Two-level a posteriori error estimation for adaptive multilevel stochastic Galerkin FEM*

Alex Bespalov†, Dirk Praetorius‡, and Michele Ruggeri‡

Abstract. The paper considers a class of parametric elliptic partial differential equations (PDEs), where the coefficients and the right-hand side function depend on infinitely many (uncertain) parameters. We introduce a two-level *a posteriori* estimator to control the energy error in multilevel stochastic Galerkin approximations for this class of PDE problems. We prove that the two-level estimator always provides a lower bound for the unknown approximation error, while the upper bound is equivalent to a saturation assumption. We propose and empirically compare three adaptive algorithms, where the structure of the estimator is exploited to perform spatial refinement as well as parametric enrichment. The paper also discusses implementation aspects of computing multilevel stochastic Galerkin approximations.

Key words. adaptive methods, a posteriori error analysis, two-level error estimation, multilevel stochastic Galerkin method, finite element method, parametric PDEs

AMS subject classifications. 35R60, 65C20, 65N15, 65N30, 65N50

1. Introduction.

1.1. Multilevel stochastic Galerkin FEM.

The effective numerical solution of partial differential equations (PDEs) with uncertain or parameter-dependent inputs requires non-trivial computational methods and efficient algorithms. Stochastic Galerkin finite element methods (SGFEMs) provide a powerful alternative to traditional sampling techniques for such problems, in particular, when the inputs and solutions are sufficiently smooth functions of parameters (for comparison between SGFEM and popular sampling methods, such as Monte Carlo and stochastic collocation finite element methods, we refer to [32, 1, 29]). Appropriate construction of the underlying approximation spaces and adaptivity are the keys to computationally efficient SGFEM implementations, particularly in the case of inputs depending on infinitely many uncertain parameters.

Stochastic Galerkin approximations are typically represented in terms of a finite generalized polynomial chaos (gPC) expansion with spatial coefficients residing in finite element spaces. If all spatial coefficients reside in the same finite element space, the corresponding SGFEM approximation space is termed *single-level* and its dimension has a multiplicative representation (i.e., the total number of degrees of freedom is equal to the number of active terms in the gPC expansion multiplied by the dimension of the finite element space). An alternative

*Submitted to the editors DATE.

Funding: The work of the first author was supported by the EPSRC under grant EP/P013791/1 and by The Alan Turing Institute under the EPSRC grant EP/N510129/1. The work of the second and third authors was supported by the Austrian Science Fund (FWF) under grants F65 and P33216.

†School of Mathematics, University of Birmingham, Edgbaston, Birmingham B15 2TT, UK (a.bespalov@bham.ac.uk).

‡Institute of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8–10, 1040 Vienna, Austria (dirk.praetorius@asc.tuwien.ac.at, michele.ruggeri@asc.tuwien.ac.at).

to this is a more flexible *multilevel* construction, where spatial gPC-coefficients may reside in different finite element spaces. In this case, the dimension of the SGFEM approximation space admits an additive representation (i.e., the total number of degrees of freedom is equal to the sum of dimensions of all involved finite element spaces).

Multilevel SGFEMs have emerged in [14, 15, 31]. These works have provided a theoretical benchmark for convergence analysis of the SGFEM. In particular, under some assumptions on parametric inputs, they have proved the existence of a sequence of multilevel approximation spaces such that the errors in the associated Galerkin solutions converge to zero with an optimal rate (i.e., with the rate of the chosen FEM for the corresponding parameter-free problem). Practical realizations of adaptive algorithms that generate these sequences of approximation spaces and Galerkin solutions have been developed in [19] and more recently in [16]. While the predicted optimal convergence behavior of multilevel SGFEM approximations has been observed numerically for parametric problems with spatially regular [16] and spatially singular [19] solutions, a provable optimality result for the developed adaptive algorithms for multilevel SGFEMs has so far remained an open problem.

Multilevel approaches based on hierarchies of spatial approximations have been studied also for sampling methods. Remaining within the context of the numerical approximation of elliptic PDEs with uncertain data, we refer, e.g., to [13] for multilevel Monte Carlo (MLMC) methods, to [35] for multilevel quasi-Monte Carlo methods, and to [45] for multilevel stochastic collocation (MLSC) methods. Adaptive strategies for MLMC and MLSC have been developed recently in [34] and [36], respectively.

1.2. Main contributions and outline of the paper. In this paper, we consider the same parametric model problem as studied in the above cited works [19, 16] (among very many other works)—the steady-state diffusion equation with a spatially varying coefficient that has affine dependence on infinitely many parameters.

For the numerical solution of this problem, we propose an adaptive algorithm that iterates the following loop of four modules:

$$\mathsf{SOLVE} \longrightarrow \mathsf{ESTIMATE} \longrightarrow \mathsf{MARK} \longrightarrow \mathsf{REFINE}$$

(see Algorithm 7 below). Let us briefly describe each of these modules emphasizing their specific features pertinent to the multilevel SGFEM.

• SOLVE: In this module, the multilevel SGFEM approximation is computed as a finite gPC expansion with coefficients in the current set of finite element spaces. One of the challenges in implementing multilevel SGFEMs is the efficient computation of nonsquare stiffness matrices associated with two different finite element meshes. The existing implementations either rely on projection techniques to compute these stiffness matrices approximately (see [19, 24]) or restrict themselves to spatial discretizations on nested uniform meshes (see [16]). In this paper, we propose an effective procedure for *direct* computation of nonsquare stiffness matrices for a pair of general, not necessarily nested meshes obtained from the same coarse mesh by finitely many steps of a fixed mesh refinement rule (in our case, newest vertex bisection). SGFEMs give rise to very large linear systems with block structure. Solving such linear systems numerically is a non-trivial task that stimulated the development and analysis of iterative solvers, preconditioning strategies, and low-rank approximation techniques; see, e.g.,

[28, 38, 46, 43, 27, 17, 3]. In order to solve the linear systems arising in the multilevel SGFEM, we use a bespoke implementation of the Minimum Residual method from [42] with the mean-based preconditioner from [28, 38].

• ESTIMATE: In this module, the error between the (unknown) exact solution and the multilevel SGFEM approximation is estimated by suitable error indicators. The *a posteriori* error estimation in multilevel SGFEMs has been addressed in [19, 16]. While explicit residual-based error estimators are employed in [19], hierarchical-type error estimators are analyzed in [16]. Building on the ideas in our recent works for a single-level SGFEM [6, 5], in this paper, we propose a novel *a posteriori* error estimation strategy for multilevel SGFEM approximations. The estimator, which combines a two-level spatial estimator and a hierarchical parametric estimator, allows to estimate the error contributions from finite element discretizations in the physical domain and those from the approximation (obtained via truncation) in the parameter domain independently from each other. We prove that the combined error estimator is always efficient, i.e., up to a multiplicative constant, it provides a lower bound for the energy error, whereas its reliability (i.e., the upper bound for the error) is equivalent to a saturation assumption (see subsection 4.1 below). This choice of the error estimation strategy is motivated by a recent success in proving optimal convergence rates for adaptive algorithms for deterministic problems; see [39]. Thus, we see our *a posteriori* error analysis in this paper as an important step towards proving the optimality result for adaptive multilevel SGFEM approximations by extending the methodology developed in [39] to the parametric setting.

• MARK: In this module, some of the spatial and parametric components of the current multilevel SGFEM approximation are selected for refinement by assessing the values of the error indicators computed in the module ESTIMATE. The application of the module MARK highlights key differences between adaptive multilevel SGFEM and multilevel sampling methods (such as MLMC and MLSC). The latter methods typically require the number of active parameters in approximations to be fixed *a priori*, and the balance between the spatial error (e.g., due to finite element discretization) and the parametric error (e.g., due to Monte Carlo sampling or high-dimensional polynomial interpolation) is achieved by employing *a priori* bounds for spatial errors and by minimizing the cost functional (see, e.g., [32, section 3.4] for MLMC and [45] for MLSC). Adaptive SGFEM algorithms are fundamentally different. Firstly, they require no sampling. Secondly, the selection of active parameters, the truncation of the gPC expansion, and the balance between spatial and parametric components of approximation errors are performed *automatically* using *a posteriori* error indicators and the adopted marking criterion. The choice of the marking criterion is critical. In this work, we propose three different marking strategies, all based on the bulk-chasing criterion proposed in the deterministic setting by Dörfler [18]. In addition to two standard marking criteria that lead to separate refinement of either spatial or parametric components at each iteration of the adaptive loop (see, e.g., [19, 20, 10, 5, 16]), we also exploit the multilevel structure of the approximation space and perform a *combined* refinement at each iteration by employing Dörfler marking on the joint set of spatial and parametric error indicators. While combined refinement is prohibitively expensive for single-level SGFEM (because of the multiplicative dependence of the dimension of the discrete space on the number of active terms in the gPC expansion), we stress that multilevel SGFEM allows for combined refinement and our experiments indicate optimal convergence behavior.

• REFINE: In this module, the finite-dimensional space for computing the next multilevel SGFEM approximation is generated by enriching the current finite-dimensional space with the spatial and parametric components selected in the module MARK. Specifically, (i) the finite element spaces are enriched by refining all marked elements of the current spatial meshes; and (ii) new terms are added to the gPC expansion.

The paper is organized as follows. Section 2 introduces the model parametric problem and its weak formulation. In section 3, we describe the main ingredients of the multilevel SGFEM discretization, introduce the multilevel approximation space, and define the corresponding Galerkin solution. Section 4 is focused on the *a posteriori* error analysis of multilevel SGFEM approximations and includes the main theoretical result of this paper (Theorem 2). Adaptive algorithms with three different marking criteria are formulated in section 5, whereas implementation aspects of computing multilevel SGFEM approximations are discussed in section 6. The effectiveness of our error estimation strategy and the performance of the proposed adaptive algorithms are assessed in a series of numerical experiments presented in section 7.

2. Problem formulation. Let $D \subset \mathbb{R}^d$ $(d = 2, 3)$ be a bounded Lipschitz domain with polytopal boundary ∂D and let $\Gamma := \prod_{m=1}^{\infty}[-1, 1]$ denote the infinitely-dimensional hypercube. We consider the elliptic boundary value problem

$$\begin{aligned} -\nabla \cdot (\boldsymbol{a}\nabla \boldsymbol{u}) &= \boldsymbol{f} \qquad \text{in } D \times \Gamma, \\ \boldsymbol{u} &= 0 \qquad \text{on } \partial D \times \Gamma, \end{aligned} \tag{2.1}$$

where the scalar coefficient $\boldsymbol{a}$ and the right-hand side function $\boldsymbol{f}$ (and, hence, the solution $\boldsymbol{u}$) depend on a countably infinite number of scalar parameters, i.e., $\boldsymbol{a} = \boldsymbol{a}(x, \mathbf{y})$, $\boldsymbol{f} = \boldsymbol{f}(x, \mathbf{y})$, and $\boldsymbol{u} = \boldsymbol{u}(x, \mathbf{y})$ with $x \in D$ and $\mathbf{y} = (y_m)_{m\in\mathbb{N}} \in \Gamma$. For the coefficient $\boldsymbol{a}$, we assume linear dependence on the parameters, i.e.,

$$\boldsymbol{a}(x, \mathbf{y}) = a_0(x) + \sum_{m=1}^{\infty} y_m a_m(x) \quad \text{for all } x \in D \text{ and } \mathbf{y} \in \Gamma. \tag{2.2}$$

We assume that $\boldsymbol{f} \in L^2_\pi(\Gamma; H^{-1}(D))$, where $\pi = \pi(\mathbf{y})$ is a measure on $(\Gamma, \mathcal{B}(\Gamma))$ with $\mathcal{B}(\Gamma)$ being the Borel σ-algebra on Γ. We assume that $\pi(\mathbf{y})$ is the product of symmetric Borel probability measures π_m on $[-1, 1]$, i.e., $\pi(\mathbf{y}) = \prod_{m=1}^{\infty} \pi_m(y_m)$.

For each $m \in \mathbb{N}_0$, the scalar functions $a_m \in L^\infty(D)$ in (2.2) are required to satisfy the following inequalities (cf. [40, Section 2.3]):

$$0 < a_0^{\min} \le a_0(x) \le a_0^{\max} < \infty \quad \text{for almost all } x \in D, \tag{2.3}$$

$$\tau := \frac{1}{a_0^{\min}} \Big\| \sum_{m=1}^{\infty} |a_m| \Big\|_{L^\infty(D)} < 1 \quad \text{and} \quad \sum_{m=1}^{\infty} \|a_m\|_{L^\infty(D)} < \infty. \tag{2.4}$$

With the Sobolev space $\mathbb{X} := H^1_0(D)$, consider the Bochner space $\mathbb{V} := L^2_\pi(\Gamma; \mathbb{X})$. Define the following bilinear forms on $\mathbb{V}$:

$$B_0(\boldsymbol{u}, \boldsymbol{v}) := \int_\Gamma \int_D a_0(x) \nabla \boldsymbol{u}(x, \mathbf{y}) \cdot \nabla \boldsymbol{v}(x, \mathbf{y}) \, \mathrm{d}x \, \mathrm{d}\pi(\mathbf{y}), \tag{2.5}$$

$$B(\boldsymbol{u}, \boldsymbol{v}) := B_0(\boldsymbol{u}, \boldsymbol{v}) + \sum_{m=1}^{\infty} \int_\Gamma \int_D y_m a_m(x) \nabla \boldsymbol{u}(x, \mathbf{y}) \cdot \nabla \boldsymbol{v}(x, \mathbf{y}) \, \mathrm{d}x \, \mathrm{d}\pi(\mathbf{y}). \tag{2.6}$$

An elementary computation shows that assumptions (2.2)–(2.4) ensure that the bilinear forms $B_0(\cdot,\cdot)$ and $B(\cdot,\cdot)$ are symmetric, continuous, and elliptic on $\mathbb{V}$. Let $|\!|\!| \cdot |\!|\!|$ (resp., $|\!|\!| \cdot |\!|\!|_0$) denote the norm induced by $B(\cdot,\cdot)$ (resp., $B_0(\cdot,\cdot)$). Then, there holds

$$\lambda\, |\!|\!| \boldsymbol{v} |\!|\!|_0^2 \le |\!|\!| \boldsymbol{v} |\!|\!|^2 \le \Lambda\, |\!|\!| \boldsymbol{v} |\!|\!|_0^2 \quad \text{for all } \boldsymbol{v} \in \mathbb{V}, \tag{2.7}$$

where $\lambda := 1-\tau$ and $\Lambda := 1+\tau$. Note that $0 < \lambda < 1 < \Lambda < 2$.

