, \mathbf{y}) = 0 & \mathbf{x} \in \partial\mathcal{B}, \end{cases} \tag{4}$$

where $\mathcal{L}$ is a differential operator and $\mathcal{F}$ is an operator on $\mathbf{x}$. In particular, in the numerical results section we consider a linear scalar elliptic equation and a linear elasticity equation. We assume well-posedness of the problem in some Hilbert space V for ρ-almost every $\mathbf{y} \in \Gamma$ (specific choices of V are detailed for each example in Section 5); observe that $\mathbf{u}$ can also be seen as an N-variate Hilbert-space-valued function $\mathbf{u}(\mathbf{y}) : \Gamma \to V$, and in particular it is convenient to introduce the Bochner space of finite-variance Hilbert-space-valued functions, $L_\rho^2(\Gamma; V) = \{\mathbf{u} : \Gamma \to V \text{ strongly measurable such that } \int_\Gamma \|\mathbf{u}(, \cdot, \mathbf{y})\|_V^2 \rho(\mathbf{y}) d\mathbf{y} < \infty\}$, to which $\mathbf{u}$ is assumed to belong.

The random variables $\mathbf{y}$ model the uncertainties in the system, i.e., account for the fact that coefficients, forcing terms, boundary/initial conditions, and domain shape are often "imperfectly" known due to measurement errors, lack of data, or intrinsic variability (e.g., when they describe phenomena like wind, rain, earthquakes). The goal of a forward UQ analysis is therefore to assess how much the variability of such random objects affects the quantities of interest of the computation, which could be either the solution $\mathbf{u}$ or a functional thereof. In particular, in this work we assume that some functional of the solution $\mathbf{u}$, $\Phi : V \to \mathbb{R}$, e.g. $\Phi(\mathbf{v}) = \int_\mathcal{B} \mathbf{v}(\mathbf{x}) d\mathbf{x}$ or $\Phi(\mathbf{v}) = \mathbf{v}(\mathbf{x}_0)$ is given, and we aim to estimate its expected value, i.e., we want to compute

$$\mathbb{E}[\Phi(\mathbf{u}(\mathbf{x}, \mathbf{y}))] = \int_\Gamma \Phi(\mathbf{u}(\mathbf{x}, \mathbf{y})) \rho(\mathbf{y}) d\mathbf{y}.$$

In this work, we numerically analyze the performance of MISC for this task. More specifically, we use MISC to select the physical and stochastic discretizations parameters, but other discretization parameters (e.g., time-steps, number of particles, solver tolerances) could be added to the set of parameters governed by MISC. The novelty of the present work consists in showing how to extend the MISC methodology to more general physical domain shapes, by replacing the multi-linear finite elements solver adopted in previous works with the IGA solvers that we present in the next section.

3. Isogeometric solvers

In this section, we briefly present the fundamentals of isogeometric analysis (IGA) and refer the reader to [25, 26, 27] for a more thorough discussion. The first ingredient of IGA is a B-splines/NURBS representation of the computational domain $\mathcal{B}$, which is usually provided by a CAD software. An isogeometric solver typically then uses the same set of basis functions to compute an approximation of the solution of the PDE. However, that this is not strictly needed, and two different sets of B-splines/NURBS functions could be used instead: one to approximate the geometry and another to approximate the solution.

The B-splines/NURBS representation of $\mathcal{B} \subset \mathbb{R}^d$ consists of a transformation from a reference domain $\widehat{\mathcal{B}}$ (typically, a square or cube) to the physical domain $\mathcal{B}$, written as a linear combination of B-splines/NURBS functions and so-called control points $\mathbf{P_i} \in \mathbb{R}^d$. Following the IGA/CAD literature, we refer in this work to $\widehat{\mathcal{B}}$ as the "parametric domain".

The B-splines/NURBS used in such representation are built by tensorization, and therefore we begin our presentation by considering the univariate case. We introduce a reference interval, $\hat{I} = [0,1]$, and a knot vector over $\hat{I}$, i.e, a non-decreasing vector $\Xi = [\xi_1, \xi_2 ..., \xi_{n+p+1}]$, where $n, p \in \mathbb{N}$ and ξ_1, ξ_{n+p+1} coincide with the extrema of $\hat{I}$, that will be used to define a set of n B-splines polynomials of degree p over $\hat{I}$; observe that Ξ can have repeated entries, for reasons that will be clear later on (see Figure 1). Each ξ_i is called "knot", and an interval (ξ_i, ξ_{i+1}) of non-zero length is an "element"; N_{el} is the number of elements. The elements are not required to have the same length, but if they do, we call that length the mesh-size, which is denoted by h. We define the non-decreasing vector $Z = [\zeta_1, \ldots, \zeta_{N_{el}+1}]$ as the vector of knots of Ξ without repetitions, and m_i is the multiplicity of ζ_i in Ξ, such that $\sum_{i=1}^{N_{el}} m_i = n + p + 1$. A knot vector is said to be "open" if its first and last knots have a multiplicity of $p + 1$.

We can now define the B-splines polynomials of degree p on $\hat{I}$ by means of the Cox-De Boor recursive formula [27]: we start with piecewise constant ($\tilde{p} = 0$),

$$\widehat{S}_{i,0}(\xi) = \begin{cases} 1 & \xi_i \leq \xi < \xi_{i+1} \\ 0 & \text{otherwise,} \end{cases} \qquad \text{for } i = 1, \ldots, n+p$$

and then, for $\tilde{p} = 1, \ldots, p$, we have the recursive step

$$\widehat{S}_{i,\tilde{p}}(\xi) = \begin{cases} \dfrac{\xi - \xi_i}{\xi_{i+\tilde{p}} - \xi_i} \widehat{S}_{i,\tilde{p}-1}(\xi) + \dfrac{\xi_{i+\tilde{p}+1} - \xi}{\xi_{i+\tilde{p}+1} - \xi_{i+1}} \widehat{S}_{i+1,\tilde{p}-1}(\xi), & \xi_i \leq \xi < \xi_{i+\tilde{p}+1} \\ 0, & \text{otherwise} \end{cases} \qquad \text{for } i = 1, \ldots, n+p-\tilde{p},$$

with the understanding that $0/0 = 0$; note that if the knot vector Ξ is open, the corresponding basis will be interpolatory in the first and last knots. The B-splines are polynomials of degree p and continuity C^{p-m_i} at ζ_i, which means that the regularity of the B-splines can be reduced by repeating multiple times the same entry of the knot vector. In particular, repeating a knot p times will result in a basis with C^0 regularity in that knot, which means that the basis will also be interpolatory at that knot; see Figure 1-left for an example. The generated B-splines are linearly independent, and we refer to their span as the "space of splines", i.e., $W_p(\Xi, \hat{I})$:

$$W_p(\Xi, \hat{I}) = \text{span}\left\{\widehat{S}_{i,p}, \ i = 1, \ldots, n\right\}.$$

Often, all the internal knots are repeated the same number of times to obtain a B-splines basis with continuity r at each point ζ_i, $0 \leq r \leq p - 1$. If that is the case, we add a superscript r to the notation, i.e., $S_{i,p}^r$ or $W_p^r(\Xi, \hat{I})$. Clearly, for a fixed polynomial degree p, the number of basis functions n decreases as r increases; see Figure 1-center and right for examples of C_p^{p-1} and C_p^0 B-splines.

Next, we define B-splines on d-dimensional domains for the case $d = 2$, with the understanding that the extension to the case $d = 3$ is trivial. The fact that multivariate B-splines are defined by tensorization is crucial for the development of the MISC methodology, as will become clear in Section 4. We define the parametric domain $\widehat{\mathcal{B}} = \hat{I} \times \hat{I}$ and consider two open knot vectors Ξ_1, Ξ_2 with $n_1 + p_1 + 1$ and $n_2 + p_2 + 1$ knots, respectively; the corresponding knots without repetitions are denoted by Z_1, Z_2. We introduce the tensor
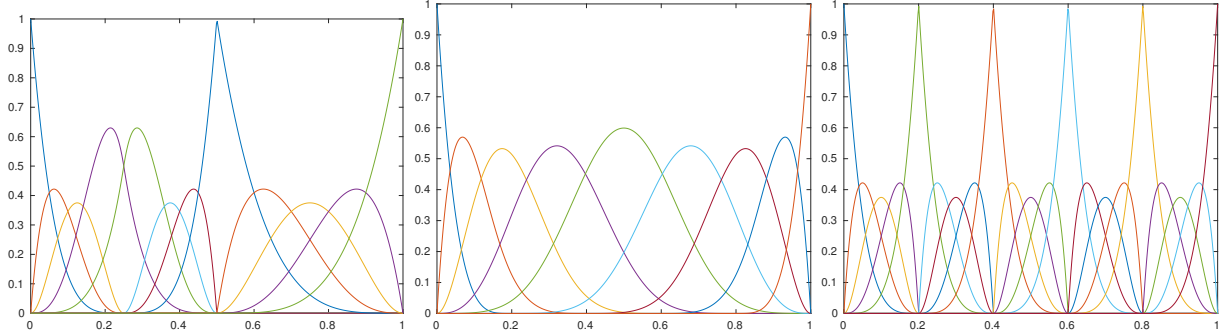
4

Figure 1: B-splines basis built on different knot vectors; each knot vector is "open", i.e., the first and last nodes are repeated $p + 1$ times, with $p = 4$. Left: $\Xi = [0, 0, 0, 0, 0, 0.25, 0.25, 0.25, 0.5, 0.5, 0.5, 0.5, 1, 1, 1, 1, 1]$. Note the different continuity at $\xi = 0.25$ and $\xi = 0.5$ due to different multiplicity: ξ is repeated 3 times, thus the B-splines are C^1 in that knot; ξ is repeated 4 times, thus the B-splines are C^0 in that knot. Center: $\Xi = [0, 0, 0, 0, 0, 0.2, 0.4, 0.6, 0.8, 1, 1, 1, 1, 1]$. In this case, the internal knots are repeated only once, resulting in maximally smooth B-splines, i.e., C^3 at the internal knots. Right: $\Xi = [0, 0, 0, 0, 0, 0.2, 0.2, 0.2, 0.2, 0.4, 0.4, 0.4, 0.4, 0.6, 0.6, 0.6, 0.6, 0.8, 0.8, 0.8, 0.8, 1, 1, 1, 1, 1]$. Internal knots are repeated p times, resulting in C^0 splines (i.e., Lagrangian finite elements of degree p). As expected, the higher the continuity of the basis functions, the smaller the cardinality of the basis.

products $\boldsymbol{\Xi} = \Xi_1 \otimes \Xi_2$ and $\mathbf{Z} = Z_1 \otimes Z_2$; $\mathbf{Z}$ generates a Cartesian mesh over $\widehat{\mathcal{B}}$ composed by $N_{el,1} \times N_{el,2}$ rectangular elements. According to the CAD/IGA literature we refer throughout this work to the two directions ξ_1 and ξ_2 as "parametric directions". A basis for the space of bi-variate splines is then obtained by taking tensor products of the univariate B-splines,

$$W_{\mathbf{p}}^{\mathbf{r}}(\boldsymbol{\Xi}, \widehat{\mathcal{B}}) = \mathrm{span}\{\widehat{S}_{\mathbf{i}, \mathbf{p}}, \mathbf{i} \leq \mathbf{n}\},$$

where $\mathbf{i} = [i_1, i_2]$, $\mathbf{p} = [p_1, p_2]$, $\mathbf{n} = [n_1, n_2]$, $\mathbf{r} = [r_1, r_2]$, and $\widehat{S}_{\mathbf{i}, \mathbf{p}}(\xi_1, \xi_2) = \widehat{S}_{i_1, p_1}(\xi_1) \widehat{S}_{i_2, p_2}(\xi_2)$.