The parametric problem (2.1) is understood in the weak sense: Given $\boldsymbol{f} \in L^2_\pi(\Gamma; H^{-1}(D))$, find $\boldsymbol{u} \in \mathbb{V}$ such that

$$B(\boldsymbol{u},\boldsymbol{v}) = F(\boldsymbol{v}) := \int_\Gamma \int_D \boldsymbol{f}(x,\mathbf{y})\boldsymbol{v}(x,\mathbf{y})\,\mathrm{d}x\,\mathrm{d}\pi(\mathbf{y}) \quad \text{for all } \boldsymbol{v} \in \mathbb{V}. \tag{2.8}$$

The existence and uniqueness of the solution $\boldsymbol{u} \in \mathbb{V}$ to (2.8) follow by the Riesz theorem.

3. Multilevel stochastic Galerkin FEM discretization.

The weak formulation (2.8) is discretized by constructing a finite-dimensional subspace $\mathbb{V}_\bullet \subset \mathbb{V}$ and using the Galerkin projection onto $\mathbb{V}_\bullet$. In the spirit of [14, 31, 19, 16], this work considers approximation spaces $\mathbb{V}_\bullet$ with a *multilevel* structure. Specifically, these spaces are constructed from tensor products of different finite element subspaces of $H^1_0(D)$ and multivariable polynomial spaces on Γ. We describe each of these ingredients in the next two subsections.

3.1. Finite element spaces and mesh refinement.

Let $\mathcal{T}_\bullet$ be a *mesh*, i.e., a conforming triangulation of D into compact non-degenerate simplices $T \in \mathcal{T}_\bullet$ (i.e., triangles for $d = 2$) and denote by $\mathcal{N}_\bullet$ the set of vertices of $\mathcal{T}_\bullet$.

We consider the space of continuous piecewise linear finite elements

$$\mathbb{X}_\bullet := \mathcal{S}^1_0(\mathcal{T}_\bullet) := \{v_\bullet \in \mathbb{X} : v_\bullet|_T \text{ is affine for all } T \in \mathcal{T}_\bullet\} \subset \mathbb{X} = H^1_0(D).$$

For $z \in \mathcal{N}_\bullet$, let $\varphi_{\bullet,z}$ be the associated hat function, i.e., $\varphi_{\bullet,z}$ is piecewise affine, globally continuous, and satisfies the Kronecker property $\varphi_{\bullet,z}(z') = \delta_{zz'}$ for all $z' \in \mathcal{N}_\bullet$. Recall that $\{\varphi_{\bullet,z} : z \in \mathcal{N}_\bullet \setminus \partial D\}$ is the standard basis of $\mathbb{X}_\bullet$.

For mesh refinement, we employ newest vertex bisection (NVB); see, e.g., [44, 33]. We assume that any mesh $\mathcal{T}_\bullet$ employed for the spatial discretization can be obtained by applying NVB refinement(s) to a given initial (coarse) mesh $\mathcal{T}_0$. In particular, we denote by $\mathrm{refine}(\mathcal{T}_0)$ the set of all meshes obtained from $\mathcal{T}_0$ by finitely many steps of refinement.

For a given mesh $\mathcal{T}_\bullet \in \mathrm{refine}(\mathcal{T}_0)$, let $\widehat{\mathcal{T}}_\bullet$ be the coarsest NVB refinement of $\mathcal{T}_\bullet$ such that: (i) for $d = 2$, all edges of $\mathcal{T}_\bullet$ have been bisected once (which corresponds to uniform refinement of all elements by three bisections); (ii) for $d = 3$, all faces contain an interior node (we refer to [25] for further discussion). Then, $\widehat{\mathcal{N}}_\bullet$ denotes the set of vertices of $\widehat{\mathcal{T}}_\bullet$, and $\mathcal{N}_\bullet^+ := (\widehat{\mathcal{N}}_\bullet \setminus \mathcal{N}_\bullet) \setminus \partial D$ is the set of new interior vertices created by this refinement of $\mathcal{T}_\bullet$.

For a set of marked vertices $\mathcal{M}_\bullet \subseteq \mathcal{N}_\bullet^+$, let $\mathcal{T}_\circ := \mathrm{refine}(\mathcal{T}_\bullet, \mathcal{M}_\bullet)$ be the coarsest NVB refinement of $\mathcal{T}_\bullet$ such that $\mathcal{M}_\bullet \subset \mathcal{N}_\circ$, i.e., all marked vertices are vertices of $\mathcal{T}_\circ$. Since NVB is a binary refinement rule, this implies that $\mathcal{N}_\bullet \subseteq \mathcal{N}_\circ \subseteq \widehat{\mathcal{N}}_\bullet$ and $(\mathcal{N}_\circ \setminus \mathcal{N}_\bullet) \setminus \partial D = \mathcal{N}_\bullet^+ \cap \mathcal{N}_\circ$. In particular, the choices $\mathcal{M}_\bullet = \emptyset$ and $\mathcal{M}_\bullet = \mathcal{N}_\bullet^+$ lead to the meshes $\mathcal{T}_\bullet = \mathrm{refine}(\mathcal{T}_\bullet, \emptyset)$ and $\widehat{\mathcal{T}}_\bullet = \mathrm{refine}(\mathcal{T}_\bullet, \mathcal{N}_\bullet^+)$, respectively.

The finite element space associated with $\widehat{\mathcal{T}}_\bullet$ is denoted by $\widehat{\mathbb{X}}_\bullet := \mathcal{S}^1_0(\widehat{\mathcal{T}}_\bullet)$, and $\{\widehat{\varphi}_{\bullet,z} : z \in \widehat{\mathcal{N}}_\bullet \setminus \partial D\}$ is the corresponding basis of hat functions. Later, we will exploit the (H^1-stable) two-level decomposition $\widehat{\mathbb{X}}_\bullet = \mathbb{X}_\bullet \oplus \operatorname{span}\{\widehat{\varphi}_{\bullet,z} : z \in \mathcal{N}^+_\bullet\}$.

We note that there exist two constants $K, K' \geq 1$ depending only on the initial mesh $\mathcal{T}_0$ such that

$$\#\big\{z \in \mathcal{N}^+_\bullet \,:\, |T \cap \operatorname{supp}(\widehat{\varphi}_{\bullet,z})| > 0\big\} \leq K < \infty \quad \text{for all } T \in \mathcal{T}_\bullet \tag{3.1}$$

and

$$\#\{T \in \mathcal{T}_\bullet : |T \cap \operatorname{supp}(\widehat{\varphi}_{\bullet,z})| > 0\} \leq K' < \infty \quad \text{for all } z \in \mathcal{N}^+_\bullet, \tag{3.2}$$

with $K = 3$ and $K' = 2$ for $d = 2$.

3.2. Polynomial spaces on Γ and parametric enrichment. First, we introduce the polynomial spaces on Γ. For each $m \in \mathbb{N}$, let $(P^m_n)_{n\in\mathbb{N}_0}$ denote the sequence of univariate polynomials which are orthogonal with respect to π_m such that P^m_n is a polynomial of degree $n \in \mathbb{N}_0$ with $\|P^m_n\|_{L^2_{\pi_m}(-1,1)} = 1$ and $P^m_0 \equiv 1$. For convenience, we also define $P^m_{-1} \equiv 0$ and, for each $n \in \mathbb{N}_0 \cup \{-1\}$, we denote by c^m_n the leading coefficient of P^m_n. It is well-known that $\{P^m_n : n \in \mathbb{N}_0\}$ is an orthonormal basis of $L^2_{\pi_m}(-1,1)$. Moreover, there holds the three-term recurrence formula

$$\beta^m_n P^m_{n+1}(y_m) = y_m P^m_n(y_m) - \beta^m_{n-1} P^m_{n-1}(y_m) \quad \text{for all } y_m \in [-1,1] \text{ and } n \in \mathbb{N}_0, \tag{3.3}$$

where $\beta^m_{n-1} = c^m_{n-1}/c^m_n$. With $\mathbb{N}_0^{\mathbb{N}} := \{\nu = (\nu_m)_{m\in\mathbb{N}} : \nu_m \in \mathbb{N}_0 \text{ for all } m \in \mathbb{N}\}$ and $\operatorname{supp}(\nu) := \{m \in \mathbb{N} : \nu_m \neq 0\}$, let $\mathfrak{J} := \{\nu \in \mathbb{N}_0^{\mathbb{N}} : \#\operatorname{supp}(\nu) < \infty\}$ be the set of all finitely supported multi-indices. Note that $\mathfrak{J}$ is countable. With

$$P_\nu(\mathbf{y}) := \prod_{m\in\mathbb{N}} P^m_{\nu_m}(y_m) = \prod_{m\in\operatorname{supp}(\nu)} P^m_{\nu_m}(y_m) \quad \text{for all } \nu \in \mathfrak{J} \text{ and all } \mathbf{y} \in \Gamma,$$

the set $\{P_\nu : \nu \in \mathfrak{J}\}$ is an orthonormal basis of $\mathbb{P} := L^2_\pi(\Gamma)$; see [40, Theorem 2.12].

For any $m \in \mathbb{N}$, let $\varepsilon_m \in \mathfrak{J}$ be the m-th unit sequence, i.e., $(\varepsilon_m)_i = \delta_{mi}$ for all $i \in \mathbb{N}$. A consequence of the three-term recurrence formula (3.3) is the identity

$$y_m P_\mu(\mathbf{y}) = \beta^m_{\mu_m} P_{\mu+\varepsilon_m}(\mathbf{y}) + \beta^m_{\mu_m-1} P_{\mu-\varepsilon_m}(\mathbf{y}) \quad \text{for all } \mu \in \mathfrak{J},\ \mathbf{y} \in \Gamma,\ \text{and } m \in \mathbb{N}. \tag{3.4}$$

Note that the Bochner space $\mathbb{V} = L^2_\pi(\Gamma;\mathbb{X})$ is isometrically isomorphic to $\mathbb{X} \otimes \mathbb{P}$ and each function $\boldsymbol{v} \in \mathbb{V}$ can be represented in the form

$$\boldsymbol{v}(x,\mathbf{y}) = \sum_{\nu\in\mathfrak{J}} v_\nu(x) P_\nu(\mathbf{y}) \quad \text{with unique coefficients } v_\nu \in \mathbb{X}. \tag{3.5}$$

Moreover, there holds (see, e.g., [6, Lemma 2.1])

$$B_0(\boldsymbol{v},\boldsymbol{w}) = \sum_{\nu\in\mathfrak{J}} \int_D a_0(x)\, \nabla v_\nu(x) \cdot \nabla w_\nu(x)\, \mathrm{d}x \quad \text{for all } \boldsymbol{v},\boldsymbol{w} \in \mathbb{V} \tag{3.6}$$

and, in particular,

$$\| \boldsymbol{v} \|_0^2 = \sum_{\nu\in\mathfrak{I}} \| v_\nu P_\nu \|_0^2 = \sum_{\nu\in\mathfrak{I}} \|a_0^{1/2}\nabla v_\nu\|_{L^2(D)}^2 \quad \text{for all } \boldsymbol{v}\in\mathbb{V}. \tag{3.7}$$

Let $\mathbf{0} = (0,0,\dots)$ denote the zero index, and let $\mathfrak{P}_\bullet \subset \mathfrak{I}$ be a finite index set such that $\mathbf{0} \in \mathfrak{P}_\bullet$. We denote by $\operatorname{supp}(\mathfrak{P}_\bullet) := \bigcup_{\nu\in\mathfrak{P}_\bullet} \operatorname{supp}(\nu)$ the set of active parameters in $\mathfrak{P}_\bullet$. Turning now to the parametric enrichment, we follow the same construction as in [10, 7, 6, 5]. For a fixed $\overline{M} \in \mathbb{N}$, we consider the *detail index set*

$$\mathfrak{Q}_\bullet := \{\mu \in \mathfrak{I}\setminus\mathfrak{P}_\bullet : \mu = \nu \pm \varepsilon_m \text{ for all } \nu\in\mathfrak{P}_\bullet \text{ and all } m = 1,\dots,M_{\mathfrak{P}_\bullet} + \overline{M}\}, \tag{3.8}$$

where $M_{\mathfrak{P}_\bullet} := \#\operatorname{supp}(\mathfrak{P}_\bullet) \in \mathbb{N}_0$ is the number of active parameters in the index set $\mathfrak{P}_\bullet$. Thus, for a given $\mathfrak{P}_\bullet \subset \mathfrak{I}$, the detail index set represents an "active boundary" of $\mathfrak{P}_\bullet$ that contains multi-indices having up to $M_{\mathfrak{P}_\bullet} + \overline{M}$ active parameters. Then, a parametric enrichment is obtained by adding some marked indices $\mathfrak{M}_\bullet \subseteq \mathfrak{Q}_\bullet$ to the current index set $\mathfrak{P}_\bullet$, i.e., $\mathfrak{P}_\bullet \subseteq \mathfrak{P}_\circ := \mathfrak{P}_\bullet \cup \mathfrak{M}_\bullet \subseteq \mathfrak{P}_\bullet \cup \mathfrak{Q}_\bullet$.