We are now in the position to introduce the B-splines representation of the computational domain $\mathcal{B}$ using a linear combination of B-splines with control points $\mathbf{P}_{\mathbf{i}} \in \mathbb{R}^2$ and $\mathbf{i} \leq \mathbf{n}$ (see also Figure 2):

$$\mathbf{x} \in \mathcal{B} \Leftrightarrow \mathbf{x} = \mathbf{G}(\boldsymbol{\xi}) = \sum_{\mathbf{i} \leq \mathbf{n}} \mathbf{P}_{\mathbf{i}} \widehat{S}_{\mathbf{i}, \mathbf{p}}(\boldsymbol{\xi}) \text{ for some } \boldsymbol{\xi} \in \widehat{\mathcal{B}}.$$

In the CAD literature, the function $\mathbf{G} : \widehat{\Omega} \to \Omega$ is often called a "parameterization" of the geometry $\mathcal{B}$. Observe that the control points need not belong to $\mathcal{B}$: this is the case only if the corresponding basis function is C^0 continuous; see again Figure 2. With a slight abuse of wording, in the following we sometimes talk of "physical directions" instead of "parametric directions"; we also use "physical directions" as a shorthand for the longer "curvilinear coordinates induced by mapping the parametric directions over the physical space". However, this talk of "physical directions" will be useful when juxtaposed with the "stochastic directions" $y_1, \ldots, y_N$.

Non-planar surfaces can be also generated in the same way, by choosing $\mathbf{P}_{\mathbf{i}} \in \mathbb{R}^3$ instead of $\mathbf{P}_{\mathbf{i}} \in \mathbb{R}^2$; see Figure 3-left. A further round of tensorizations, again with $\mathbf{P}_{\mathbf{i}} \in \mathbb{R}^3$, would allow us to represent volumetric computational domains $\mathcal{B}$; see Figure 3-right. Incidentally, note that many geometries of practical interest, and in particular all conic sections other than parabulae, cannot be represented exactly by B-splines. To this end, the so-called *non-uniform rational B-splines* (NURBS) have been introduced, see again [25, 26, 27] for details. As the name suggests, NURBS are ratios of B-splines and retain most of the properties of B-splines.

In IGA, the B-splines basis is also used to approximate the solution to (4). We therefore introduce the B-splines space on the physical domain $\mathcal{B}$ as

$$W_{\mathbf{p}}(\boldsymbol{\Xi}, \mathcal{B}) = \mathrm{span}\{S_{\mathbf{i}, \mathbf{p}} = \widehat{S}_{\mathbf{i}, \mathbf{p}} \circ \mathbf{G}^{-1}, \mathbf{i} \leq \mathbf{n}\},$$

and then approximate the solution to (4) for a fixed value of the random variables $\mathbf{y}$ as

$$\mathbf{u}(\mathbf{x}, \mathbf{y}) \approx \mathbf{u}_{\mathbf{n}}(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{i} \leq \mathbf{n}} c_{\mathbf{i}} S_{\mathbf{i}, \mathbf{p}}(\mathbf{x}), \tag{5}$$
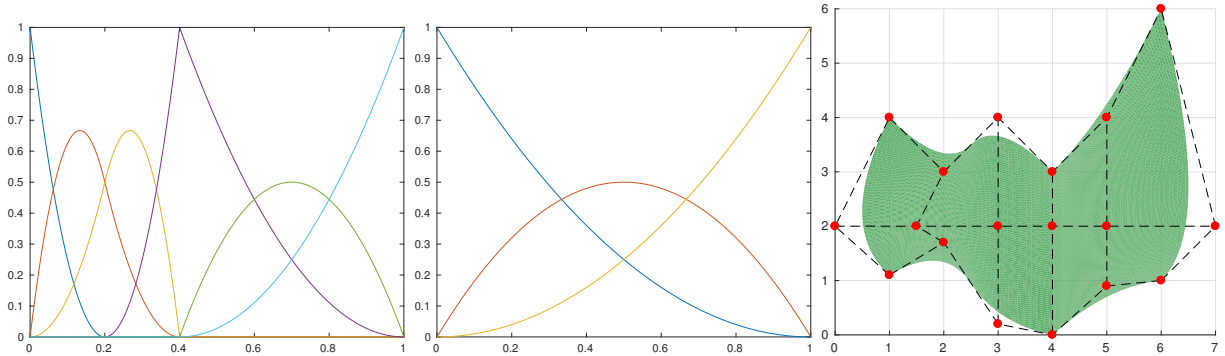
Figure 2: Surfaces are built by linear combinations of control points and tensorized B-splines. Left: B-spline basis along the first reference interval; center: B-spline basis along the second reference interval. The tensor product of these two bases forms the basis (with cardinality 18) of the linear combination used to represent the surface in the right plot. The corresponding 18 control points are marked in red. The dotted lines connecting the control points represent the so-called "control net".
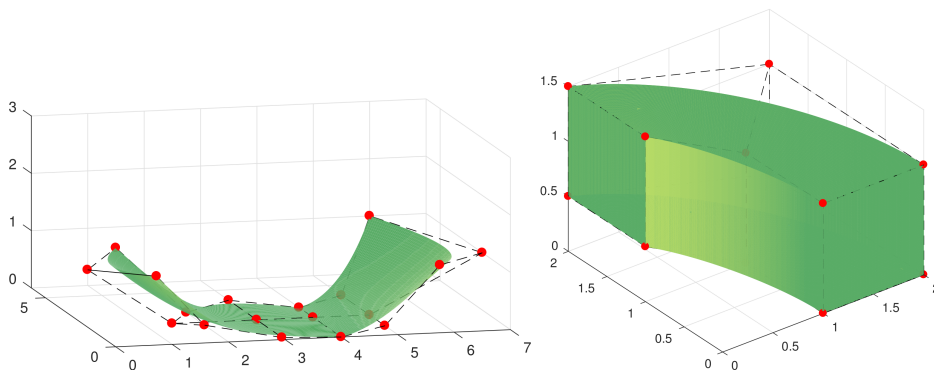

Figure 3: Left: A non-planar surface, obtained from the example in Figure 2 by choosing control points in $\mathbb{R}^3$ instead of $\mathbb{R}^2$. Right: A volume is represented by taking tensor products of three B-splines bases.

where the coefficients $c_{\mathbf{i}}$ depend on $\mathbf{y}$ (for simplicity, we do not show this dependence in the equation). The coefficients $c_{\mathbf{i}}$ can be computed either by following a standard Galerkin approach or by collocation methods, i.e., inserting (5) in the strong form of the problem, (4), and enforcing that the equation be satisfied in a set of collocation points, which results in a system of linear equations to be solved [29, 30, 31, 32]. In this work, we employ a standard Galerkin method; we remark again that, as will be clearer later, pre-existing solvers can be readily reused in the context of the MISC method.

As expected, the approximation of $\mathbf{u}$ in (5), $u_{\mathbf{n}}$, converges to $\mathbf{u}$ as the cardinality of $W_{\mathbf{p}}(\Xi, \mathcal{B})$ increases, and the convergence results with respect to both h and p are analogous to the standard finite elements results.

In the following, we fix the polynomial degrees along each parametric direction, i.e., $p_1 = \ldots = p_d = p$, and consider h-refinements only, i.e., we increase the cardinality of the basis by adding knots to the initial knot vectors Ξ_1 and Ξ_2, though it is also possible to devise a p-MISC technique (or a combined $h - p$ version). In particular, for each parametric direction $\xi_1, i = 1, \ldots, d$, we consider a sequence of knot vectors Ξ_i indexed by an integer $\alpha_i \geq 1$, whose number of elements doubles whenever α_i increases by 1: $N_{el,i} \sim 2^{\alpha_i}$. We let $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_d] \in \mathbb{N}^d$, and denote the associated solution of the PDE by $\mathbf{u}_{\boldsymbol{\alpha}}$.

4. Multi-Index Stochastic Collocation (MISC)

Besides IGA, the other basic building block of MISC is a tensorized quadrature formula over the stochastic domain Γ that can be used to evaluate, e.g., expected value, variance and higher-order moments of multi-

6

variate random functions defined over Γ.