3.3. Multilevel approximation spaces. For each index $\nu \in \mathfrak{P}_\bullet$, let $\mathcal{T}_{\bullet\nu} \in \operatorname{refine}(\mathcal{T}_0)$ be a mesh and $\mathbb{X}_{\bullet\nu} := \mathcal{S}_0^1(\mathcal{T}_{\bullet\nu})$ be the corresponding finite element space. Furthermore, for all indices $\nu \in \mathfrak{I}\setminus\mathfrak{P}_\bullet$, we set $\mathcal{T}_{\bullet\nu} := \mathcal{T}_0$. Following [19], our discretization of (2.8) is based on the finite-dimensional subspace

$$\mathbb{V}_\bullet := \bigoplus_{\nu\in\mathfrak{P}_\bullet} \mathbb{V}_{\bullet\nu} \subset \mathbb{V} \quad \text{with} \quad \mathbb{V}_{\bullet\nu} := \mathbb{X}_{\bullet\nu} \otimes \operatorname{span}\{P_\nu\} = \operatorname{span}\big\{\varphi_{\bullet\nu,z}P_\nu \,:\, z\in\mathcal{N}_{\bullet\nu}\big\}. \tag{3.9}$$

Note that the sum of the spaces $\mathbb{V}_{\bullet\nu}$ in (3.9) is orthogonal and hence direct. We emphasize that, in contrast to [20, 6, 5], where $\mathbb{X}_{\bullet\nu} = \mathbb{X}_{\bullet\mu} =: \mathbb{X}_\bullet$ for all $\nu,\mu\in\mathfrak{P}_\bullet$ and, hence, $\mathbb{V}_\bullet = \mathbb{X}_\bullet \otimes \operatorname{span}\big\{P_\nu \,:\, \nu\in\mathfrak{P}_\bullet\big\}$ has the tensor product structure (the so-called *single-level* approximation space), the approximation space $\mathbb{V}_\bullet$ defined in (3.9) has a *multilevel* structure that allows $\mathbb{X}_{\bullet\nu} \neq \mathbb{X}_{\bullet\mu}$ for $\mu\neq\nu$. Furthermore, while each mesh $\mathcal{T}_{\bullet\nu}$ ($\nu\in\mathfrak{P}_\bullet$) is obtained by a local refinement of the same coarse mesh $\mathcal{T}_0$, any two meshes $\mathcal{T}_{\bullet\nu}$, $\mathcal{T}_{\bullet\mu}$ ($\nu,\mu\in\mathfrak{P}_\bullet$) are not necessarily nested. This is a more general construction than that considered in [16], where the meshes $\mathcal{T}_{\bullet\nu}$, $\mathcal{T}_{\bullet\mu}$ ($\nu\neq\mu$) were assumed to be nested.

The Galerkin discretization of (2.8) reads as follows: Find $\boldsymbol{u}_\bullet \in \mathbb{V}_\bullet$ such that

$$B(\boldsymbol{u}_\bullet, \boldsymbol{v}_\bullet) = F(\boldsymbol{v}_\bullet) \quad \text{for all } \boldsymbol{v}_\bullet \in \mathbb{V}_\bullet. \tag{3.10}$$

Again, the Riesz theorem proves the existence and uniqueness of the solution $\boldsymbol{u}_\bullet \in \mathbb{V}_\bullet$. Moreover, the mapping $\mathbb{V} \ni \boldsymbol{u} \mapsto \boldsymbol{u}_\bullet \in \mathbb{V}_\bullet$ is the orthogonal projection onto $\mathbb{V}_\bullet$ with respect to the bilinear form $B(\cdot,\cdot)$. Therefore, there holds the best approximation property

$$\| \boldsymbol{u} - \boldsymbol{u}_\bullet \| = \min_{\boldsymbol{v}_\bullet\in\mathbb{V}_\bullet} \| \boldsymbol{u} - \boldsymbol{v}_\bullet \|.$$

4. A posteriori error estimation.

4.1. Saturation assumption. Given a multilevel subspace $\mathbb{V}_\bullet$ from (3.9), we adopt the approach of [10, Remark 4.3] and consider an enriched subspace $\widehat{\mathbb{V}}_\bullet \supseteq \mathbb{V}_\bullet$ defined as

$$\widehat{\mathbb{V}}_\bullet := \bigoplus_{\nu\in\mathfrak{P}_\bullet} \big[\widehat{\mathbb{X}}_{\bullet\nu} \otimes \operatorname{span}\{P_\nu\}\big] \oplus \bigoplus_{\nu\in\mathfrak{Q}_\bullet} \big[\mathbb{X}_0 \otimes \operatorname{span}\{P_\nu\}\big] \subset \mathbb{V}, \tag{4.1}$$

where we recall that $\mathcal{T}_{\bullet\nu} = \mathcal{T}_0$ for all $\nu \in \mathfrak{Q}_\bullet \subset \mathfrak{I}\backslash\mathfrak{P}_\bullet$. Note that $\mathbb{V}_\bullet \subseteq \mathbb{V}_\circ \subseteq \widehat{\mathbb{V}}_\bullet$, where $\mathbb{V}_\circ$ is obtained from $\mathbb{V}_\bullet$ by *one step* of (adaptive) refinement/enrichment, i.e., $\mathbb{V}_\circ$ is represented in the form (3.9) with

$$\mathfrak{P}_\circ = \mathfrak{P}_\bullet \cup \mathfrak{M}_\bullet \subseteq \mathfrak{P}_\bullet \cup \mathfrak{Q}_\bullet, \tag{4.2a}$$

$$\mathcal{T}_{\circ\nu} = \operatorname{refine}(\mathcal{T}_{\bullet\nu}, \mathcal{M}_{\bullet\nu}) \text{ for all } \nu \in \mathfrak{P}_\bullet \text{ and } \mathcal{T}_{\circ\nu} = \mathcal{T}_0 \text{ for all } \nu \in \mathfrak{P}_\circ\backslash\mathfrak{P}_\bullet. \tag{4.2b}$$

Let $\widehat{\boldsymbol{u}}_\bullet \in \widehat{\mathbb{V}}_\bullet$ be the unique Galerkin solution to

$$B(\widehat{\boldsymbol{u}}_\bullet, \widehat{\boldsymbol{v}}_\bullet) = F(\widehat{\boldsymbol{v}}_\bullet) \quad \text{for all } \widehat{\boldsymbol{v}}_\bullet \in \widehat{\mathbb{V}}_\bullet. \tag{4.3}$$

Existence and uniqueness of the solution $\widehat{\boldsymbol{u}}_\bullet \in \widehat{\mathbb{V}}_\bullet$ follow from the Riesz theorem. We emphasize that $\widehat{\boldsymbol{u}}_\bullet \in \widehat{\mathbb{V}}_\bullet$ is only needed for analysis and will not be computed throughout.

We suppose that there exists a uniform constant $0 < q_{\mathrm{sat}} < 1$ such that the following saturation assumption holds:

$$|\!|\!| \boldsymbol{u} - \widehat{\boldsymbol{u}}_\bullet |\!|\!| \le q_{\mathrm{sat}} \, |\!|\!| \boldsymbol{u} - \boldsymbol{u}_\bullet |\!|\!|. \tag{4.4}$$

We recall the orthogonal decomposition

$$|\!|\!| \boldsymbol{u} - \widehat{\boldsymbol{u}}_\bullet |\!|\!|^2 + |\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|^2 = |\!|\!| \boldsymbol{u} - \boldsymbol{u}_\bullet |\!|\!|^2.$$

Elementary calculation thus proves that the saturation assumption (4.4) is equivalent to

$$|\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|^2 \le |\!|\!| \boldsymbol{u} - \boldsymbol{u}_\bullet |\!|\!|^2 \le \frac{1}{1-q_{\mathrm{sat}}^2} \, |\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|^2, \tag{4.5}$$

i.e., the Galerkin error $|\!|\!| \boldsymbol{u}-\boldsymbol{u}_\bullet |\!|\!|$ of the (computed) coarse-space solution $\boldsymbol{u}_\bullet \in \mathbb{V}_\bullet$ is equivalent to the error reduction $|\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|$ with respect to the (non-computed) fine-space solution $\widehat{\boldsymbol{u}}_\bullet \in \widehat{\mathbb{V}}_\bullet$.

Remark 1. The saturation assumption (4.4) is a strong restriction (which may even fail in general [12]), if required for all discrete subspaces $\mathbb{V}_\bullet$. In practice, however, it is only required for the sequence of nested discrete subspaces generated by an adaptive solution process.

4.2. Error estimator and main result. The error in multilevel stochastic Galerkin approximations has two principal components: the *parametric* error arising from the choice of the index set $\mathfrak{P}_\bullet$ and the *spatial* error due to finite element discretizations for each $\nu \in \mathfrak{P}_\bullet$. We estimate the contributions to the error from each of these two components separately. To abbreviate notation, let $\langle w \,,\, v\rangle_D := \int_D a_0 \nabla w \cdot \nabla v \mathrm{d}x$ be the energy scalar product on the space $\mathbb{X} = H^1_0(D)$ in the physical domain and let $\|\cdot\|_D := \|a_0^{1/2}\nabla(\cdot)\|_{L^2(D)}$ be the induced energy norm on $\mathbb{X}$.

The *parametric* error is estimated by means of hierarchical error indicators (cf. [4, 10])

$$\tau_\bullet(\nu) := \|e_{\bullet\nu}\|_D \quad \text{for all } \nu \in \mathfrak{Q}_\bullet, \tag{4.6a}$$

where $e_{\bullet\nu} \in \mathbb{X}_0$ is the unique solution of

$$\langle e_{\bullet\nu}\,,\, v_0\rangle_D = F(v_0 P_\nu) - B(\boldsymbol{u}_\bullet, v_0 P_\nu) \quad \text{for all } v_0 \in \mathbb{X}_0. \tag{4.6b}$$

In order to estimate the errors due to *spatial* discretizations, we employ the two-level error estimation strategy, which has been analyzed in [6] for single-level approximation spaces. Specifically, we define the two-level error indicators

$$\tau_\bullet(\nu, z) := \frac{|F(\widehat{\varphi}_{\bullet\nu,z} P_\nu) - B(\boldsymbol{u}_\bullet, \widehat{\varphi}_{\bullet\nu,z} P_\nu)|}{\|\widehat{\varphi}_{\bullet\nu,z}\|_D} \quad \text{for all } \nu \in \mathfrak{P}_\bullet \text{ and all } z \in \mathcal{N}_{\bullet\nu}^+. \tag{4.7}$$

Overall, we thus consider the computable *a posteriori* error estimate

$$\tau_\bullet := \Big(\sum_{\nu\in\mathfrak{P}_\bullet} \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \tau_\bullet(\nu,z)^2 + \sum_{\nu\in\mathfrak{Q}_\bullet} \tau_\bullet(\nu)^2 \Big)^{1/2}. \tag{4.8}$$

The following theorem is the main theoretical result of this work.

Theorem 2. *Let $\mathbb{V}_\bullet$ be a given multilevel approximation space (3.9), and let $\widehat{\mathbb{V}}_\bullet$ be the enriched space as defined in (4.1). Then, for two Galerkin approximations $\boldsymbol{u}_\bullet \in \mathbb{V}_\bullet$ and $\widehat{\boldsymbol{u}}_\bullet \in \widehat{\mathbb{V}}_\bullet$ satisfying (3.10) and (4.3), respectively, there holds*

$$C_{\rm est}^{-1} \,|\!|\!|\, \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet \,|\!|\!|\, \le \tau_\bullet \overset{(4.8)}{=} \Big(\sum_{\nu\in\mathfrak{P}_\bullet} \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \tau_\bullet(\nu,z)^2 + \sum_{\nu\in\mathfrak{Q}_\bullet} \tau_\bullet(\nu)^2 \Big)^{1/2} \le C_{\rm est} \,|\!|\!|\, \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet \,|\!|\!|\,. \tag{4.9}$$

Furthermore, if $\boldsymbol{u} \in \mathbb{V}$ is the solution to problem (2.8), then, under the saturation assumption (4.4), the estimates (4.9) are equivalent to

$$\frac{(1-q_{\rm sat}^2)^{1/2}}{C_{\rm est}} \,|\!|\!|\, \boldsymbol{u} - \boldsymbol{u}_\bullet \,|\!|\!|\, \le \tau_\bullet \le C_{\rm est} \,|\!|\!|\, \boldsymbol{u} - \boldsymbol{u}_\bullet \,|\!|\!|\,, \tag{4.10}$$

i.e., the proposed error estimator is reliable (under the saturation assumption) and (always) efficient. The constant $C_{\rm est} \ge 1$ in (4.9)–(4.10) is generic and depends only on uniform shape regularity of the refinements of $\mathcal{T}_0$, the mean field a_0, and the constant $\tau > 0$ from (2.4).

4.3. Auxiliary results in deterministic setting. Throughout this section, we denote by $\mathcal{T}_\star \in \mathrm{refine}(\mathcal{T}_0)$ an arbitrary refinement of the initial mesh. Recall that $\mathbb{X} = H_0^1(D)$ and $\mathbb{X}_\star := \mathcal{S}_0^1(\mathcal{T}_\star)$. The proof of Theorem 2 will employ the (spatial) orthogonal projections

$$\mathbb{G}_\star : \mathbb{X} \to \mathbb{X}_\star \quad \text{and} \quad \widehat{\mathbb{G}}_{\star,z} : \mathbb{X} \to \widehat{\mathbb{X}}_{\star,z} := \mathrm{span}\{\widehat{\varphi}_{\star,z}\} \quad \text{for } z \in \mathcal{N}_\star^+$$

defined by

$$\langle \mathbb{G}_\star w\,,\, v_\star\rangle_D = \langle w\,,\, v_\star\rangle_D \qquad \text{for all } v_\star \in \mathbb{X}_\star, \tag{4.11}$$

$$\langle \widehat{\mathbb{G}}_{\star,z} w\,,\, \widehat{v}_{\star,z}\rangle_D = \langle w\,,\, \widehat{v}_{\star,z}\rangle_D \qquad \text{for all } \widehat{v}_{\star,z} \in \widehat{\mathbb{X}}_{\star,z}. \tag{4.12}$$

First, we recall the norm equivalence from [6, Proof of Lemma 3.4, Steps 1–2].

Lemma 3. *For all $z \in \mathcal{N}_\star^+$, let $\widehat{w}_{\star,z} \in \operatorname{span}\{\widehat{\varphi}_{\star,z}\}$. Then, there holds*

$$K^{-1} \Big\| \sum_{z\in\mathcal{N}_\star^+} \widehat{w}_{\star,z} \Big\|_D^2 \le \sum_{z\in\mathcal{N}_\star^+} \|\widehat{w}_{\star,z}\|_D^2 \le C_{\mathrm{loc}} \Big\| \sum_{z\in\mathcal{N}_\star^+} \widehat{w}_{\star,z} \Big\|_D^2. \tag{4.13}$$

Here, $C_{\mathrm{loc}} > 0$ depends only on the shape regularity of $\widehat{\mathcal{T}}_\star$ and the mean field a_0, whereas $K > 0$ is the constant from (3.1).

Second, we recall that nodal interpolation is stable on finite-dimensional subspaces; see [6, Proof of Lemma 3.5, Step 1].

Lemma 4. *For $\widehat{v}_\star \in \widehat{\mathbb{X}}_\star$, let $v_\star := \sum_{z\in\mathcal{N}_\star} \widehat{v}_\star(z)\varphi_{\star,z}$ be the nodal interpolation onto $\mathbb{X}_\star$. Then*

$$\widehat{v}_\star - v_\star = \sum_{z\in\mathcal{N}_\star^+} \widehat{w}_{\star,z} \quad \text{with} \quad \widehat{w}_{\star,z} \in \operatorname{span}\{\widehat{\varphi}_{\star,z}\} \tag{4.14}$$

and there holds

$$\|\widehat{v}_\star - v_\star\|_D \le C_{\mathrm{stb}} \|\widehat{v}_\star\|_D, \tag{4.15}$$

where $C_{\mathrm{stb}} > 0$ depends only on the shape regularity of $\widehat{\mathcal{T}}_\star$ and the mean field a_0.

4.4. Proof of Theorem 2. Recall the orthogonal projectors $\mathbb{G}_\star : \mathbb{X} \to \mathbb{X}_\star$ and $\widehat{\mathbb{G}}_{\star,z} : \mathbb{X} \to \widehat{\mathbb{X}}_{\star,z}$ defined in (4.11) and (4.12), respectively. The following lemma provides the key argument for the proof of Theorem 2.

Lemma 5. *For any $\widehat{\boldsymbol{v}}_\bullet = \sum_{\nu\in\mathfrak{P}_\bullet} \widehat{v}_{\bullet\nu}P_\nu + \sum_{\nu\in\mathfrak{Q}_\bullet} v_{\bullet\nu}P_\nu \in \widehat{\mathbb{V}}_\bullet$, where $\widehat{v}_{\bullet\nu} \in \widehat{\mathbb{X}}_{\bullet\nu}$ for $\nu \in \mathfrak{P}_\bullet$ and $v_{\bullet\nu} \in \mathbb{X}_{\bullet\nu} = \mathbb{X}_0$ for $\nu \in \mathfrak{Q}_\bullet$, the following estimates hold*

$$C_Y^{-1} \|\!|\, \widehat{\boldsymbol{v}}_\bullet \,|\!\|_0^2 \le \sum_{\nu\in\mathfrak{P}_\bullet} \Big(\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\|_D^2 + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\|_D^2 \Big) + \sum_{\nu\in\mathfrak{Q}_\bullet} \|v_{\bullet\nu}\|_D^2 \le 2K \|\!|\, \widehat{\boldsymbol{v}}_\bullet \,|\!\|_0^2. \tag{4.16}$$

Here, $C_Y \ge 1$ depends only on the shape regularity of $\widehat{\mathcal{T}}$ and the mean field a_0, whereas $K > 0$ is the constant from (3.1). Moreover, the upper bound holds with the constant K (instead of $2K$) if $\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu} = 0$ for all $\nu \in \mathfrak{P}_\bullet$.

Proof. Using (3.7), we have

$$\|\!|\, \widehat{\boldsymbol{v}}_\bullet \,|\!\|_0^2 = \sum_{\nu\in\mathfrak{P}_\bullet} \|\widehat{v}_{\bullet\nu}\|_D^2 + \sum_{\nu\in\mathfrak{Q}_\bullet} \|v_{\bullet\nu}\|_D^2. \tag{4.17}$$

For all $\nu \in \mathfrak{P}_\bullet$, we apply Lemma 4 to $\widehat{v}_{\bullet\nu} \in \widehat{\mathbb{X}}_{\bullet\nu}$ in order to find $v_{\bullet\nu} \in \mathbb{X}_{\bullet\nu}$ and $\widehat{w}_{\bullet\nu,z} \in \operatorname{span}\{\widehat{\varphi}_{\bullet\nu,z}\}$ for all $z \in \mathcal{N}_{\bullet\nu}^+$ such that (4.14)–(4.15) hold.

Step 1. First, we prove the lower bound in (4.16). The Cauchy inequality yields that

$$\begin{aligned} \|\widehat{v}_{\bullet\nu}\|_D^2 &\overset{(4.14)}{=} \Big\langle \widehat{v}_{\bullet\nu}\,,\, v_{\bullet\nu} + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \widehat{w}_{\bullet\nu,z} \Big\rangle_D = \langle \mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\,,\, v_{\bullet\nu}\rangle_D + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \langle \widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\,,\, \widehat{w}_{\bullet\nu,z}\rangle_D \\ &\le \Big(\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\|_D^2 + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\|_D^2 \Big)^{1/2} \Big(\|v_{\bullet\nu}\|_D^2 + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{w}_{\bullet\nu,z}\|_D^2 \Big)^{1/2}. \end{aligned}$$

Stability (4.15) shows that

$$\|v_{\bullet\nu}\|_D \le \|\widehat{v}_{\bullet\nu}\|_D + \|\widehat{v}_{\bullet\nu} - v_{\bullet\nu}\|_D \overset{(4.15)}{\le} (1 + C_{\mathrm{stb}})\|\widehat{v}_{\bullet\nu}\|_D.$$

The upper bound in (4.13) proves that

$$\sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{w}_{\bullet\nu,z}\|_D^2 \le C_{\mathrm{loc}} \Big\| \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \widehat{w}_{\bullet\nu,z}\Big\|_D^2 = C_{\mathrm{loc}}\, \|\widehat{v}_{\bullet\nu} - v_{\bullet\nu}\|_D^2 \overset{(4.15)}{\le} C_{\mathrm{loc}} C_{\mathrm{stb}}^2\, \|\widehat{v}_{\bullet\nu}\|_D^2.$$

Combining the latter three estimates, we conclude that

$$\|\widehat{v}_{\bullet\nu}\|_D \le \big[(1 + C_{\mathrm{stb}})^2 + C_{\mathrm{loc}} C_{\mathrm{stb}}^2\big]^{1/2} \Big(\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\|_D^2 + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\|_D^2\Big)^{1/2}.$$

Using this estimate together with (4.17), we prove the lower bound in (4.16) with $C_Y = (1 + C_{\mathrm{stb}})^2 + C_{\mathrm{loc}} C_{\mathrm{stb}}^2 \ge 1$.

Step 2. To prove the upper bound in (4.16), we proceed analogously. For all $\nu \in \mathfrak{P}_\bullet$, there holds

$$\begin{aligned}\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\|_D^2 + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\|_D^2 &= \langle \mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\,,\, \widehat{v}_{\bullet\nu}\rangle_D + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \langle \widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\,,\, \widehat{v}_{\bullet\nu}\rangle_D \\ &= \Big\langle \mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu} + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\,,\, \widehat{v}_{\bullet\nu}\Big\rangle_D \le \Big\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu} + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\Big\|_D \|\widehat{v}_{\bullet\nu}\|_D.\end{aligned}$$

Using the lower bound in (4.13) and the fact that $K \ge 1$, we prove that

$$\begin{aligned}\Big\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu} + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\Big\|_D^2 &\le 2\Big(\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\|_D^2 + \Big\| \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\Big\|_D^2\Big) \\ &\le 2K\Big(\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\|_D^2 + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\|_D^2\Big).\end{aligned}$$

The latter two estimates imply that

$$\Big(\|\mathbb{G}_{\bullet\nu}\widehat{v}_{\bullet\nu}\|_D^2 + \sum_{z\in\mathcal{N}_{\bullet\nu}^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{v}_{\bullet\nu}\|_D^2\Big)^{1/2} \le \sqrt{2K}\, \|\widehat{v}_{\bullet\nu}\|_D.$$

By substituting this estimate into (4.17), we conclude the proof. ∎

Proof of Theorem 2. The equivalence of estimates (4.9) and (4.10) is an immediate consequence of (4.5). Therefore, it only remains to prove (4.9). The proof consists of three steps.

Step 1. Define $\widehat{e}_\bullet := \sum_{\nu\in\mathfrak{P}_\bullet} \widehat{e}_{\bullet\nu}P_\nu + \sum_{\nu\in\mathfrak{Q}_\bullet} e_{\bullet\nu}P_\nu \in \widehat{\mathbb{V}}_\bullet$, where $e_{\bullet\nu} \in \mathbb{X}_0 = \mathbb{X}_{\bullet\nu}$ for $\nu \in \mathfrak{Q}_\bullet$ is given by (4.6), while $\widehat{e}_{\bullet\nu} \in \widehat{\mathbb{X}}_{\bullet\nu}$ for $\nu \in \mathfrak{P}_\bullet$ is the unique solution to

$$\langle \widehat{e}_{\bullet\nu}\,,\, \widehat{v}_{\bullet\nu}\rangle_D = B(\widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet, \widehat{v}_{\bullet\nu}P_\nu) \quad \text{for all } \widehat{v}_{\bullet\nu} \in \widehat{\mathbb{X}}_{\bullet\nu}. \tag{4.18}$$

For all $\nu \in \mathfrak{P}_\bullet$, Galerkin orthogonality implies that

$$\langle \widehat{e}_{\bullet\nu}\,,\, v_{\bullet\nu}\rangle_D = B(\widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet, v_{\bullet\nu}P_\nu) = 0 \quad \text{for all } v_{\bullet\nu} \in \mathbb{X}_{\bullet\nu}.$$

Hence, we see that $\mathbb{G}_{\bullet\nu}\widehat{e}_{\bullet\nu} = 0$ for all $\nu \in \mathfrak{P}_\bullet$. In conclusion, Lemma 5 yields that

$$|\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!|_0^2 \simeq \sum_{\nu\in\mathfrak{P}_\bullet}\sum_{z\in\mathcal{N}_\nu^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{e}_{\bullet\nu}\|_D^2 + \sum_{\nu\in\mathfrak{Q}_\bullet} \|e_{\bullet\nu}\|_D^2 \overset{(4.6a)}{=} \sum_{\nu\in\mathfrak{P}_\bullet}\sum_{z\in\mathcal{N}_\nu^+} \|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{e}_{\bullet\nu}\|_D^2 + \sum_{\nu\in\mathfrak{Q}_\bullet} \tau_\bullet(\nu)^2.$$

Step 2. The orthogonal projection onto the one-dimensional space $\operatorname{span}\{\widehat{\varphi}_{\bullet\nu,z}\}$ takes the explicit form

$$\widehat{\mathbb{G}}_{\bullet\nu,z} v = \frac{\langle v\,,\,\widehat{\varphi}_{\bullet\nu,z}\rangle_D}{\|\widehat{\varphi}_{\bullet\nu,z}\|_D^2}\,\widehat{\varphi}_{\bullet\nu,z} \quad \text{for any } v \in \mathbb{X}.$$

Hence, for all $\nu \in \mathfrak{P}_\bullet$ and for each $z \in \mathcal{N}_\nu^+$, there holds

$$\|\widehat{\mathbb{G}}_{\bullet\nu,z}\widehat{e}_{\bullet\nu}\|_D = \frac{|\langle \widehat{e}_{\bullet\nu}\,,\,\widehat{\varphi}_{\bullet\nu,z}\rangle_D|}{\|\widehat{\varphi}_{\bullet\nu,z}\|_D} = \frac{|B(\widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet, \widehat{\varphi}_{\bullet\nu,z}P_\nu)|}{\|\widehat{\varphi}_{\bullet\nu,z}\|_D} \overset{(4.7)}{=} \tau_\bullet(\nu,z).$$

This leads to the equivalence

$$|\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!|_0^2 \simeq \sum_{\nu\in\mathfrak{P}_\bullet}\sum_{z\in\mathcal{N}_\nu^+} \tau_\bullet(\nu,z)^2 + \sum_{\nu\in\mathfrak{Q}_\bullet} \tau_\bullet(\nu)^2 = \tau_\bullet^2,$$

where the hidden constants depend only on uniform shape regularity of the meshes $\mathcal{T}_\star \in \operatorname{refine}(\mathcal{T}_0)$, the (local) mesh-refinement rule, and the mean field a_0.