Let us therefore start by introducing a quadrature operator for a univariate real-valued continuous function $v(t) : \widetilde{\Gamma} \to \mathbb{R}$, with the understanding that $\widetilde{\Gamma}$ is a placeholder for any of the univariate sub-domains $\Gamma_1, \ldots, \Gamma_N$ composing the random space $\Gamma = \Gamma_1 \times \ldots \times \Gamma_N$; therefore, it comes with an associated probability density function, that we denote by $\widetilde{\rho}$. The quadrature operator is defined as

$$\mathcal{Q}^{m(\beta)} : C^0(\widetilde{\Gamma}) \to \mathbb{R}, \quad \mathcal{Q}^{m(\beta)}[v] = \sum_{j=1}^{m(\beta)} v(t_{\beta,j})\omega_{\beta,j},$$

where $\beta \geq 1$ is a positive integer (usually referred to as the "refinement level" or just "level" of the quadrature operator), $m(\beta)$ a strictly increasing function giving the number of distinct quadrature points to be used, $\{t_{\beta,j}\}_{j=1}^{m(\beta)} \subset \widetilde{\Gamma}$, with corresponding weights $\{\omega_{\beta,j}\}_{j=1}^{m(\beta)}$. The quadrature points should be chosen according to the underlying probability measure $\widetilde{\rho}$; see, e.g., [5, 33]. Moreover, for refinement purposes it is advantageous if the quadrature points are chosen to be "nested", i.e., $\{t_{\beta,j}\}_{j=1}^{m(\beta)} \subset \{t_{\beta+1,j}\}_{j=1}^{m(\beta+1)}, \forall \beta \geq 1$. In the numerical examples of this work we consider problems depending on uniform random variables, for which a number of different families of nested quadrature points exist: Clenshaw–Curtis points, several variants of Leja points, and Gauss–Patterson points; see, e.g., [5, 34, 35, 36, 37]. In particular, we adopt the Clenshaw-Curtis points, which are defined as

$$t_{\beta,j} = \cos\left(\frac{(j-1)\pi}{m(\beta)-1}\right), \quad 1 \leq j \leq m(\beta),$$

and are nested provided that the function $m(\beta)$ is chosen as

$$m(0) = 0, \ m(1) = 1, \ m(\beta) = 2^{\beta-1} + 1, \ \beta \geq 2.$$

The extension to multi-variate real-valued continuous functions $v(\boldsymbol{t}) : \Gamma \to \mathbb{R}$ is obtained by tensorization of the univariate quadrature operators. In detail, we introduce a multi-index $\boldsymbol{\beta} \in \mathbb{N}^N$, whose i-th component gives the level of the univariate quadrature to be used along Γ_i, and we define the multi-variate quadrature operator as

$$\mathcal{Q}^{m(\boldsymbol{\beta})} : C^0(\Gamma) \to \mathbb{R}, \quad \mathcal{Q}^{m(\boldsymbol{\beta})} = \bigotimes_{1 \leq i \leq N} \mathcal{Q}^{m(\beta_i)}, \quad \mathcal{Q}^{m(\boldsymbol{\beta})}[v] = \sum_{j=1}^{\#\boldsymbol{m}(\boldsymbol{\beta})} v(\boldsymbol{t}_j)\omega_j,$$

where $\boldsymbol{t}_j$ are points in the Cartesian grid $\bigotimes_{1 \leq i \leq N} \{z_{\beta_i,j}\}_{j=1}^{m(\beta_i)}$, ω_j are the corresponding products of weights of the one-dimensional quadrature rules, and $\#\boldsymbol{m}(\boldsymbol{\beta})$ denotes the total number of quadrature points in the Cartesian grid, $\#\boldsymbol{m}(\boldsymbol{\beta}) = \prod_{i=1}^N m(\beta_n)$.

We are now in the position to introduce the MISC approximation of the expected value of a functional of the solution of the PDE, $\mathbb{E}[\Phi(\mathbf{u}(\mathbf{x}, \mathbf{y}))]$. For ease of presentation, we introduce the function $\phi : \Gamma \to \mathbb{R}$ which associates each $\mathbf{y}$ with its corresponding value of the functional $\phi(\mathbf{y}) = \Phi(\mathbf{u}(\mathbf{x}, \mathbf{y}))$; thus, our goal becomes computing an approximation of $\mathbb{E}[\phi(\mathbf{y})]$.

Clearly, the value of $\phi(\mathbf{y})$ is accessible to us only by solving the PDE after fixing the value of the random vector $\mathbf{y}$: in particular, we denote by $\phi_{\boldsymbol{\alpha}}$ the value of ϕ obtained by post-process of $\mathbf{u}_{\boldsymbol{\alpha}}$, where (as already mentioned) $\boldsymbol{\alpha}$ prescribes the number of elements in the knot vectors used to build the IGA solver, $N_{el,i} \sim 2^{\alpha_i}$.

The fully discrete approximation of $\mathbb{E}[\phi(\mathbf{y})]$ is therefore completely determined by the choice of the discretizations $\boldsymbol{\alpha}$ in the physical space and by $\boldsymbol{\beta}$ in the probability space, i.e., $\mathbb{E}[\phi(\mathbf{y})] \approx M_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \mathcal{Q}^{\boldsymbol{m}(\boldsymbol{\beta})}[\phi_{\boldsymbol{\alpha}}]$. Of course, the ideal approximation $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ could be obtained by setting $\alpha_1 = \ldots = \alpha_d = \bar{\alpha} \gg 1$ and $\beta_1 = \ldots = \beta_N = \bar{\beta} \gg 1$, but this is out of reach for even moderate values of $d, N, \bar{\alpha}$, and $\bar{\beta}$, due to its combinatorial computational cost. Instead, we resort in MISC to the classical "sparsification" construction, which was already introduced in the literature for solving high-dimensional PDEs [14] and quadrature problems [17].

To this end, we need to introduce the so-called univariate and multivariate "detail operators" on the physical and stochastic domains as follows, with the understanding that $M_{\boldsymbol{\alpha},\boldsymbol{\beta}} = 0$ when at least one component of $\boldsymbol{\alpha}$ or $\boldsymbol{\beta}$ is zero:

Univariate physical detail: $\quad \Delta_i^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = M_{\boldsymbol{\alpha},\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\boldsymbol{e}_i,\boldsymbol{\beta}}$ with $1 \le i \le d$;

Univariate stochastic detail: $\quad \Delta_i^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = M_{\boldsymbol{\alpha},\boldsymbol{\beta}} - M_{\boldsymbol{\alpha},\boldsymbol{\beta}-\boldsymbol{e}_i}$ with $1 \le i \le N$;

Multivariate physical detail: $\quad \boldsymbol{\Delta}^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \bigotimes\limits_{i=1}^{d} \Delta_i^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]$;

Multivariate stochastic detail: $\boldsymbol{\Delta}^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \bigotimes\limits_{j=1}^{N} \Delta_j^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]$;

Mixed multivariate detail: $\quad \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \boldsymbol{\Delta}^{\mathrm{stoc}}\left[\boldsymbol{\Delta}^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]\right].$

Observe that taking tensor products of univariate details amounts to composing their actions, e.g.,

$$\boldsymbol{\Delta}^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \bigotimes\limits_{i=1}^{d} \Delta_i^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \Delta_1^{\mathrm{phys}}\left[\Delta_2^{\mathrm{phys}}\left[\cdots \Delta_d^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]\right]\right],$$

and analogously for the stochastic multivariate detail operators, $\boldsymbol{\Delta}^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]$. Crucially, this in turn implies that the multivariate operators can be evaluated by evaluating certain full-tensor approximations $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ and then taking linear combinations:

$$\boldsymbol{\Delta}^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \Delta_1^{\mathrm{phys}}\left[\Delta_2^{\mathrm{phys}}\left[\cdots \Delta_D^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]\right]\right] = \sum\limits_{\mathbf{j} \in \{0,1\}^d} (-1)^{|\mathbf{j}|} M_{\boldsymbol{\alpha}-\mathbf{j},\boldsymbol{\beta}};$$

$$\boldsymbol{\Delta}^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \sum\limits_{\mathbf{j} \in \{0,1\}^N} (-1)^{|\mathbf{j}|} M_{\boldsymbol{\alpha},\boldsymbol{\beta}-\mathbf{j}}.$$

The latter expression is known in the sparse-grids community as "combination-technique", and can be very useful for practical implementations, especially for evaluating $\boldsymbol{\Delta}^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]$: indeed, it allows to evaluate detail operators by calling pre-existing softwares on different meshes (in this case, IGA solvers) up to 2^d times in a "black-box" fashion. As an example, if $d = 2$, then

$$\begin{aligned}
\boldsymbol{\Delta}^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] &= \Delta_2^{\mathrm{phys}}\left[\Delta_1^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]\right] \\
&= \Delta_2^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}}] \\
&= M_{\boldsymbol{\alpha},\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_2,\boldsymbol{\beta}} + M_{\boldsymbol{\alpha}-\mathbf{1},\boldsymbol{\beta}}.
\end{aligned}$$

We remark that the four meshes needed to evaluate the combination-technique expression above are possibly anisotropic, i.e., they may have different levels of discretization along the different physical directions. A similar expression holds for the stochastic details, as well as for the mixed details. Specifically, evaluating $\boldsymbol{\Delta}^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]$ requires evaluating up to 2^N operators $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ over different quadrature grids, and evaluating $\boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]$ requires requires evaluating up to 2^{d+N} operators $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ over different quadrature grids and physical meshes. For instance, if $d = N = 2$, then

$$\boldsymbol{\Delta}^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = M_{\boldsymbol{\alpha},\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_2,\boldsymbol{\beta}} + M_{\boldsymbol{\alpha}-\mathbf{1},\boldsymbol{\beta}};$$

$$\begin{aligned}
\boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] &= \boldsymbol{\Delta}^{\mathrm{stoc}}[\boldsymbol{\Delta}^{\mathrm{phys}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}]] \\
&= \boldsymbol{\Delta}^{\mathrm{stoc}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_2,\boldsymbol{\beta}} + M_{\boldsymbol{\alpha}-\mathbf{1},\boldsymbol{\beta}}] \\
&= M_{\boldsymbol{\alpha},\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}} - M_{\boldsymbol{\alpha}-\mathbf{e}_2,\boldsymbol{\beta}} + M_{\boldsymbol{\alpha}-\mathbf{1},\boldsymbol{\beta}} \\
&\quad - (M_{\boldsymbol{\alpha},\boldsymbol{\beta}-\mathbf{e}_1} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}-\mathbf{e}_1} - M_{\boldsymbol{\alpha}-\mathbf{e}_2,\boldsymbol{\beta}-\mathbf{e}_1} + M_{\boldsymbol{\alpha}-\mathbf{1},\boldsymbol{\beta}-\mathbf{e}_1}) \\
&\quad - (M_{\boldsymbol{\alpha},\boldsymbol{\beta}-\mathbf{e}_2} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}-\mathbf{e}_2} - M_{\boldsymbol{\alpha}-\mathbf{e}_2,\boldsymbol{\beta}-\mathbf{e}_2} + M_{\boldsymbol{\alpha}-\mathbf{1},\boldsymbol{\beta}-\mathbf{e}_2}) \\
&\quad + M_{\boldsymbol{\alpha},\boldsymbol{\beta}-\mathbf{1}} - M_{\boldsymbol{\alpha}-\mathbf{e}_1,\boldsymbol{\beta}-\mathbf{1}} - M_{\boldsymbol{\alpha}-\mathbf{e}_2,\boldsymbol{\beta}-\mathbf{1}} + M_{\boldsymbol{\alpha}-\mathbf{1},\boldsymbol{\beta}-\mathbf{1}}.
\end{aligned}$$

Observe that by introducing these operators we have access to a hierarchical decomposition of $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$; indeed, the following is a telescopic identity

$$M_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \sum_{[\mathbf{i},\mathbf{j}]\leq[\boldsymbol{\alpha},\boldsymbol{\beta}]} \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\mathbf{i},\mathbf{j}}],$$

i.e., it can be easily verified by replacing each term $\boldsymbol{\Delta}^{\mathrm{mix}}[M_{\mathbf{i},\mathbf{j}}]$ with the the corresponding combination-technique formula that all terms except $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ will cancel. For instance, recalling that by definition $M_{i,j} = 0$ when either $i = 0$ or $j = 0$, if $d = N = 1$ we have

$$\sum_{[i,i]\leq[2,2]} \boldsymbol{\Delta}^{\mathrm{mix}}[M_{i,j}] = \boldsymbol{\Delta}^{\mathrm{mix}}[M_{1,1}] + \boldsymbol{\Delta}^{\mathrm{mix}}[M_{1,2}] + \boldsymbol{\Delta}^{\mathrm{mix}}[M_{2,1}] + \boldsymbol{\Delta}^{\mathrm{mix}}[M_{2,2}] \tag{6}$$

$$= M_{1,1} + (M_{1,2} - M_{1,1}) + (M_{2,1} - M_{1,1}) + M_{2,2} - M_{2,1} - M_{1,2} - M_{1,1}$$

$$= M_{2,2}.$$

The crucial observation is that (under suitable regularity assumptions on $\mathbf{u}$), not all of the details in the above hierarchical decomposition contribute equally to the approximation, i.e., they can be discarded and the resulting formula will retain good approximation properties at a fraction of the computational cost. Thus, we introduce the MISC approximation of $\mathbb{E}[\phi]$ as

$$\mathcal{I}_{\Lambda}^{\mathrm{MISC}} = \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}]\in\Lambda} \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}],$$

for a suitable multi-index set $\Lambda \subset \mathbb{N}^{d+N}$, which should be chosen as downward closed, see (1). Clearly, the MISC estimator has a combination-technique expression as well, which can be written in compact form as

$$\mathcal{I}_{\Lambda}^{\mathrm{MISC}} = \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}]\in\Lambda} \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] = \sum_{\substack{[\boldsymbol{\alpha},\boldsymbol{\beta}]\in\Lambda \\ }} \sum_{\substack{[\mathbf{i},\mathbf{j}]\in\{0,1\}^{d+N} \\ [\boldsymbol{\alpha}+\mathbf{i},\boldsymbol{\beta}+\mathbf{j}]\in\Lambda}} (-1)^{|[\mathbf{i},\mathbf{j}]|} M_{\boldsymbol{\alpha},\boldsymbol{\beta}}, \tag{7}$$

which means that again we can evaluate MISC by evaluating full-tensor operators $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ independently and combining them linearly according to (7).

Of course, the effectiveness of the MISC estimator depends on the choice of the multi-index set Λ; the general principle underlying the "sparse" construction is that Λ should be chosen to exclude isotropic full-tensor operators from the estimate, i.e., those operators that simultaneously refine both the mesh on the physical domain and the quadrature grid on the stochastic domain. Rather, indices that refine only a subset of the physical and/or stochastic directions should be chosen, and then the combination-technique formula should be used to combine the partial results; this general idea is exemplified in Figure 4. In a sense, the sparsification approach can be considered a Richardson extrapolation method, in the sense that an ensemble of coarse approximations are combined in such a way that the result is more accurate than each of the components alone. A simple yet quite effective choice of Λ that keeps to a minimum the number of isotropic full-tensor operators to be computed is the so-called "total-degree" set [14, 38]

$$\Lambda_{TD}(\boldsymbol{\kappa},\boldsymbol{g},w) = \left\{ [\boldsymbol{\alpha},\boldsymbol{\beta}] \in \mathbb{N}^{d+N} : \sum_{i=1}^{d} \kappa_i\alpha_i + \sum_{i=1}^{N} g_i\beta_i \leq w \right\}, \text{ for some } w \in \mathbb{N}, \tag{8}$$

where κ_i, g_i are positive real values that can be used to allow more refinement along selected physical or stochastic directions (the smaller the coefficient, the larger the maximum refinement level allowed along that specific direction). For instance, if $d = N = 1$, $w = 2$, and the weights are chosen as $\kappa_i, g_i = 1$, we have the set $\Lambda_{TD}(1,1,2) = \{[i,j] \in \mathbb{N}^2 : i + j \leq 2\}$, which amounts to

$$\sum_{[i,j]\in\Lambda_{TD}(1,1,2)} \boldsymbol{\Delta}^{\mathrm{mix}}[M_{i,j}] = \boldsymbol{\Delta}^{\mathrm{mix}}[M_{1,1}] + \boldsymbol{\Delta}^{\mathrm{mix}}[M_{1,2}] + \boldsymbol{\Delta}^{\mathrm{mix}}[M_{2,1}]$$

$$= M_{1,1} + (M_{1,2} - M_{1,1}) + (M_{2,1} - M_{1,1})$$

$$= M_{1,2} + M_{2,1} - M_{1,1}.$$

Isotropic full-tensor operator $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$

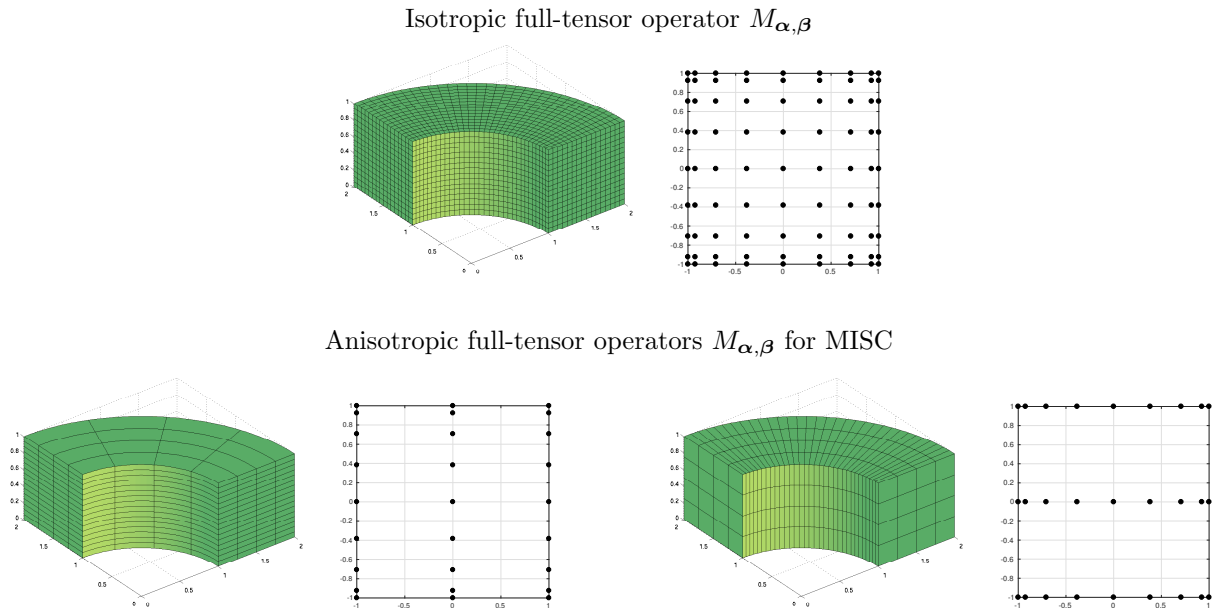Anisotropic full-tensor operators $M_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ for MISC

Figure 4: A full-tensor method requires sampling the stochastic domain with an isotropic Cartesian grid, and solving the PDE on a mesh refined along all physical directions for each sample, see top row. Instead, the MISC method requires that the stochastic domain be sampled with anisotropic Cartesian grids, and the PDE be solved on anisotropic meshes in space, which are combined in such a way that not all the physical and stochastic directions are simultaneously refined: some meshes satisfying this principle are shown in the second and third row. The results need to be linearly combined according the combination-technique formula in (7).

A similar choice of set Λ, i.e., total-degree sets with all weights set to 1, was advocated also in [39, 22]. The last expression should be contrasted with (6), and shows that the most refined operator, $M_{2,2}$, no longer needs to be computed. To devise an optimal strategy for selecting a good multi-index set, we introduce the error decomposition

$$
\begin{aligned}
|\mathbb{E}[\phi] - \mathcal{I}_{\Lambda}^{\mathrm{MISC}}| &= \left| \mathbb{E}[\phi] - \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}] \in \Lambda} \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] \right| \\
&= \left| \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}] \notin \Lambda} \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] \right| \leq \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}] \notin \Lambda} \left| \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] \right| \leq \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}] \notin \Lambda} E_{\boldsymbol{\alpha},\boldsymbol{\beta}},
\end{aligned}
\tag{9}
$$

where we have defined $E_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \left| \boldsymbol{\Delta}^{\mathrm{mix}}[M_{\boldsymbol{\alpha},\boldsymbol{\beta}}] \right|$; $E_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ thus represents the "error contribution" of $[\boldsymbol{\alpha},\boldsymbol{\beta}]$, i.e., the reduction in the approximation error due to having added $[\boldsymbol{\alpha},\boldsymbol{\beta}]$ to the current index-set Λ. Similarly, we define the "work contribution" $W_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ as the work required to add $[\boldsymbol{\alpha},\boldsymbol{\beta}]$ to the current index-set Λ, for instance, summing the degrees of freedom of all the new PDEs that need to be evaluated due to the addition of $[\boldsymbol{\alpha},\boldsymbol{\beta}]$. Thus, it can be easily seen that the strategy that delivers the best choice of Λ consists of adding to Λ only the set of multi-indices with the largest profit $P_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \frac{E_{\boldsymbol{\alpha},\boldsymbol{\beta}}}{W_{\boldsymbol{\alpha},\boldsymbol{\beta}}}$ [11, 12]:

$$
\Lambda_{\epsilon} = \left\{ [\boldsymbol{\alpha},\boldsymbol{\beta}] \in \mathbb{N}^{d+N} : \frac{E_{\boldsymbol{\alpha},\boldsymbol{\beta}}}{W_{\boldsymbol{\alpha},\boldsymbol{\beta}}} \geq \epsilon \right\}, \text{ for some } \epsilon > 0.
\tag{10}
$$