Step 3. It remains to prove the equivalence $|\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!|_0 \simeq |\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|$. To that end, we note that the variational formulation (4.18) implies that

$$B_0(\widehat{\boldsymbol{e}}_\bullet, \widehat{\boldsymbol{v}}_\bullet) = B(\widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet, \widehat{\boldsymbol{v}}_\bullet) \quad \text{for all } \widehat{\boldsymbol{v}}_\bullet \in \widehat{\mathbb{V}}_\bullet.$$

Hence, using norm equivalence (2.7), we obtain that

$$|\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!|_0^2 = B(\widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet, \widehat{\boldsymbol{e}}_\bullet) \le |\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|\, |\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!| \le \Lambda^{1/2}\, |\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|\, |\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!|_0$$

and

$$|\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|^2 = B_0(\widehat{\boldsymbol{e}}_\bullet, \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet) \le |\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!|_0 |\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|_0 \le \lambda^{-1/2}\, |\!|\!| \widehat{\boldsymbol{e}}_\bullet |\!|\!|_0 |\!|\!| \widehat{\boldsymbol{u}}_\bullet - \boldsymbol{u}_\bullet |\!|\!|.$$

This concludes the proof. ∎

Remark 6. Let $\mathbb{V}_\circ$ be a multilevel approximation space that is obtained from $\mathbb{V}_\bullet$ by one step of (adaptive) refinement/enrichment (see (4.2)) such that $\mathbb{V}_\bullet \subseteq \mathbb{V}_\circ \subseteq \widehat{\mathbb{V}}_\bullet$. For $d = 2$, newest vertex bisection ensures that $\widehat{\varphi}_{\bullet\nu,z} = \varphi_{\circ\nu,z}$ for all $\nu \in \mathfrak{P}_\bullet$ and for each $z \in \mathcal{N}_\nu^+ \cap \mathcal{N}_{\circ\nu}$. If $\boldsymbol{u}_\bullet \in \mathbb{V}_\bullet$ and $\boldsymbol{u}_\circ \in \mathbb{V}_\circ$ are two Galerkin approximations, then by arguing as in the proof of

Theorem 2, we obtain that

$$C_{\mathrm{est}}^{-1}\,|\!|\!|\, \boldsymbol{u}_\circ - \boldsymbol{u}_\bullet \,|\!|\!|\, \leq \Bigg(\sum_{\nu\in\mathfrak{P}_\bullet} \sum_{z\in\mathcal{N}_\nu^+\cap\mathcal{N}_{\circ\nu}} \tau_\bullet(\nu,z)^2 + \sum_{\nu\in\mathfrak{Q}_\bullet\cap\mathfrak{P}_\circ} \tau_\bullet(\nu)^2 \Bigg)^{1/2} \leq C_{\mathrm{est}}\,|\!|\!|\, \boldsymbol{u}_\circ - \boldsymbol{u}_\bullet \,|\!|\!|\,. \tag{4.19}$$

Therefore, in this setting (at least in 2D), the two-level estimator allows to control the error reduction due to adaptive enrichment of the multilevel approximation space $\mathbb{V}_\bullet$.

5. Adaptive algorithms.

In this section, we present adaptive algorithms with three different Dörfler-type marking criteria (and hence, different refinement strategies). These algorithms generate sequences of successively enriched multilevel approximation spaces, as well as the corresponding Galerkin approximations and error estimates.

We consider the following standard adaptive loop

$$\mathsf{SOLVE} \longrightarrow \mathsf{ESTIMATE} \longrightarrow \mathsf{MARK} \longrightarrow \mathsf{REFINE},$$

where the precise marking strategy is to be specified in the subsections below.

Algorithm 7. ***Input:*** *$\mathfrak{P}_0 = \{\mathbf{0}\}$ and $\mathcal{T}_{0\nu} := \mathcal{T}_0$ for all $\nu \in \mathfrak{P}_0 \cup \mathfrak{Q}_0$; marking criterion. Set the counter $\ell := 0$.*

(i) *Compute the discrete solution $\boldsymbol{u}_\ell \in \mathbb{V}_\ell$ by solving (3.10).*
(ii) *Compute spatial error indicators $\tau_\ell(\nu,z)$ from (4.7) for all $\nu \in \mathfrak{P}_\ell$ and all $z \in \mathcal{N}_{\ell\nu}^+$.*
(iii) *Compute parametric error indicators $\tau_\ell(\nu)$ from (4.6) for all $\nu \in \mathfrak{Q}_\ell$.*
(iv) *Use marking criterion to determine $\mathcal{M}_{\ell\nu} \subseteq \mathcal{N}_{\ell\nu}^+$ for all $\nu \in \mathfrak{P}_\ell$ and $\mathfrak{M}_\ell \subseteq \mathfrak{Q}_\ell$.*
(v) *For all $\nu \in \mathfrak{P}_\ell$, set $\mathcal{T}_{(\ell+1)\nu} := \mathrm{refine}(\mathcal{T}_{\ell\nu}, \mathcal{M}_{\ell\nu})$.*
(vi) *Set $\mathfrak{P}_{\ell+1} := \mathfrak{P}_\ell \cup \mathfrak{M}_\ell$ and $\mathcal{T}_{(\ell+1)\nu} := \mathcal{T}_0$ for all $\nu \in \mathfrak{Q}_{\ell+1}$.*
(vii) *Increase the counter $\ell \mapsto \ell+1$ and goto* (i).

Output: *For all $\ell \in \mathbb{N}_0$, the algorithm returns the multilevel stochastic Galerkin approximation $\boldsymbol{u}_\ell \in \mathbb{V}_\ell$ as well as the corresponding error estimate τ_ℓ.*

5.1. Separate spatial and parametric marking/enrichment.

The two marking criteria presented below follow the same approach as utilized in [4, 10, 7, 5] in the case of single-level stochastic Galerkin FEM. Under this approach, *either* a spatial refinement *or* a parametric enrichment is performed at each iteration. The choice between the two is made by comparing the respective contributions to the total error estimate $\tau_\bullet$ given by (4.8) (Criterion A) or by comparing the associated error reduction indicators (Criterion B; cf. Remark 6).

Criterion A. ***Input:*** *error indicators $\{\tau_\ell(\nu,z) : \nu \in \mathfrak{P}_\ell,\ z \in \mathcal{N}_{\ell\nu}^+\}$ and $\{\tau_\ell(\nu) : \nu \in \mathfrak{Q}_\ell\}$; marking parameters $0 < \theta_{\mathbb{X}}, \theta_{\mathfrak{P}} \leq 1$, and $\vartheta > 0$.*

- *If $\vartheta \sum_{\nu\in\mathfrak{Q}_\ell} \tau_\ell(\nu)^2 \leq \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\mathcal{N}_{\ell\nu}^+} \tau_\ell(\nu,z)^2$, then proceed as follows:*
  - *Set $\mathfrak{M}_\ell := \emptyset$.*
  - *Determine $\mathcal{M}_{\ell\nu} \subseteq \mathcal{N}_{\ell\nu}^+$ for all $\nu \in \mathfrak{P}_\ell$ such that*

$$\theta_{\mathbb{X}} \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\mathcal{N}_{\ell\nu}^+} \tau_\ell(\nu,z)^2 \leq \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\mathcal{M}_{\ell\nu}} \tau_\ell(\nu,z)^2, \tag{5.1}$$

where the cumulative cardinality $\sum_{\nu\in\mathfrak{P}_\ell} \#\mathcal{M}_{\ell\nu}$ is minimal (amongst all sets which satisfy (5.1)).

- *Otherwise, if* $\vartheta \sum_{\nu\in\mathfrak{Q}_\ell} \tau_\ell(\nu)^2 > \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\mathcal{N}_{\ell\nu}^+} \tau_\ell(\nu,z)^2$, *then proceed as follows:*
  - *Set* $\mathcal{M}_{\ell\nu} := \emptyset$ *for all* $\nu \in \mathfrak{P}_\ell$.
  - *Determine* $\mathfrak{M}_\ell \subseteq \mathfrak{Q}_\ell$ *such that*

$$\theta_{\mathfrak{P}} \sum_{\nu\in\mathfrak{Q}_\ell} \tau_\ell(\nu)^2 \le \sum_{\nu\in\mathfrak{M}_\ell} \tau_\ell(\nu)^2, \tag{5.2}$$

  where the cardinality $\#\mathfrak{M}_\ell$ *is minimal (amongst all sets which satisfy* (5.2)*).*

Output: $\mathcal{M}_{\ell\nu} \subseteq \mathcal{N}_{\ell\nu}^+$ *for all* $\nu \in \mathfrak{P}_\ell$ *and* $\mathfrak{M}_\ell \subseteq \mathfrak{Q}_\ell$.

Criterion B. ***Input:*** *error indicators* $\{\tau_\ell(\nu,z) : \nu \in \mathfrak{P}_\ell,\ z \in \mathcal{N}_{\ell\nu}^+\}$ *and* $\{\tau_\ell(\nu) : \nu \in \mathfrak{Q}_\ell\}$*; marking parameters* $0 < \theta_{\mathbb{X}}, \theta_{\mathfrak{P}} \le 1$, *and* $\vartheta > 0$.

- *Determine* $\widetilde{\mathcal{M}}_{\ell\nu} \subseteq \mathcal{N}_{\ell\nu}^+$ *for all* $\nu \in \mathfrak{P}_\ell$ *such that*

$$\theta_{\mathbb{X}} \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\mathcal{N}_{\ell\nu}^+} \tau_\ell(\nu,z)^2 \le \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\widetilde{\mathcal{M}}_{\ell\nu}} \tau_\ell(\nu,z)^2, \tag{5.3}$$

  where the cumulative cardinality $\sum_{\nu\in\mathfrak{P}_\ell} \#\widetilde{\mathcal{M}}_{\ell\nu}$ *is minimal (amongst all sets which satisfy* (5.3)*).*
- *Define* $\widetilde{\mathcal{R}}_{\ell\nu} := \mathcal{N}_{\ell\nu}^+ \cap \widetilde{\mathcal{N}}_{\ell\nu}$ *for all* $\nu \in \mathfrak{P}_\ell$, *where* $\widetilde{\mathcal{N}}_{\ell\nu}$ *is the set of vertices of* $\widetilde{\mathcal{T}}_{\ell\nu} = \operatorname{refine}(\mathcal{T}_{\ell\nu}, \widetilde{\mathcal{M}}_{\ell\nu})$.
- *Determine* $\widetilde{\mathfrak{M}}_\ell \subseteq \mathfrak{Q}_\ell$ *such that*

$$\theta_{\mathfrak{P}} \sum_{\nu\in\mathfrak{Q}_\ell} \tau_\ell(\nu)^2 \le \sum_{\nu\in\widetilde{\mathfrak{M}}_\ell} \tau_\ell(\nu)^2, \tag{5.4}$$

  where the cardinality $\#\widetilde{\mathfrak{M}}_\ell$ *is minimal (amongst all sets which satisfy* (5.4)*).*
- *If* $\vartheta \sum_{\nu\in\widetilde{\mathfrak{M}}_\ell} \tau_\ell(\nu)^2 \le \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\widetilde{\mathcal{R}}_{\ell\nu}} \tau_\ell(\nu,z)^2$, *then proceed as follows:*
  - *set* $\mathfrak{M}_\ell := \emptyset$ *and* $\mathcal{M}_{\ell\nu} := \widetilde{\mathcal{M}}_{\ell\nu}$ *for all* $\nu \in \mathfrak{P}_\ell$.
- *Otherwise, if* $\vartheta \sum_{\nu\in\widetilde{\mathfrak{M}}_\ell} \tau_\ell(\nu)^2 > \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\widetilde{\mathcal{R}}_{\ell\nu}} \tau_\ell(\nu,z)^2$, *then proceed as follows:*
  - *set* $\mathfrak{M}_\ell := \widetilde{\mathfrak{M}}_\ell$ *and* $\mathcal{M}_{\ell\nu} := \emptyset$ *for all* $\nu \in \mathfrak{P}_\ell$.

Output: $\mathcal{M}_{\ell\nu} \subseteq \mathcal{N}_{\ell\nu}^+$ *for all* $\nu \in \mathfrak{P}_\ell$ *and* $\mathfrak{M}_\ell \subseteq \mathfrak{Q}_\ell$.

5.2. Combined marking/enrichment. In the case of single-level approximation spaces (where $\mathcal{T}_{\bullet\nu} = \mathcal{T}_\bullet$ for all $\nu \in \mathfrak{P}_\bullet \cup \mathfrak{Q}_\bullet$), a combined enrichment of spatial and parametric components at each iteration of the adaptive algorithm is prohibitively expensive due to the multiplicative increase of the total number of degrees of freedom (i.e., $\dim \mathbb{V}_\bullet = (\#\mathfrak{P}_\bullet)\cdot\dim \mathcal{S}^1_0(\mathcal{T}_\bullet)$). The situation is considerably different for multilevel approximation spaces defined by (3.9), for which combined enrichment always results in *additive* increase in the total number of degrees of freedom, i.e., $\dim \mathbb{V}_\bullet = \sum_{\nu\in\mathfrak{P}_\bullet} \dim \mathcal{S}^1_0(\mathcal{T}_{\bullet\nu})$. In the context of Algorithm 7, this enrichment is steered by the Dörfler marking performed on the joint set of all spatial and parametric error indicators, as presented in the following marking criterion.

Criterion C. ***Input:*** *error indicators* $\{\tau_\ell(\nu,z) : \nu \in \mathfrak{P}_\ell,\ z \in \mathcal{N}_{\ell\nu}^+\}$ *and* $\{\tau_\ell(\nu) : \nu \in \mathfrak{Q}_\ell\}$*; marking parameter* $0 < \theta \le 1$.

- *Determine the sets* $\mathcal{M}_{\ell\nu} \subseteq \mathcal{N}_{\ell\nu}^+$ *for all* $\nu \in \mathfrak{P}_\ell$ *and the set* $\mathfrak{M}_\ell \subseteq \mathfrak{Q}_\ell$ *such that*

$$\theta \left(\sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\mathcal{N}_{\ell\nu}^+} \tau_\ell(\nu,z)^2 + \sum_{\nu\in\mathfrak{Q}_\ell} \tau_\ell(\nu)^2 \right) \leq \sum_{\nu\in\mathfrak{P}_\ell} \sum_{z\in\mathcal{M}_{\ell\nu}} \tau_\ell(\nu,z)^2 + \sum_{\nu\in\mathfrak{M}_\ell} \tau_\ell(\nu)^2, \tag{5.5}$$

where the overall cardinality $\#\mathfrak{M}_\ell + \sum_{\nu\in\mathfrak{P}_\ell} \#\mathcal{M}_{\ell\nu}$ *is minimal (amongst all sets which satisfy* (5.5)*).*

Output: $\mathcal{M}_{\ell\nu} \subseteq \mathcal{N}_{\ell\nu}^+$ *for all* $\nu \in \mathfrak{P}_\ell$ *and* $\mathfrak{M}_\ell \subseteq \mathfrak{Q}_\ell$.

In what follows, we will write, e.g., Algorithm 7.A to refer to the algorithm obtained by employing Criterion A in Step (iv) of Algorithm 7.

6. Computing multilevel stochastic Galerkin approximations: implementation aspects.

The adaptive multilevel strategies outlined in section 5 are implemented within the open-source MATLAB toolbox Stochastic T-IFISS [8]. The toolbox has been developed as an extension of the FEM software package T-IFISS [41] to compute stochastic Galerkin approximations of PDE problems with parametric or uncertain inputs. Overall, this software aims at creating an environment for testing different discretization and error estimation strategies, exploring new algorithms, as well as for replication, validation and verification of computational results (see [9] for a recent review).

In this section, we briefly discuss some implementation aspects of the multilevel stochastic Galerkin FEM. In particular, we focus on assembling components of the Galerkin matrix and solving the resulting linear system.

6.1. Matrix formulation of the multilevel stochastic Galerkin FEM.

For each $\mu \in \mathfrak{P}_\bullet$, we denote by $N_{\bullet\mu}$ the dimension of the finite element space $\mathbb{X}_{\bullet\mu} = \mathcal{S}_0^1(\mathcal{T}_{\bullet\mu})$ (i.e., $N_{\bullet\mu} = \#(\mathcal{N}_{\bullet\mu} \setminus \partial D)$). Recalling (3.9), the multilevel stochastic Galerkin approximation $\boldsymbol{u}_\bullet \in \mathbb{V}_\bullet$ can be represented as follows:

$$\boldsymbol{u}_\bullet(x,\mathbf{y}) = \sum_{\mu\in\mathfrak{P}_\bullet} \sum_{j=1}^{N_{\bullet\mu}} u_{\bullet\mu,z_j} \varphi_{\bullet\mu,z_j}(x) P_\mu(\mathbf{y}). \tag{6.1}$$

Hence, by taking test functions $\boldsymbol{v}_\bullet = \varphi_{\bullet\nu,z_i} P_\nu$ for all $\nu \in \mathfrak{P}_\bullet$ and all $i = 1, 2, \ldots, N_{\bullet\nu}$, the discrete formulation (3.10) yields a linear system $\mathbf{Au} = \mathbf{b}$ for finding the unknown coefficients $u_{\bullet\mu,z_j} \in \mathbb{R}$ in (6.1).

Since the approximation space $\mathbb{V}_\bullet$ is built from tensor products of different subspaces of $\mathbb{X} = H_0^1(D)$ and $\mathbb{P} = L_\pi^2(\Gamma)$ (see (3.9)), the matrix $\mathbf{A}$ and the vectors $\mathbf{u}$ and $\mathbf{b}$ have block structure, with individual blocks indexed by multi-indices of $\mathfrak{P}_\bullet$ as follows:

$$\mathbb{R}^{N_\bullet \times N_\bullet} \ni \mathbf{A} = (\mathbf{A}_{\nu\mu})_{\nu,\mu\in\mathfrak{P}_\bullet}, \qquad \mathbb{R}^{N_\bullet} \ni \mathbf{b} = (\mathbf{b}_\nu)_{\nu\in\mathfrak{P}_\bullet}, \qquad \mathbb{R}^{N_\bullet} \ni \mathbf{u} = (\mathbf{u}_\mu)_{\mu\in\mathfrak{P}_\bullet},$$

where $N_\bullet := \dim \mathbb{V}_\bullet = \sum_{\nu\in\mathfrak{P}_\bullet} N_{\bullet\nu}$,

$$[\mathbf{A}_{\nu\mu}]_{ij} = [\mathbf{A}_{\mu\nu}]_{ji} = B(\varphi_{\bullet\mu,z_j} P_\mu, \varphi_{\bullet\nu,z_i} P_\nu), \qquad [\mathbf{b}_\nu]_i = F(\varphi_{\bullet\nu,z_i} P_\nu), \qquad [\mathbf{u}_\mu]_j = u_{\bullet\mu,z_j}$$

for $i = 1, \dots, N_{\bullet\nu}$ and $j = 1, \dots, N_{\bullet\mu}$. Hence, recalling (2.5), (2.6), (3.6) and (2.8), we find

$$[\mathbf{A}_{\nu\mu}]_{ij} = \delta_{\nu\mu} \int_D a_0(x) \nabla\varphi_{\bullet\mu,z_j}(x) \cdot \nabla\varphi_{\bullet\nu,z_i}(x)\,\mathrm{d}x + \sum_{m=0}^{\infty} \int_\Gamma y_m P_\mu(\mathbf{y}) P_\nu(\mathbf{y})\,\mathrm{d}\pi(\mathbf{y}) \int_D a_m(x) \nabla\varphi_{\bullet\mu,z_j}(x) \cdot \nabla\varphi_{\bullet\nu,z_i}(x)\,\mathrm{d}x$$

and

$$[\mathbf{b}_\nu]_i = \int_\Gamma \int_D \boldsymbol{f}(x,\mathbf{y}) \varphi_{\bullet\nu,z_i}(x) P_\nu(\mathbf{y})\,\mathrm{d}x\,\mathrm{d}\pi(\mathbf{y}).$$

Thus, for all $\nu, \mu \in \mathfrak{P}_\bullet$, the $\nu\mu$-th block in the Galerkin matrix $\mathbf{A}$ is given by

$$\mathbf{A}_{\nu\mu} = \sum_{m=0}^{\infty} [G_m]_{\nu\mu} K_m^{\nu\mu} = \sum_{m=0}^{M} [G_m]_{\nu\mu} K_m^{\nu\mu}, \tag{6.2}$$

where, for $m \in \mathbb{N}_0$,

$$[G_m]_{\nu\mu} = \begin{cases} \delta_{\nu\mu} & \text{if } m = 0, \\ \int_\Gamma y_m P_\mu(\mathbf{y}) P_\nu(\mathbf{y})\,\mathrm{d}\pi(\mathbf{y}) \overset{(3.4)}{=} \beta_{\mu_m}^m \delta_{\mu+\varepsilon_m,\nu} + \beta_{\mu_m-1}^m \delta_{\mu-\varepsilon_m,\nu} & \text{if } m \in \mathbb{N} \end{cases} \tag{6.3}$$

and $K_m^{\nu\mu}$ are the finite element (stiffness) matrices defined by

$$[K_m^{\nu\mu}]_{ij} = \int_D a_m(x) \nabla\varphi_{\bullet\mu,z_j}(x) \cdot \nabla\varphi_{\bullet\nu,z_i}(x)\,\mathrm{d}x \tag{6.4}$$

for $i = 1, \dots, N_{\bullet\nu}$ and $j = 1, \dots, N_{\bullet\mu}$, whereas $M = \#\operatorname{supp}(\mathfrak{P}_\bullet)$ is the number of active parameters in $\mathfrak{P}_\bullet$; here, we used the fact that $G_m = 0$ for all $m \notin \operatorname{supp}(\mathfrak{P}_\bullet)$ (due to the symmetry of the measure π_m on $\Gamma_m = [-1,1]$ for all $m \in \mathbb{N}$) and implicitly assumed that $\operatorname{supp}(\mathfrak{P}_\bullet) = \{1, 2, \dots, M\}$. For a detailed study of the properties of the matrices $\{G_m\}_{m=1}^M$, we refer, e.g., to [26].

At first glance, there are $(M+1)(\#\mathfrak{P}_\bullet)^2$ stiffness matrices to compute; see (6.2). However, as discussed in [16, Section 3.1], the actual number of matrices that need to be computed is significantly less. Indeed, it follows from (6.2) that one only needs to compute the matrix $K_m^{\nu\mu}$ if the corresponding entry $[G_m]_{\nu\mu}$ is nonzero. The matrices G_m are very sparse: while G_0 is the identity matrix, it follows from (6.3) that the matrices $\{G_m\}_{m=1}^M$ have at most two nonzero entries per row (see also [37, Theorem 9.59]). This reduces the number of stiffness matrices to be computed to $(2M+1)\#\mathfrak{P}_\bullet$ at most. Furthermore, since the measure π_m is symmetric on $\Gamma_m = [-1,1]$ for all $m \in \mathbb{N}$, the matrices G_m, $m \in \mathbb{N}$, are also symmetric and have zero diagonal entries. In addition to the sparsity and symmetry of G_m, we observe that $K_m^{\nu\mu} = (K_m^{\mu\nu})^\mathsf{T}$ for all $m = 0, 1, \dots, M$ and $\nu, \mu \in \mathfrak{P}_\bullet$. Therefore, the number of stiffness matrices one actually needs to compute is at most $(M+1)\#\mathfrak{P}_\bullet$.

6.2. Computation of stiffness matrices. Let us now address the computation of the stiffness matrices $K_m^{\nu\mu}$ given by (6.4). To that end, we fix $m \in \{1, 2, \dots, M\}$ (the computation process is the same for each m) and set $\mu = \nu \pm \varepsilon_m \in \mathfrak{P}_\bullet$ for some $\nu \in \mathfrak{P}_\bullet$; cf. (6.3).

Note that the entries of $K_m^{\nu\mu}$ are the spatial integrals involving finite element basis functions associated with the meshes $\mathcal{T}_{\bullet\nu}$ and $\mathcal{T}_{\bullet\mu}$, which may be different and not necessarily nested. As a consequence, $K_m^{\nu\mu}$ are in general non-square if $\mathcal{T}_{\bullet\nu} \neq \mathcal{T}_{\bullet\mu}$, and efficient computation of these matrices is the main difficulty in the implementation of the multilevel stochastic Galerkin FEM.

The assembly of stiffness matrices in the context of the multilevel stochastic Galerkin FEM has been previously discussed in [30, 19, 16]. In [30], the action of any non-square stiffness matrix $K_m^{\nu(\nu\pm\varepsilon_m)}$ (in the context, e.g., of the preconditioned conjugate gradient method) is *approximated* via a projection $\Pi_\nu^{\nu\pm\varepsilon_m} : \mathbb{X}_{\bullet(\nu\pm\varepsilon_m)} \to \mathbb{X}_{\bullet\nu}$, such that only square matrices $K_m^{\nu\nu}$ need to be assembled. A more elaborate and computationally expensive approach involving the union of meshes $\mathcal{T}_{\bullet\nu}$ and $\mathcal{T}_{\bullet(\nu\pm\varepsilon_m)}$ is proposed in [19, Section 10]. Again, only square stiffness matrices need to be assembled. On the other hand, assuming that the meshes $\{\mathcal{T}_{\bullet\nu} : \nu \in \mathfrak{P}_\bullet\}$ (and, hence, the corresponding finite element spaces $\mathbb{X}_{\bullet\nu}$ in (3.9)) are *nested*, it is shown in [16] that non-square stiffness matrices $K_m^{\nu\mu}$ can be computed quickly and efficiently without resorting to approximations involving square matrices.

In our implementation, we aim for *direct* computation of non-square stiffness matrices $K_m^{\nu\mu}$ for a pair of general, not necessarily nested, meshes $\mathcal{T}_{\bullet\nu} \neq \mathcal{T}_{\bullet\mu} \in \operatorname{refine}(\mathcal{T}_0)$ (ν, $\mu = \nu \pm \varepsilon_m \in \mathfrak{P}_\bullet$).

First, exploiting the fact that the finite element basis functions $\varphi_{\bullet\nu,z}$ in our construction of $\mathbb{V}_\bullet$ are piecewise linear, we find

$$\begin{aligned}[K_m^{\nu\mu}]_{ij} &\overset{(6.4)}{=} \int_D a_m(x)\nabla\varphi_{\bullet\mu,z_j}\cdot\nabla\varphi_{\bullet\nu,z_i}\,\mathrm{d}x \\ &= \sum_{T_\nu\in\mathcal{T}_{\bullet\nu}} \sum_{\substack{T_\mu\in\mathcal{T}_{\bullet\mu}\\ |T_\mu\cap T_\nu|\neq 0}} \big(\nabla\varphi_{\bullet\mu,z_j}|_{T_\mu}\cdot\nabla\varphi_{\bullet\nu,z_i}|_{T_\nu}\big)\int_{T_\mu\cap T_\nu} a_m(x)\,\mathrm{d}x. \end{aligned} \tag{6.5}$$

Thus, efficient identification of all intersections $T_\mu \cap T_\nu$ is critical for the whole computation. The key observation here is that NVB is a binary refinement rule. Note that every element $T \in \mathcal{T}_\bullet \in \operatorname{refine}(\mathcal{T}_0)$ naturally comes with a *level* that can be defined in the following inductive way:

- for all $T \in \mathcal{T}_0$, define $\operatorname{level}(T) := 0$;
- if $T \in \mathcal{T}_\bullet \in \operatorname{refine}(\mathcal{T}_0)$ is bisected into two elements T_1 and T_2, then define $\operatorname{level}(T_1) := \operatorname{level}(T) + 1 =: \operatorname{level}(T_2)$.

Now, for any $T \in \mathcal{T}_\bullet \in \operatorname{refine}(\mathcal{T}_0)$, we denote by $T_0(T)$ the unique element of the initial mesh $\mathcal{T}_0$ such that $T \subseteq T_0(T)$. Then, the above definition implies that

$$|T|/|T_0(T)| = 2^{-\operatorname{level}(T)}. \tag{6.6}$$

Furthermore, there holds the following lemma, which, in particular, proves that the intersection $T_\mu \cap T_\nu$ is either T_μ, or T_ν, or a set of measure zero.

Lemma 8. *Let $\mathcal{T}_\bullet, \mathcal{T}'_\bullet \in \operatorname{refine}(\mathcal{T}_0)$. Let $T \in \mathcal{T}_\bullet$ and $T' \in \mathcal{T}'_\bullet$. Let $s_T \in T$ denote the center of mass of T. Then, there hold the following statements* (i)–(ii):

(i) If $\operatorname{level}(T) = \operatorname{level}(T')$, then there holds either $T = T'$ or $|T \cap T'| = 0$. Moreover, $T = T'$ is equivalent to $s_T \in \operatorname{interior}(T')$.