Such a set can be determined by classic adaptive algorithms such as those discussed for quadrature problems in [40, 41, 42], or according to a-priori bounds on the size of $E_{\boldsymbol{\alpha},\boldsymbol{\beta}}$, $W_{\boldsymbol{\alpha},\boldsymbol{\beta}}$; in this work, we consider the latter approach. In particular, all the problems considered in the numerical sections consist of elliptic PDEs, for which an expression of the optimal Λ was derived in [11], under the assumptions that the problem depends on uniform random variables and that the univariate quadrature operator used on the stochastic

domain is built over Clenshaw–Curtis points. In detail, for some positive real values $r_i, c_i, i = 1 \ldots, d$ and $g_j, j = 1 \ldots, N$ that we will define in a moment, the following bounds were introduced in [11]:

$$E_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \mathcal{O}(\widetilde{E}_{\boldsymbol{\alpha},\boldsymbol{\beta}}), \quad \widetilde{E}_{\boldsymbol{\alpha},\boldsymbol{\beta}} = 2^{-\sum_{i=1}^{d} \alpha_i r_i - \sum_{i=1}^{N} g_i 2^{\beta_i} \log_2 e}, \tag{11}$$

$$W_{\boldsymbol{\alpha},\boldsymbol{\beta}} = \mathcal{O}(\widetilde{W}_{\boldsymbol{\alpha},\boldsymbol{\beta}}), \quad \widetilde{W}_{\boldsymbol{\alpha},\boldsymbol{\beta}} = 2^{\sum_{i=1}^{d} \alpha_i c_i + \sum_{i=1}^{N} \beta_i}, \tag{12}$$

from which the following expression for (10) can be derived:

$$\Lambda_{OP}(\boldsymbol{r}, \boldsymbol{c}, \boldsymbol{g}, w) = \left\{ [\boldsymbol{\alpha}, \boldsymbol{\beta}] \in \mathbb{N}^{d+N} : \sum_{i=1}^{d} (r_i + c_i)\alpha_i + \sum_{i=1}^{N} (\beta_i + g_i 2^{\beta_i} \log_2 e) \leq w \right\}, \quad \text{for some } w \in \mathbb{N}, \tag{13}$$

which is reminiscent of the total-degree set (8). The coefficients $r_i, c_i, i = 1 \ldots, d$ and $g_j, j = 1 \ldots, N$ are defined as follows, and can either be determined a-priori or learnt during the execution of the MISC algorithm:

- $r_1, \ldots, r_d$ are the h-convergence rates of the PDE solver for the approximation of $\phi(\mathbf{y})$ for a fixed $\mathbf{y}$, i.e.,

$$|\phi_{\boldsymbol{\alpha}}(\mathbf{y}) - \phi(\mathbf{y})| \leq C \prod_{i=1}^{d} N_{el,i}^{-r_i} = C \prod_{i=1}^{d} 2^{-\alpha_i r_i}, \tag{14}$$

  where α_i can be determined a-priori by standard finite element theory, keeping into account possible corner/edge singularities as well as grading of meshes which may be introduced to mitigate the effects of such singularities; see, e.g., [43, 44]. However, as already mentioned, we will measure r_i numerically.

- $c_1, \ldots, c_d$ are the rates of the increase of the cost of computing the approximation of $\phi(\mathbf{y})$ for a fixed $\mathbf{y}$ as the physical mesh becomes finer,

$$\text{cost}[\phi_{\boldsymbol{\alpha}}(\mathbf{y})] \leq C \prod_{i=1}^{d} 2^{\alpha_i c_i}. \tag{15}$$

  This cost is dominated by assembling and solving the linear system, since evaluating linear functionals of the solution is typically very cheap (e.g., a matrix-vector multiplication). We mention in-passing that relating the cost of assembling and solving the IGA linear system to the number of degrees of freedom is a delicate operation, see e.g. [45]; we simply fit these rates from numerical experiments.

- $g_1, \ldots, g_N$ are the decay rates of the following bound on coefficients of the multivariate Legendre expansion of $\phi(\mathbf{y})$:

$$\phi(\mathbf{y}) = \sum_{\mathbf{i} \in \mathbb{N}^N} \hat{\phi}_{\mathbf{i}} \mathscr{L}_{\mathbf{i}}(\mathbf{y}), \quad |\hat{\phi}_{\mathbf{i}}| \leq C e^{-\sum_{n=1}^{N} g_n i_n}.$$

In practice, we use the implementation reported in Algorithm 1. The stopping criterion in Algorithm 1 above consists of checking that the sum of the error contributions in the margin of the index set Λ, see (2), is smaller than the required tolerance,

$$\sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}] \in \text{Mar}(\Lambda)} E_{\boldsymbol{\alpha},\boldsymbol{\beta}} < \text{TOL}. \tag{16}$$

This choice of criterion is motivated by the error decomposition in (9), where we further approximate the error bound by

$$|\mathbb{E}[\Phi(u)] - \mathcal{I}^{\text{MISC}}(u)| \leq \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}] \notin \Lambda} E_{\boldsymbol{\alpha},\boldsymbol{\beta}} \approx \sum_{[\boldsymbol{\alpha},\boldsymbol{\beta}] \in \text{Mar}(\Lambda)} \widetilde{E}_{\boldsymbol{\alpha},\boldsymbol{\beta}}.$$

The approximation above is only reasonable if the size of the details $E_{\boldsymbol{\alpha},\boldsymbol{\beta}}$ decreases quickly enough, which is true under sufficient smoothness hypotheses for the problem at hand both with respect to the physical variables and the random variables; see, e.g., [46].

Algorithm 1: MISC implementation

Multi Index Stochastic Collocation$(r_1, \ldots, r_d, c_1, \ldots, c_d, w_0, TOL)$

  $\Lambda = \Lambda_{TD}(\mathbf{1}, \mathbf{1}, w_0)$;

  Compute MISC estimate $\mathcal{I}_\Lambda^{\mathrm{MISC}}$ as in (7) ;

  Least squares fit of rates g_i in (11) with $E_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$ for $[\boldsymbol{\alpha}, \boldsymbol{\beta}] \in \Lambda$;

  Compute $\widetilde{E}_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$, $\widetilde{W}_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$ as in (11), (12), and $\widetilde{P}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} = \widetilde{E}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} / \widetilde{W}_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$ for $[\boldsymbol{\alpha}, \boldsymbol{\beta}] \in \mathrm{Mar}(\Lambda)$;

  while $\sum_{[\boldsymbol{\alpha}, \boldsymbol{\beta}] \in Mar(\Lambda)} \widetilde{E}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} > TOL$ **do**

   $\Theta = \left\{ [\boldsymbol{\alpha}, \boldsymbol{\beta}] \in \mathrm{Red}(\Lambda) : \widetilde{P}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} = \max_{[\boldsymbol{\alpha}, \boldsymbol{\beta}] \in \mathrm{Red}(\Lambda)} \left\{ \widetilde{P}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \right\} \right\}$;

   $\Lambda = \Lambda \cup \Theta$;

   Compute MISC estimate $\mathcal{I}_\Lambda^{\mathrm{MISC}}$ as in (7) ;

   Least squares fit of rates g_i in (11) with $E_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$ for $[\boldsymbol{\alpha}, \boldsymbol{\beta}] \in \Lambda$;

   Compute $\widetilde{E}_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$, $\widetilde{W}_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$ as in (11), (12), and $\widetilde{P}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} = \widetilde{E}_{\boldsymbol{\alpha}, \boldsymbol{\beta}} / \widetilde{W}_{\boldsymbol{\alpha}, \boldsymbol{\beta}}$ for $[\boldsymbol{\alpha}, \boldsymbol{\beta}] \in \mathrm{Mar}(\Lambda)$;

  end

end

---

5. Numerical results

In this section, we illustrate the performance of the MISC methodology using some numerical examples. The random variables are considered to be uniformly distributed, and therefore we employ Clenshaw–Curtis quadrature points for the approximation over the stochastic space. Computational times were recorded on single-core runs of MISC on a workstation equipped with Intel Xeon E5 processors with a clock rate of 2.8 GHz and an Ubuntu 16.04 operative system. The IGA solver used is provided by the Matlab/Octave package GeoPDEs, available at `http://rafavzqz.github.io/geopdes/`, see also [47].

5.1. Test 1 - 3d linear elliptic PDE with random diffusion coefficient

In this test we consider a classic UQ benchmark, i.e., a linear elliptic PDE with random diffusion coefficient, for which a vast body of literature exist, see e.g. [23, 48] and references therein,

$$\begin{cases} -\mathrm{div}[a(\mathbf{x}, \mathbf{y}) \nabla u(\mathbf{x}, \mathbf{y})] = 1 & \mathbf{x} \in \mathcal{B}, \\ u(\mathbf{x}, \mathbf{y}) = 0 & \mathbf{x} \in \partial\mathcal{B}. \end{cases}$$

We consider the "thick quarter of ring" in Figure 5-top-left as physical domain; this shape is a typical benchmark geometry in the IGA literature. The random vector $\mathbf{y}$ is composed of $N = 3$ i.i.d. uniform random variables, $y_i \sim \mathcal{U}(-1, 1)$, i.e., $\Gamma = [-1, 1]^3$. The random field $a(\mathbf{x}, \mathbf{y})$ models the variability in the properties of the material, e.g., uneven heat capacity due to imperfections. Because of the peculiar shape of the computational domain, we express the random field in cylindrical coordinates:

$$a([\rho, \theta, z], \mathbf{y}) = e^{c\,\gamma([\rho, \theta, z], \mathbf{y})}, \quad \text{with } c = 4 \text{ and}$$
$$\gamma([\rho, \theta, z], \mathbf{y}) = y_1 \sin(2\theta) \sin(\pi(\rho - 1)) \sin(\pi z) +$$
$$0.4 y_2 \sin(8\theta) \sin(\pi(\rho - 1)) \sin(\pi z) +$$
$$0.1 y_3 \sin(16\theta) \sin(\pi(\rho - 1)) \sin(\pi z).$$