(ii) If $\mathrm{level}(T) > \mathrm{level}(T')$, then there holds either $T \subsetneqq T'$ or $|T \cap T'| = 0$. Moreover, $T \subsetneqq T'$ is equivalent to $s_T \in \mathrm{interior}(T')$.

Proof. Since NVB is a binary refinement rule, the intersection $T \cap T'$ satisfies one of the following four conditions:

- $|T \cap T'| = 0$;
- $T \cap T' = T = T'$;
- $T \cap T' = T \subsetneqq T'$;
- $T \cap T' = T' \subsetneqq T$.

Due to (6.6), knowing the element's level is sufficient for determining its size. Moreover, the center of mass of an element always lies in the interior of all of its NVB ancestors. ∎

Thus, given two meshes $\mathcal{T}_{\bullet\nu}, \mathcal{T}_{\bullet\mu} \in \mathrm{refine}(\mathcal{T}_0)$ for $\mu \neq \nu$, the computation of the matrix entries $[K_m^{\nu\mu}]_{ij}$ in (6.5) essentially boils down to the construction of two sets $\mathcal{U}_{\nu\mu}, \mathcal{U}_{\mu\nu}^\circ \subset \mathcal{T}_{\bullet\nu} \times \mathcal{T}_{\bullet\mu}$ satisfying the following properties (U1)–(U3):

(U1) For all $(T_\nu, T_\mu) \in \mathcal{U}_{\nu\mu}$, there holds $T_\nu \subseteq T_\mu$;

(U2) For all $(T_\nu, T_\mu) \in \mathcal{U}_{\mu\nu}^\circ$, there holds $T_\mu \subsetneqq T_\nu$;

(U3) $\mathcal{T}_{\bullet\nu} \oplus \mathcal{T}_{\bullet\mu} := \{T_\nu : (T_\nu, T_\mu) \in \mathcal{U}_{\nu\mu}\} \cup \{T_\mu : (T_\nu, T_\mu) \in \mathcal{U}_{\mu\nu}^\circ\}$ is a mesh[1] of D.

Indeed, with the sets $\mathcal{U}_{\nu\mu}, \mathcal{U}_{\mu\nu}^\circ$ at hand, the formula (6.5) for computing $[K_m^{\nu\mu}]_{ij}$ can be written as follows:

$$[K_m^{\nu\mu}]_{ij} = \sum_{(T_\nu,T_\mu)\in\mathcal{U}_{\nu\mu}} \big(\nabla\varphi_{\bullet\mu,z_j}|_{T_\mu} \cdot \nabla\varphi_{\bullet\nu,z_i}|_{T_\nu}\big) \int_{T_\nu} a_m(x)\,\mathrm{d}x + \sum_{(T_\nu,T_\mu)\in\mathcal{U}_{\mu\nu}^\circ} \big(\nabla\varphi_{\bullet\mu,z_j}|_{T_\mu} \cdot \nabla\varphi_{\bullet\nu,z_i}|_{T_\nu}\big) \int_{T_\mu} a_m(x)\,\mathrm{d}x.$$

The following searching algorithm provides a simple and surprisingly effective strategy for constructing the sets $\mathcal{U}_{\nu\mu}$ and $\mathcal{U}_{\mu\nu}^\circ$. In this algorithm, for each simplex T, we denote by $s_T \in T$ the center of mass of T. Furthermore, we denote by $\lambda_{T,1}(x)$, $\lambda_{T,2}(x)$, $\lambda_{T,3}(x)$ the barycentric coordinates of $x \in D$ with respect to T, i.e., $x = \sum_{j=1}^3 \lambda_{T,j}(x) z_{T,j}$ and $\sum_{j=1}^3 \lambda_{T,j}(x) = 1$, where $z_{T,1}$, $z_{T,2}$, $z_{T,3}$ are the vertices of T. We recall that $\lambda_{T,j}(x)$ are uniquely defined for given x and T, and $x \in T$ is equivalent to $\lambda_{T,1}(x), \lambda_{T,2}(x), \lambda_{T,3}(x) \geq 0$.

Algorithm 9 (construction of $\mathcal{U}_{\nu\mu}$ and $\mathcal{U}_{\mu\nu}^\circ$). ***Input:*** *Meshes $\mathcal{T}_{\bullet\nu}$ and $\mathcal{T}_{\bullet\mu}$.*

1: ***for all*** $T_0 \in \mathcal{T}_0$ ***do***
2: *Define* $\mathcal{T}_{\bullet\nu}|_{T_0} := \{T_\nu \in \mathcal{T}_{\bullet\nu} : T_\nu \subseteq T_0\} \subseteq \mathcal{T}_{\bullet\nu}$.
3: *Define* $\mathcal{T}_{\bullet\mu}|_{T_0} := \{T_\mu \in \mathcal{T}_{\bullet\mu} : T_\mu \subseteq T_0\} \subseteq \mathcal{T}_{\bullet\mu}$.
4: ***for all*** $T_\nu \in \mathcal{T}_{\bullet\nu}|_{T_0}$ ***do***
5: *Define* $\mathcal{V}_{\bullet\mu}(T_\nu) := \{T_\mu \in \mathcal{T}_{\bullet\mu}|_{T_0} : \mathrm{level}(T_\mu) \leq \mathrm{level}(T_\nu)\} \subseteq \mathcal{T}_{\bullet\mu}|_{T_0}$.
6: *Compute* $\lambda_{T_\mu,i}(s_{T_\nu})$ *for all* $i = 1,2,3$ *and* $T_\mu \in \mathcal{V}_{\bullet\mu}(T_\nu)$.
7: ***if*** *there exists (a unique)* $T_\mu \in \mathcal{V}_{\bullet\mu}(T_\nu)$ *with* $\lambda_{T_\mu,i}(s_{T_\nu}) > 0$ *for all* $i = 1,2,3$ ***then***
8: *Assign* (T_ν, T_μ) *to* $\mathcal{U}_{\nu\mu}$ *(because* $T_\nu \subseteq T_\mu$*)*.
9: ***else***

[1] Note that the notation used in (U3) is deliberate, in the sense that $\mathcal{T}_{\bullet\nu} \oplus \mathcal{T}_{\bullet\mu}$ is indeed the overlay of the meshes $\mathcal{T}_{\bullet\nu}$ and $\mathcal{T}_{\bullet\mu}$ (i.e., their coarsest common refinement).

Compute $\lambda_{T_\nu,i}(s_{T_\mu})$ *for all* $i = 1,2,3$ *and* $T_\mu \in \mathcal{T}_{\bullet\mu}|_{T_0} \setminus \mathcal{V}_{\bullet\mu}(T_\nu)$.
Define $\mathcal{W}_{\bullet\mu}(T_\nu) := \{T_\mu \in \mathcal{T}_{\bullet\mu}|_{T_0} \setminus \mathcal{V}_{\bullet\mu}(T_\nu) : \lambda_{T_\nu,i}(s_{T_\mu}) > 0 \text{ for all } i = 1,2,3\}$.
Assign (T_ν, T_μ) *to* $\mathcal{U}^\circ_{\mu\nu}$ *for all* $T_\mu \in \mathcal{W}_{\bullet\mu}(T_\nu)$ *(because* $T_\mu \subsetneqq T_\nu$ *if* $T_\mu \in \mathcal{W}_{\bullet\mu}(T_\nu)$*).*
end if
end for
end for

Output: *Sets* $\mathcal{U}_{\nu\mu}$ *and* $\mathcal{U}^\circ_{\mu\nu}$ *satisfying* (U1)–(U3).

Algorithm 9 has a computational complexity of $\mathcal{O}((\#\mathcal{T}_{\bullet\nu})(\#\mathcal{T}_{\bullet\mu}))$ in the worst case. However, its only intention is to show that unlike [19] it is possible to compute stiffness matrices associated to different meshes exactly (up to quadrature). We conjecture that one can build the matrix $K_m^{\nu\mu}$ from (6.5) in log-linear complexity $\mathcal{O}((\#\mathcal{T}_{\bullet\nu} + \#\mathcal{T}_{\bullet\mu})\log(\#\mathcal{T}_{\bullet\nu} + \#\mathcal{T}_{\bullet\mu}))$ by exploiting the binary tree structure of NVB. This aspect of the implementation will be the subject of future research.

6.3. Numerical solution of Galerkin system. Efficient linear solver is an important ingredient of any stochastic Galerkin implementation. Sparse factorizations of the (full) system matrix $\mathbf{A}$ are memory intensive and computationally costly, therefore, performing those efficiently is not feasible. In fact, the coefficient matrix $\mathbf{A}$ is never explicitly assembled in stochastic Galerkin FEM implementations (see, e.g., [19, 16, 9]). Instead, 'matrix-free' iterative solvers are employed, where the matrix-vector products with $\mathbf{A}$ are computed blockwise from individual matrix components of $\mathbf{A}$ as follows:

$$[\mathbf{A}\mathbf{x}]_\nu = \sum_{\mu\in\mathfrak{P}_\bullet} \mathbf{A}_{\nu\mu}\mathbf{x}_\mu \overset{(6.2)}{=} \sum_{\mu\in\mathfrak{P}_\bullet}\sum_{m=0}^{M} [G_m]_{\nu\mu} K_m^{\nu\mu}\mathbf{x}_\mu, \quad \mathbf{x} = (\mathbf{x}_\mu)_{\mu\in\mathfrak{P}_\bullet}, \quad \nu \in \mathfrak{P}_\bullet.$$

The default iterative solver in Stochastic T-IFISS is a bespoke implementation of the Minimum Residual method, called ESTMINRES [42] (an alternative solver based on the conjugate gradient method and utilizing the built-in MATLAB function `pcg` is included as an option).

For the iterative solver to be fast, it requires a suitably chosen preconditioner. In the context of stochastic Galerkin FEM, particularly for parametric PDEs with coefficients having linear dependence on the parameters, the mean-based preconditioner [28, 38] is a standard choice (for alternative approaches, we refer, e.g., to [46, 43, 3]). Specifically, we employ a block-diagonal preconditioner with diagonal blocks given by the stiffness matrices $K_0^{\nu\nu}$, $\nu \in \mathfrak{P}_\bullet$, defined in (6.4). Thus, the action of the inverse of the preconditioner on residual vectors can be effected blockwise. For each $\nu \in \mathfrak{P}_\bullet$, this is done by computing sparse triangular factorizations of $K_0^{\nu\nu}$, followed by forward and backward substitutions on the corresponding block of the residual vector. In agreement with theoretical results in [38] for the *single-level* stochastic Galerkin FEM, our experiments with *multilevel* approximations have shown that the number of preconditioned ESTMINRES iterations needed to satisfy the default tolerance of 10^{-9} is less than 20, independent of $\#\mathfrak{P}_\bullet$ and the resolution of finite element meshes in the multilevel construction.

7. Numerical experiments. In this section, we present a collection of numerical results that illustrate the effectiveness of the error estimation strategy developed in section 4 and

demonstrate the performance of the multilevel adaptive algorithms described in section 5. Here, we stay within the context of the two-dimensional diffusion problem (2.1) with the parametric coefficient $\boldsymbol{a} = \boldsymbol{a}(x, \mathbf{y})$ in the affine form (2.2) satisfying assumptions (2.3)–(2.4). In addition, we assume that the parameters $\mathbf{y} = (y_m)_{m\in\mathbb{N}}$ are images of independent uniformly distributed mean-zero random variables on $[-1, 1]$, i.e., $\mathrm{d}\pi_m(y_m) = \mathrm{d}y_m/2$ for all $m \in \mathbb{N}$. All computations have been performed using the MATLAB toolbox Stochastic T-IFISS; see section 6.

In our experiments, we use five adaptive algorithms: two multilevel algorithms with separate spatial and parametric enrichments (i.e., Algorithm 7.A and Algorithm 7.B from section 5), their single-level precursors (see, e.g., Algorithms 4.A and 4.B in [5], respectively), and the novel multilevel algorithm with combined enrichment (Algorithm 7.C). For the sake of brevity, we will refer to these five algorithms as `ML-A`, `ML-B`, `SL-A`, `SL-B`, and `ML-C`, respectively. The parameters in these algorithms are selected as follows:

- We set the marking parameters $\theta_{\mathbb{X}} = \theta_{\mathfrak{P}} = 0.5$ in `ML-A`, `ML-B`, `SL-A`, `SL-B` and $\theta = 0.5$ in `ML-C`.
- For the parameter $\overline{M}$ in (3.8), we choose $\overline{M} = 1$ in subsection 7.1 and $\overline{M} = 9$ in subsection 7.2.
- Except in the last experiment in subsection 7.2, the parameter ϑ modulating the choice of the enrichment type in the algorithms with separate spatial and parametric enrichments (i.e., `ML-A`, `ML-B` and `SL-A`, `SL-B`) is chosen to be $\vartheta = 1$.

7.1. Benchmark problem. The following problem has been considered in several works addressing the numerical approximation of parametric PDEs (see, e.g., in [19, 20, 10, 22, 7, 16, 5]) and has thus become a benchmark problem for testing novel discretization strategies. Let $\boldsymbol{f} \equiv 1$ in (2.1) and choose the expansion coefficients in (2.2) to represent planar Fourier modes of increasing total order; for $x = (x_1, x_2)$, these coefficients are given by

$$a_0(x) = 1, \quad a_m(x_1, x_2) = Am^{-\sigma} \cos(2\pi\beta_1(m)x_1) \cos(2\pi\beta_2(m)x_2) \quad \text{for } m \in \mathbb{N},$$

where $A, \sigma > 0$ are constants, $\beta_1(m) = m - k(m)[k(m) + 1]/2$, $\beta_2(m) = k(m) - \beta_1(m)$, and $k(m) = \lfloor -1/2 + \sqrt{1/2 + 2m} \rfloor$. With this choice, the diffusion coefficient $\boldsymbol{a}(x, \mathbf{y})$ trivially satisfies (2.3) with $a_0^{\min} = a_0^{\max} = 1$. Furthermore, we set $\sigma = 2$ (yielding a slow decay of the coefficients) and choose $A = 0.9/\zeta(\sigma) \approx 0.547$, so that both inequalities in (2.4) are satisfied (here, $\zeta(\cdot)$ denotes the Riemann zeta function).