The expression for $\gamma([\rho, \theta, z], \mathbf{y})$ mimics the expression that one would obtain by applying a spectral decomposition like Fourier [49] or Karhunen–Loève [50] to a random field and then truncating it to retain only the most important modes. Three different realizations of the random field can be seen in Figure 5. Note the different scales of the point-wise values of the realizations (due to the magnifying effect of the exponential operation in the definition of the random field), as well as the differences among the frequencies of oscillations. By construction, there exist two real values $0 < m < M$ such that $m < a([\rho, \theta, z], \mathbf{y}) < M$ for
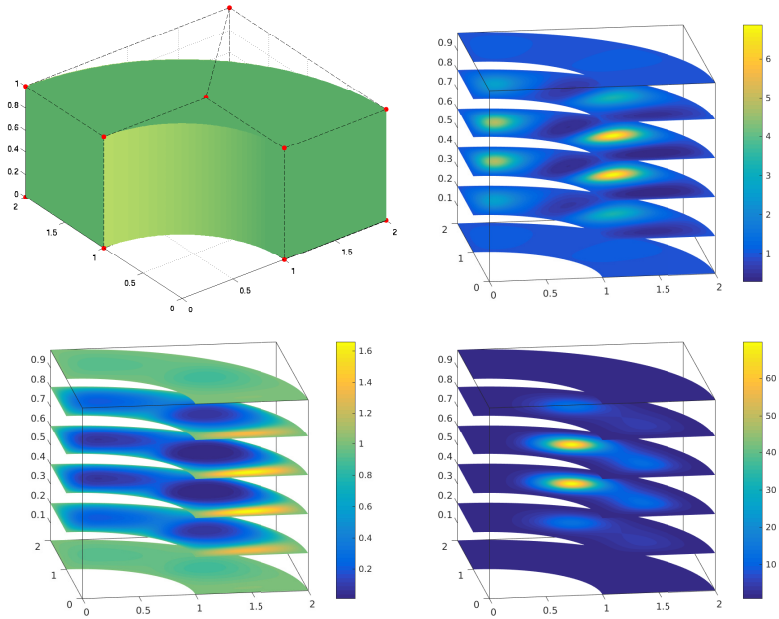
Figure 5: Test 1. Computational domain (top-left) and three different realizations of the random field.

ρ-almost every $\mathbf{y} \in \Gamma$, which guarantees that the problem is well-posed in $V = H_0^1(\mathcal{B})$, and that there exists an optimally convergent approximation of u in $L_\rho^2(\Gamma; V)$ as well as $L_\rho^\infty(\Gamma; V)$ based on either ρ-orthonormal polynomials or interpolation processes; see, e.g., [5, 51]. We are interested in computing the expected value of the integral of the solution over the physical domain, $\Phi(\mathbf{v}) = \int_{\mathcal{B}} \mathbf{v}(\mathbf{x}, \cdot) d\mathbf{x}, \forall \mathbf{v} \in V$.

Concerning the IGA solver, note that the computational domain in this example cannot be described exactly by B-splines, because some of the edges are circle arcs. Therefore, as already pointed out in Section 3, we resort to using NURBS basis functions instead of B-splines, which however does not imply any change in the MISC procedure. Specifically, we employ NURBS of degree $p = 2$ and maximal continuity, C^1; incidentally, note that this means that a degree-elevation operation (i.e., p-refinement) will have to be performed prior to actually starting the MISC computation, since the "thick quarter of ring" geometry is defined by linear polynomials in two out of three parametric directions. The knots in the parametric domain are not uniform; rather, they are scaled towards the edges of the domain according to a power law with exponent set to 3 to capture the edge singularities of the PDE solution for fixed $\mathbf{y}$, and thus improve the convergence of the solution as the mesh-size decreases. We chose the exponent based on a numerical exploration aimed at recovering the optimal convergence of the IGA solver on analogous problems (albeit without randomness), see [28] and references therein for more details. For a fixed realization of the random field, the Galerkin stiffness matrix is typically less sparse for B-splines and NURBS than for finite elements (due to the larger support of B-splines/NURBS, which is proportional to the degree p, [45]), so we employ a direct solver (Matlab's backslash).

We will compare the convergence results of MISC with two different multi-level methods, namely, the Multi-Level Monte Carlo (MLMC, [8]) in the implementation proposed in [52], and its refined version Multi-Index Monte Carlo (MIMC, [9]).

The convergence of MLMC and MIMC for elliptic PDEs with random coefficients was discussed respectively in [8] and in [9] and depends on the rates r_i, c_i in (14) and (15). We have found numerically that r_i, c_i have approximate values of $r_i = [4, 4, 4]$ and $c_i = [1, 1, 1]$, which implies that the computational cost for reaching an accuracy TOL is expected to be $\mathcal{O}(\text{TOL}^{-2})$ for both MLMC and MIMC, see [8, 9] for details. This is the optimal rate for sampling schemes, i.e., most of the sampling is done on the coarsest mesh levels, such that the computational cost is "equivalent to sampling a random variable" (of course, up to the cost
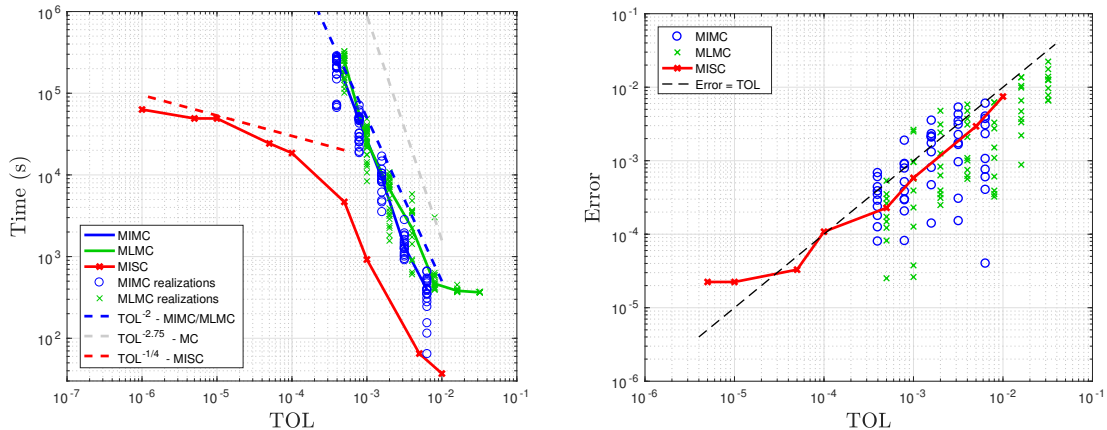
Figure 6: Test 1. Left: Convergence results. Right: Consistency of stopping criterion; the error has been computed against a sufficiently refined MISC solution. Markers for MLMC and MIMC show the value of the error attained for each method run. We have enforced that the error for MLMC and MIMC should be less than TOL with 95% probability (asymptotically).

of solving the linear system corresponding to the coarsest levels, which is still non-negligible). The cost of a standard Monte Carlo analysis where, for an assigned tolerance TOL, we choose a physical mesh and a number of samples in the stochastic domain such that both the deterministic and statistical errors are smaller than TOL/2, would instead be proportional to $\mathcal{O}(\mathrm{TOL}^{-2.75})$; see, e.g., [8].

Concerning MISC, convergence results are available in [11] for the case of a PDE depending on a finite number of random variables (as is the case in the current example) and in [12] for the case of a PDE depending on a countable sequence of random variables. The convergence result for finitely many random variables in [11] depends on the rates r_i, c_i only, i.e., g_i do not play any role; in other words, the approximation over the probability space by tensorized quadrature is expected to converge fast enough that it would not impact the overall convergence rate. More precisely, the result in [11] predicts for the a-priori chosen set in (13) the asymptotic convergence estimate

$$|\mathbb{E}[\Phi(u)] - \mathcal{I}^{\mathrm{MISC}}(u)| \leq C\mathrm{Work}^{-4}(\log\mathrm{Work})^{10},$$

which would in turn imply that the computational cost of reaching an accuracy TOL is expected to be asymptotically $\mathcal{O}(\mathrm{TOL}^{-1/4})$ up to logarithmic terms.

We report the computational results in Figure 6-left. On the horizontal axis we show the tolerance, TOL, which is used in the stopping criterion of each algorithm, and on the vertical axis the recorded computational time: thus, the flatter the convergence curve, the more effective the method, i.e., moving to smaller tolerance does not require a dramatic increase in the computational time. Of course, the implicit assumption behind this plot is that the error actually achieved once the algorithm stops is similar in size to the tolerance enforced as stopping criterion. For MLMC and MIMC, this is guaranteed with "high probability" by the choice of number of samples per mesh, while for MISC we employ the criterion in (16); see Figure 6-right for the effectiveness of these stopping criteria.

We observe from the numerical results that the convergence rates of MLMC and MIMC are roughly 2, in agreement with the theory discussed above. Observe also that the performances of MLMC and MIMC are very close, due to the fact that the convergence of the IGA approximation of the physical problem is very fast ($r_i = 4$), hence sampling the physical problem over anisotropic meshes in space does not provide as significant advantage. MISC features instead a convergence rate of roughly 1/4, as discussed above, after a substantial pre-asymptotic regime which was also expected. Moreover, MISC has a significantly smaller error, which testifies its superior performance compared to the other methods reported, in agreement with previous numerical investigations reported in [11, 12].
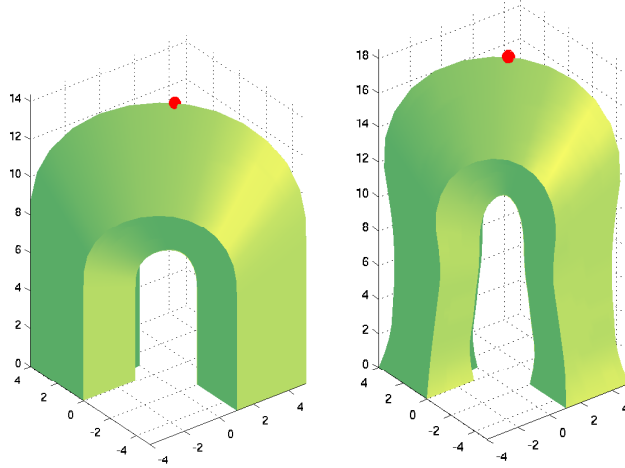
Figure 7: Test 2. Left: Undeformed geometry with **Q** marked by a red dot. Right: Deformation for a random value of the Lamè parameters.

5.2. Test 2 - 3d linear elasticity PDE with uncertain Lamé parameters

In this test we consider a slightly more complex problem than in the previous test, i.e., a linear elastic equation, whose strong form reads

$$\begin{cases} -\mathrm{div}[\sigma(\mathbf{u}(\mathbf{x}, \mathbf{y}))] = \mathbf{f}(\mathbf{x}) & \mathbf{x} \in \mathcal{B}, \\ \mathbf{u}(\mathbf{x}, \mathbf{y}) = 0 & \mathbf{x} \in \partial\mathcal{B}_{\mathrm{clamped}}, \\ \sigma(\mathbf{u}(\mathbf{x}, \mathbf{y})) \cdot \mathbf{n} = 0 & \mathbf{x} \in \partial\mathcal{B}_{\mathrm{free}}. \end{cases}$$

Although still elliptic in nature, this problem is computationally more demanding than the previous case because the unknown is now the tri-dimensional displacement field $\mathbf{u} : \mathcal{B} \to \mathbb{R}^3$. The quantity $\sigma(\mathbf{u}(\mathbf{x}, \mathbf{y}))$ is the Cauchy stress tensor which, upon assuming that the body is undergoing small deformations, can be related to the displacement $\mathbf{u}$ as

$$\sigma(\mathbf{u}(\mathbf{x}, \mathbf{y})) = 2\mu(\mathbf{y}) \frac{\nabla\mathbf{u} + [\nabla\mathbf{u}]^T}{2} + \lambda(\mathbf{y}) \mathrm{div}(\mathbf{u})I,$$

where μ, λ are the Lamé constants and $I \in \mathbb{R}^{3\times3}$ is the identity matrix. The Lamé constants encode the mechanical properties of the material and in this test they are assumed to be random variables, to model imperfect knowledge of such properties. One notable practical example in which this might occur is 3d-printing, where the printer manufacturer guarantees such properties only within a confidence range; see, e.g., `https://www.eos.info/material-m` . In our experiments, we consider typical value ranges for titanium. More specifically, we consider the following ranges for the Young's modulus E and the Poission's ratio ν,

$$E \sim \mathcal{U}(105 \times 10^9 \text{ Pa}, 120 \times 10^9 \text{ Pa}), \quad \nu \sim \mathcal{U}(0.265, 0.34),$$

and then link these to the Lamè parameters by the well-known equations

$$\mu = \frac{E}{2(1 + \nu)}, \quad \lambda = \frac{E\nu}{(1 + \nu)(1 - 2\nu)}.$$

Finally, we let $y_1, y_2 \sim \mathcal{U}([-1, 1])$, and $y_1 \to E$, $y_2 \to \nu$ by linear maps. We consider the "horse-shoe" domain in Figure 7-left as computational domain $\mathcal{B}$. The bottom end $z = 0$ is kept fixed ($\mathcal{B}_{\mathrm{clamped}}$ in the equation) and the rest of the body is free of any constraints ($\mathcal{B}_{\mathrm{free}}$ in the equation). The body is pulled upward by a
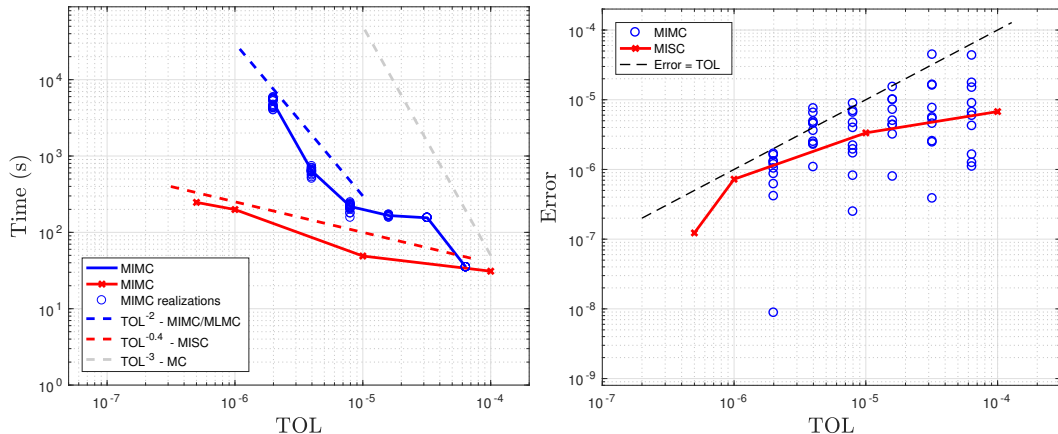
Figure 8: Test 2, computational results. Left: convergence results. Right: consistency of stopping criterion; the error has been computed against a sufficiently refined MISC solution.

vertical force $\mathbf{f} = [0, 0, 10^6]$ N/m^3, and we are interested in computing the expected elongation measured at $\mathbf{Q}$, the point marked by a red dot in Figure 7-left, i.e., $\Phi(\mathbf{u}) = \mathbf{u}(\mathbf{Q})$; an example of deformation obtained by two random values of the Lamè parameters is shown in Figure 7-right (magnified by a factor suitable to make it visible in a plot). It is straightforward to see that the problem is well-posed for ρ-almost every $\mathbf{y} \in \Gamma$ in the vector-valued Hilbert space $V = [H^1(\mathcal{B})]^3$. As in Test 1, the IGA solver is set to use NURBS of degree $p = 2$ with maximal continuity everywhere except along the ridge of the horseshoe, and power-law-scaled knots in the parametric domain. The rates r_i, c_i are assessed numerically as $r_i = [2.5, 2.5, 2.5]$, $c_1 = [1, 1, 1]$, implying convergence $\mathcal{O}(\text{TOL}^{-2})$ for MIMC and MLMC, $\mathcal{O}(\text{TOL}^{-3})$ for Monte Carlo and $\mathcal{O}(\text{TOL}^{-0.4})$ up to preasymptotic and logarithmic terms for MISC. Results are shown in Figure 8, and again confirm the predicted rates and show that MISC is significantly better than MIMC. We omit the convergence results for MLMC, which, as in the previous test, is expected to converge with a trend analogous to MIMC.

We conclude the discussion on this test by mentioning in-passing that in a sense we are artificially increasing the complexity of the problem by using the Lamè parameters instead of the Young's Modulus and Poisson's ratio. Indeed, since both Lamé constants depend linearly on the Young Modulus E, the solution is inversely proportional to E, given the linearity of the PDE at hand. Thus, $\phi(E, \nu) = 1/E \times \phi^*(\nu)$ and therefore the MISC algorithm could be used over four indices (three in space and one for ν) instead of five. Nonetheless, we choose the formulation with five indices because our goal is to showcase the computational efficiency of MISC in high-dimensional problems.

6. Conclusions

In this paper we extended the MISC methodology for solving elliptic PDEs with random coefficients to non-square domains by using isogeometric analysis (IGA) solvers, which fit perfectly into the MISC framework due to their tensor-structure construction. Observe that in principle any tensor method able to deal with non-square geometries in a tensorized fashion, such as Finite Differences, Finite Volumes, and $\mathbb{Q}^k$ finite element on hexahedral meshes could be used as well. We have shown the effectiveness of MISC-IGA using a few numerical test cases. Research directions currently under investigations that could benefit from this approach are:

- forward UQ problems on domains with uncertain shape (and as a further step, shape optimization under uncertainty): indeed, the B-splines/NURBS representation of a geometry allows us to describe deviations from a nominal domain in a very straightforward manner.

- UQ problems defined on unions of disjoint subdomains ("patches" in the IGA literature). In this case, MISC could be allowed to choose different meshes in each subdomain, resulting in anisotropic meshes

that would refine only a few of the subdomains. The resolution of the PDE on the corresponding non-conformal meshes could be performed by resorting to e.g. Lagrange multipliers [53, 54, 55] or discontinuous Galerkin methods [56, 57, 58] to enforce continuity at the interfaces of the subdomains.

Acknowledgment

References

[1] R. Ghanem, D. Higdon, H. Owhadi, Handbook of Uncertainty Quantification, Handbook of Uncertainty Quantification, Springer International Publishing, 2016.

[2] T. Sullivan, Introduction to Uncertainty Quantification, Texts in Applied Mathematics, Springer International Publishing, 2015.

[3] R. Smith, Uncertainty Quantification: Theory, Implementation, and Applications, Computational Science and Engineering, Society for Industrial and Applied Mathematics, 2013.

[4] F. Nobile, R. Tempone, C. Webster, An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data, SIAM J. Numer. Anal. 46 (5) (2008) 2411–2442.

[5] F. Nobile, L. Tamellini, R. Tempone, Convergence of quasi-optimal sparse-grid approximation of Hilbert-space-valued functions: application to random elliptic PDEs, Numerische Mathematik 134 (2) (2016) 343–388.

[6] A. Cohen, R. Devore, C. Schwab, Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE'S, Anal. Appl. (Singap.) 9 (1) (2011) 11–47.

[7] A. Chkifa, A. Cohen, R. Devore, C. Schwab, Sparse adaptive Taylor approximation algorithms for parametric and stochastic elliptic PDEs, ESAIM: Mathematical Modelling and Numerical Analysis 47 (1) (2013) 253–280.

[8] K. Cliffe, M. Giles, R. Scheichl, A. Teckentrup, Multilevel monte carlo methods and applications to elliptic pdes with random coefficients, Computing and Visualization in Science 14 (1) (2011) 3–15.

[9] A.-L. Haji-Ali, F. Nobile, R. Tempone, Multi-index Monte Carlo: when sparsity meets sampling, Numerische Mathematik (2015) 1–40.

[10] B. Peherstorfer, K. Willcox, M. Gunzburger, Survey of multifidelity methods in uncertainty propagation, inference, and optimization, SIAM Review 60 (3) (2018) 550–591.

[11] A. Haji-Ali, F. Nobile, L. Tamellini, R. Tempone, Multi-index stochastic collocation for random {PDEs}, Computer Methods in Applied Mechanics and Engineering 306 (2016) 95 – 122.

[12] A.-L. Haji-Ali, F. Nobile, L. Tamellini, R. Tempone, Multi-index Stochastic Collocation convergence rates for random PDEs with parametric regularity, Foundations of Computational Mathematics 16 (6) (2016) 1555–1605.

[13] H.-J. Bungartz, M. Griebel, D. Röschke, C. Zenger, Pointwise convergence of the combination technique for the Laplace equation, East-West J. Numer. Math. 2 (1994) 21–45.

[14] H. Bungartz, M. Griebel, Sparse grids, Acta Numer. 13 (2004) 147–269.

[15] M. Griebel, M. Schneider, C. Zenger, A combination technique for the solution of sparse grid problems, in: P. de Groen, R. Beauwens (Eds.), Iterative Methods in Linear Algebra, IMACS, Elsevier, North Holland, 1992, pp. 263–281.

[16] M. Hegland, J. Garcke, V. Challis, The combination technique and some generalisations, Linear Algebra and its Applications 420 (23) (2007) 249 – 275.

[17] S. Smolyak, Quadrature and interpolation formulas for tensor products of certain classes of functions, Dokl. Akad. Nauk SSSR 4 (1963) 240–243.

[18] V. Barthelmann, E. Novak, K. Ritter, High dimensional polynomial interpolation on sparse grids, Adv. Comput. Math. 12 (4) (2000) 273–288.

[19] A. Quarteroni, R. Sacco, F. Saleri, Numerical mathematics, 2nd Edition, Vol. 37 of Texts in Applied Mathematics, Springer-Verlag, Berlin, 2007.

[20] A. L. Teckentrup, P. Jantsch, C. G. Webster, M. Gunzburger, A Multilevel Stochastic Collocation Method for Partial Differential Equations with Random Input Data, SIAM/ASA Journal on Uncertainty Quantification 3 (1) (2015) 1046–1074.

[21] H. W. van Wyk, Multilevel sparse grid methods for elliptic partial differential equations with random coefficients, arXiv arXiv:1404.0963, e-print (2014).

[22] H. Harbrecht, M. Peters, M. Siebenmorgen, On multilevel quadrature for elliptic stochastic partial differential equations, in: Sparse Grids and Applications, Vol. 88 of Lecture Notes in Computational Science and Engineering, Springer, 2013, pp. 161–179.

[23] I. Babuška, F. Nobile, R. Tempone, A stochastic collocation method for elliptic partial differential equations with random input data, SIAM Review 52 (2) (2010) 317–355.

[24] D. Xiu, J. Hesthaven, High-order collocation methods for differential equations with random inputs, SIAM J. Sci. Comput. 27 (3) (2005) 1118–1139.

[25] T. J. R. Hughes, J. A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering 194 (39) (2005) 4135–4195.

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

[27] L. Beirao Da Veiga, A. Buffa, G. Sangalli, R. Vázquez, Mathematical analysis of variational isogeometric methods, Acta Numerica 23 (2014) 157–287.

[28] J. Beck, G. Sangalli, L. Tamellini, A sparse-grid isogeometric solver, Computer Methods in Applied Mechanics and Engineering 335 (–) (2018) 128–151.

[29] M. Montardini, G. Sangalli, L. Tamellini, Optimal-order isogeometric collocation at Galerkin superconvergent points, Computer Methods in Applied Mechanics and Engineering.

[30] H. Gomez, L. De Lorenzis, The variational collocation method, Computer Methods in Applied Mechanics and Engineering 309 (2016) 152–181.

[31] C. Anitescu, Y. Jia, Y. J. Zhang, T. Rabczuk, An isogeometric collocation method using superconvergent points, Computer Methods in Applied Mechanics and Engineering 284 (2015) 1073–1097.

[32] H. Casquero, L. Liu, Y. Zhang, A. Reali, H. Gomez, Isogeometric collocation using analysis-suitable t-splines of arbitrary degree, Computer Methods in Applied Mechanics and Engineering 301 (2016) 164–186.

[33] O. G. Ernst, B. Sprungk, L. Tamellini, Convergence of Sparse Collocation for Functions of Countably Many Gaussian Random Variables (with Application to Lognormal Elliptic Diffusion Problems), SIAM Journal on Numerical Analysis 56 (2) (2018) 877–905.

[34] L. N. Trefethen, Is Gauss quadrature better than Clenshaw-Curtis?, SIAM Rev. 50 (1) (2008) 67–87.

[35] F. Nobile, L. Tamellini, R. Tempone, Comparison of ClenshawCurtis and Leja Quasi-Optimal Sparse Grids for the Approximation of Random PDEs, in: R. M. Kirby, M. Berzins, J. S. Hesthaven (Eds.), Spectral and High Order Methods for Partial Differential Equations - ICOSAHOM '14, Vol. 106 of Lecture Notes in Computational Science and Engineering, Springer International Publishing, 2015, pp. 475–482.

[36] A. Narayan, J. D. Jakeman, Adaptive Leja Sparse Grid Constructions for Stochastic Collocation and High-Dimensional Approximation, SIAM Journal on Scientific Computing 36 (6) (2014) A2952–A2983.

[37] A. Chkifa, On the Lebesgue constant of Leja sequences for the complex unit disk and of their real projection, Journal of Approximation Theory 166 (0) (2013) 176 – 200.

[38] J. Bäck, F. Nobile, L. Tamellini, R. Tempone, Stochastic spectral Galerkin and collocation methods for PDEs with random coefficients: a numerical comparison, in: Spectral and High Order Methods for Partial Differential Equations, Vol. 76 of Lecture Notes in Computational Science and Engineering, Springer, 2011, pp. 43–62.

[39] M. Bieri, A sparse composite collocation finite element method for elliptic spdes., SIAM Journal on Numerical Analysis 49 (6) (2011) 2277–2301.

[40] T. Gerstner, M. Griebel, Dimension-adaptive tensor-product quadrature, Computing 71 (1) (2003) 65–87.

[41] C. Schillings, C. Schwab, Sparse, adaptive Smolyak quadratures for Bayesian inverse problems, Inverse Problems 29 (6).

[42] F. Nobile, L. Tamellini, F. Tesei, R. Tempone, An adaptive sparse grid algorithm for elliptic PDEs with lognormal diffusion coefficient, in: J. Garcke, D. Pflüger (Eds.), Sparse Grids and Applications – Stuttgart 2014, Vol. 109 of Lecture Notes in Computational Science and Engineering, Springer International Publishing Switzerland, 2016, pp. 191–220.

[43] B. Guo, I. Babuška, The h-p version of the finite element method, Computational Mechanics 1 (3) (1986) 203–220.

[44] I. Babuška, T. Strouboulis, The finite element method and its reliability, Numerical Mathematics and Scientific Computation, The Clarendon Press Oxford University Press, New York, 2001.

[45] D. Schillinger, J. Evans, A. Reali, M. Scott, T. Hughes, Isogeometric collocation: cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations, Comput. Methods Appl. Mech. Engrg. 267 (2013) 170 – 232.

[46] D. S. Guignard, F. Nobile, A posteriori error estimation for the stochastic collocation finite element method, Mathicse Report nr 24.2017.

[47] R. Vazquez, A new design for the implementation of isogeometric analysis in octave and matlab: Geopdes 3.0, Computers & Mathematics with Applications 72 (3) (2016) 523 – 554.

[48] A. Cohen, R. DeVore, Approximation of high-dimensional parametric pdes, Acta Numerica 24 (2015) 1159.

[49] J. Beck, F. Nobile, L. Tamellini, R. Tempone, A Quasi-optimal Sparse Grids Procedure for Groundwater Flows, in: Spectral and High Order Methods for Partial Differential Equations - ICOSAHOM 2012, Vol. 95 of Lecture Notes in Computational Science and Engineering, Springer, 2014, pp. 1–16.

[50] R. G. Ghanem, P. D. Spanos, Stochastic finite elements: a spectral approach, Springer-Verlag, New York, 1991.

[51] A. Chkifa, A. Cohen, C. Schwab, Breaking the curse of dimensionality in sparse polynomial approximation of parametric PDEs, Journal de Mathmatiques Pures et Appliques 103 (2) (2015) 400 – 428.

[52] N. Collier, A. Haji-Ali, F. Nobile, E. von Schwerin, R. Tempone, A continuation multilevel monte carlo algorithm, BIT Numerical Mathematics (2014) 1–34.

[53] E. Brivadis, A. Buffa, B. Wohlmuth, L. Wunderlich, Isogeometric mortar methods, Computer Methods in Applied Mechanics and Engineering 284 (2015) 292 – 319, isogeometric Analysis Special Issue.

[54] C. Hesch, P. Betsch, Isogeometric analysis and domain decomposition methods, Computer Methods in Applied Mechanics and Engineering 213216 (2012) 104 – 112.

[55] İ. Temizer, P. Wriggers, T. Hughes, Three-dimensional mortar-based frictional contact treatment in isogeometric analysis with NURBS, Comput. Methods Appl. Mech. Engrg. 209–212 (2012) 115 – 128.

[56] P. Antonietti, I. Mazzieri, A. Quarteroni, F. Rapetti, Non-conforming high order approximations of the elastodynamics equation, Computer Methods in Applied Mechanics and Engineering 209212 (2012) 212 – 238.

[57] J. Nitsche, Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind, Abh. Math. Sem. Univ. Hamburg 36 (1971) 9–15, collection of articles dedicated to Lothar Collatz on his sixtieth birthday.

[58] D. N. Arnold, F. Brezzi, B. Cockburn, L. D. Marini, Unified analysis of discontinuous galerkin methods for elliptic problems, SIAM Journal on Numerical Analysis 39 (5) (2002) 1749–1779. doi:10.1137/S0036142901384162.

Recent titles from the IMATI-REPORT Series:

2018

18-01: *Arbitrary-order time-accurate semi-Lagrangian spectral approximations of the Vlasov-Poisson system*, L. Fatone, D. Funaro, G. Manzini.

18-02: *A study on 3D shape interaction in virtual reality*, E. Cordeiro, F. Giannini, M. Monti, A. Ferreira.

18-03: *Standard per la gestione e procedure di validazione di dati meteo da sensori eterogenei e distribuiti,* A. Clematis, B. Bonino, A. Galizia.

18-04: *Strumenti e procedure per la gestione e la visualizzazione di dati meteo prodotti da sensori eterogenei e distribuiti tramite interfacce web basate su mappe geografiche,* L. Roverelli, G. Zereik, B. Bonino, A. Gallizia, A. Clematis.

18-05: *TopChart: from functions to quadrangulations*, T. Sorgente, S. Biasotti, M. Livesu, M. Spagnuolo.

18-06: *Adaptive sampling of enviromental variables (ASEV)*, S. Berretta, D. Cabiddu, S. Pittaluga, M. Mortara, M. Spagnuolo, M. Vetuschi Zuccolini.

18-07: *Multi-criteria similarity assessment for CAD assembly models retrieval*, K. Lupinetti, F. Giannini, M. Monti, J,-P. Pernot.

18-08: *IGA-based Multi-Index Stochastic Collocation for random PDEs on arbitrary domains*, J. Beck, L. Tamellini, R. Tempone