7.1.1. Square domain. Let us numerically solve the benchmark problem on the square domain $D = (0, 1)^2$. For all algorithms, we choose the initial mesh $\mathcal{T}_0$ to be a uniform mesh of 512 right-angled triangles and we terminate computations when the error estimate τ_ℓ given by (4.8) falls below the tolerance $\mathtt{tol} = 6 \cdot 10^{-4}$.

In the first experiment, we assess the effectiveness of our error estimation strategy by computing the error estimate τ_ℓ at each iteration of the adaptive loop and comparing τ_ℓ with the energy norm of the true error $\boldsymbol{u} - \boldsymbol{u}_\ell$ approximated by

$$|\!|\!| \boldsymbol{u} - \boldsymbol{u}_\ell |\!|\!| = \big(|\!|\!| \boldsymbol{u} |\!|\!|^2 - |\!|\!| \boldsymbol{u}_\ell |\!|\!|^2 \big)^{1/2} \approx \big(|\!|\!| \boldsymbol{u}_{\mathrm{ref}} |\!|\!|^2 - |\!|\!| \boldsymbol{u}_\ell |\!|\!|^2 \big)^{1/2}.$$

Here, the equality follows from the Galerkin orthogonality and the unknown energy $|\!|\!| \boldsymbol{u} |\!|\!|$ is

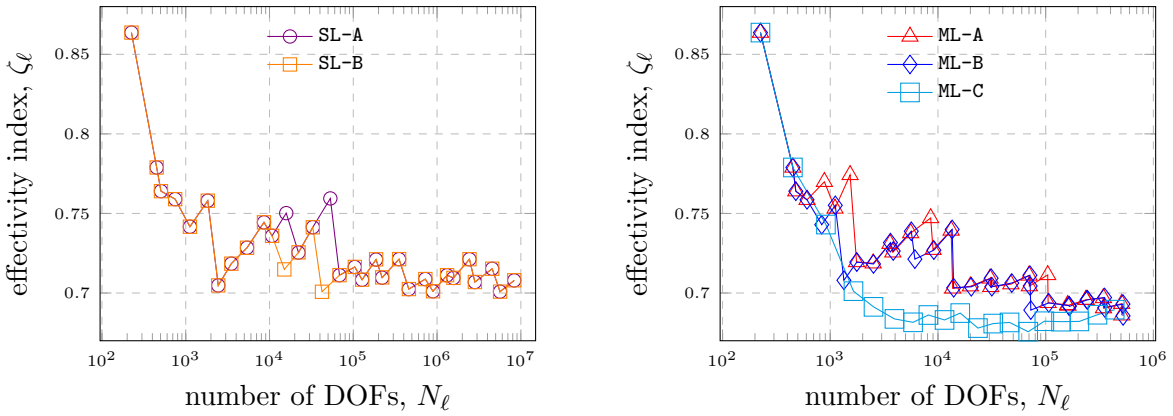


Figure 1: Experiments in subsection 7.1.1: Effectivity indices ζ_ℓ for the error estimates τ_ℓ in the SGFEM approximations generated by single-level (left) and multilevel (right) adaptive algorithms.

approximated by the energy of a sufficiently accurate reference solution $\boldsymbol{u}_{\mathrm{ref}}$ computed with quadratic (Q2) SGFEM approximations; cf. [10, Section 6]. The *effectivity index*

$$\zeta_\ell := \frac{\tau_\ell}{\left(\| \boldsymbol{u}_{\mathrm{ref}} \|^2 - \| \boldsymbol{u}_\ell \|^2\right)^{1/2}}$$

is then computed at each iteration of the adaptive loop.

In Figure 1, for all adaptive algorithms, we plot the effectivity indices ζ_ℓ versus the total number of degrees of freedom (DOFs) N_ℓ in SGFEM approximations. For each algorithm, the effectivity indices vary in a range between 0.68 and 0.87 throughout all iterations. The error is therefore slightly underestimated. For single-level approximations generated by `SL-A` and `SL-B`, this is in agreement with the results presented in [5, Figure 3]. Thus, this experiment provides a numerical evidence that in terms of effectivity, our error estimation strategy for multilevel SGFEM approximations is on a par with similar strategies for single-level approximations. The presented results also suggest that by employing the two-level spatial error estimates we underestimate the true energy error more than by using hierarchical spatial estimates; see [7] and [16] for hierarchical spatial estimates in adaptive single-level and multilevel SGFEMs, respectively. However, the better accuracy of hierarchical estimators comes at the price of solving extra linear systems when computing spatial contributions to the total error estimate at each iteration.

Figure 2 (left) shows the decay of the error estimates τ_ℓ versus the total number of degrees of freedom N_ℓ in SGFEM approximations generated by five adaptive algorithms. For single-level approximations, the error estimates decay with suboptimal rate $\mathcal{O}(N_\ell^{-0.33})$; the same rate was observed in [7]. For multilevel approximations, the decay rate is much faster. In particular, for approximations generated by `ML-C`, the error estimates decay with the optimal rate $\mathcal{O}(N_\ell^{-0.5})$, which is the convergence rate of linear (P1) FEM for the corresponding parameter-free problem. As a consequence, multilevel SGFEM approximations reach the pre-

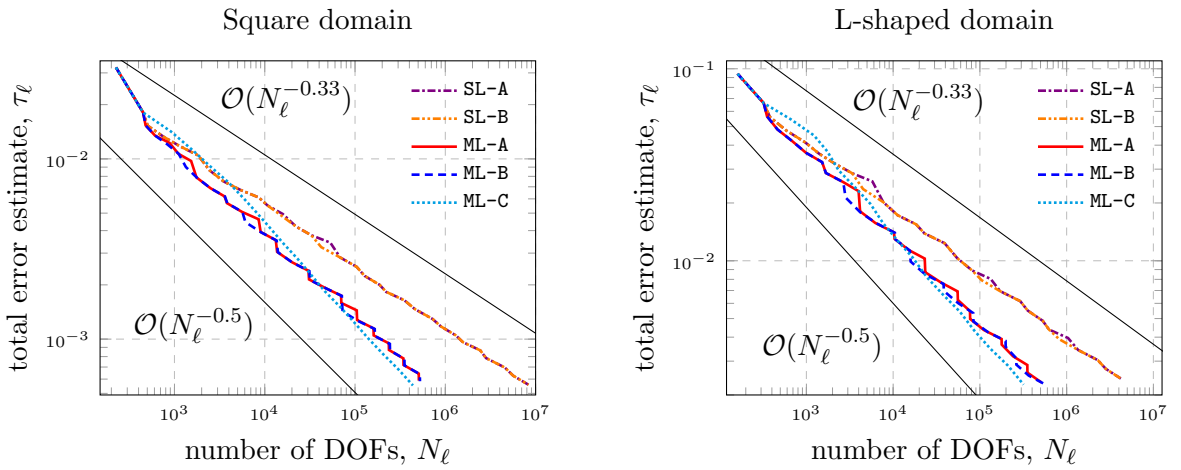


Figure 2: Experiments in subsection 7.1.1 (left) and subsection 7.1.2 (right): Total error estimates τ_ℓ versus the number of degrees of freedom N_ℓ for all adaptive algorithms.

scribed accuracy with significantly less degrees of freedom than their single-level counterparts (in the asymptotic regime, the number of degrees of freedom in multilevel approximations are less by at least one order of magnitude compared to the number of degrees of freedom in the single-level approximations having the same accuracy).

7.1.2. L-shaped domain. Let us now consider the benchmark problem on the L-shaped domain $D = (-1,1)^2 \setminus (-1,0]^2$. In contrast to the problem in subsection 7.1.1, the exact solution $\boldsymbol{u}$ now exhibits a geometric singularity at the reentrant corner. For this problem, we run all five adaptive algorithms with the same initial mesh $\mathcal{T}_0$ (a uniform mesh of 384 right-angled triangles) and the same stopping tolerance $\texttt{tol} = 2.5 \cdot 10^{-3}$.

In Figure 2 (right), for all adaptive algorithms, we plot the error estimates τ_ℓ against the number of degrees of freedom N_ℓ. Despite the singular behavior of the exact solution, we observe the same empirical convergence rates as in the previous experiment on the square domain. In particular, the error estimates for all multilevel approximations decay much faster than those for single-level approximations, while the latter converge with suboptimal rate $\mathcal{O}(N_\ell^{-0.33})$.

Let us look in more detail at the performance of multilevel algorithms in this experiment. In Figure 3, for the algorithms `ML-A`, `ML-B` and `ML-C`, we plot the total error estimates τ_ℓ along with their spatial and parametric components given by

$$\tau_{\mathbb{X}_\ell} := \Big(\sum_{\nu \in \mathfrak{P}_\ell} \sum_{z \in \mathcal{N}_{\ell\nu}^+} \tau_\ell(\nu, z)^2 \Big)^{1/2} \quad \text{and} \quad \tau_{\mathfrak{P}_\ell} := \Big(\sum_{\nu \in \mathfrak{Q}_\ell} \tau_\ell(\nu)^2 \Big)^{1/2},$$

respectively, and the reference energy error $\| \boldsymbol{u}_{\mathrm{ref}} - \boldsymbol{u}_\ell \|$, where $\boldsymbol{u}_{\mathrm{ref}}$ denotes a reference solution computed by running the algorithm `ML-C` to a lower tolerance. Note that $\tau_\ell^2 = \tau_{\mathbb{X}_\ell}^2 + \tau_{\mathfrak{P}_\ell}^2$; see (4.8). For the algorithms with separate spatial and parametric enrichments (i.e., `ML-A` and

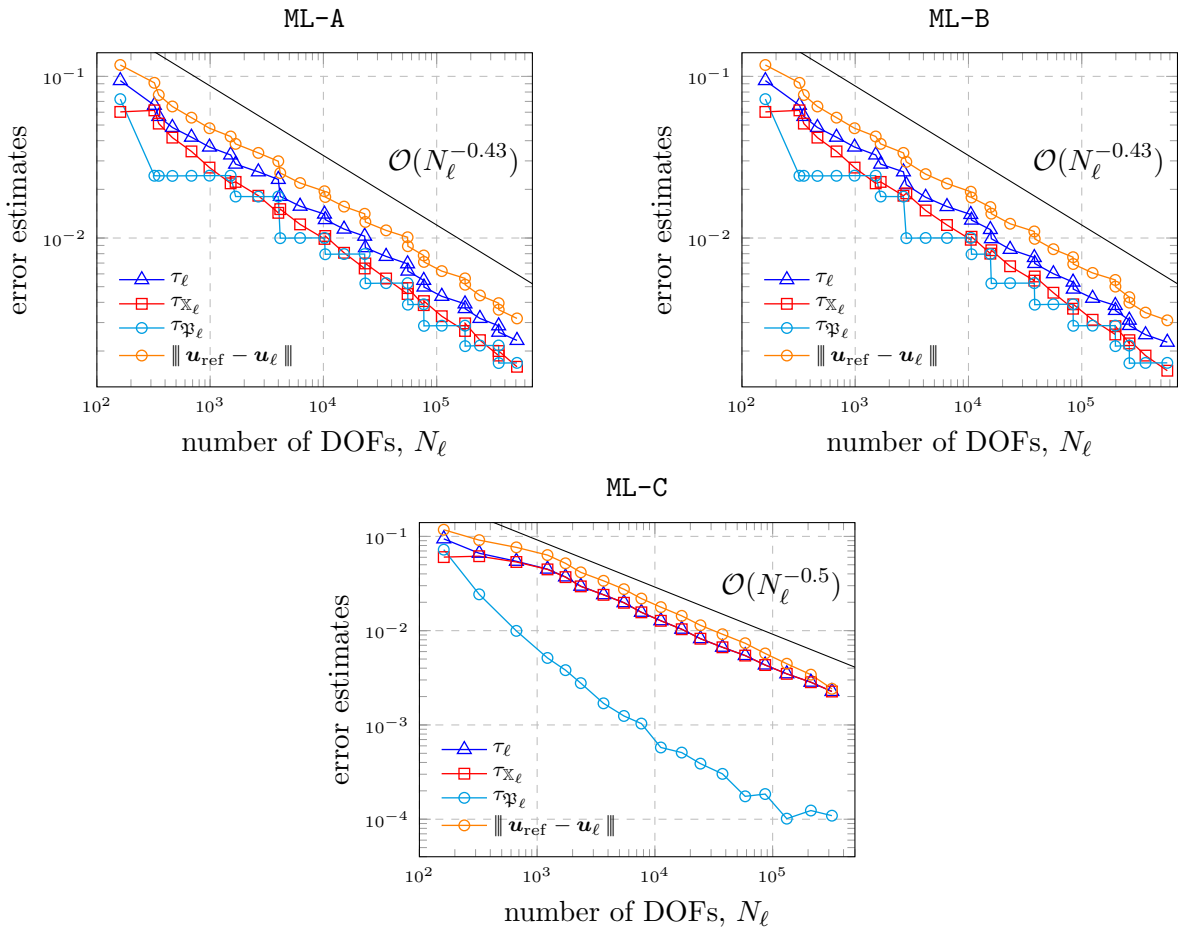


Figure 3: Experiments in subsection 7.1.2: Decay of the error estimates (total, spatial, and parametric) and the reference errors computed at each iteration of the adaptive multilevel algorithms.

ML-B), the plots in Figure 3 look very similar. For both these algorithms, we observe that the parametric error estimates $\tau_{\mathfrak{P}_\ell}$ remain essentially constant during mesh refinement iterations, whereas the spatial error estimates $\tau_{\mathbb{X}_\ell}$ exhibit a noticeable increase at the iteration following each parametric enrichment. The latter observation is a consequence of assigning the coarse mesh $\mathcal{T}_0$ to every new index introduced by the parametric enrichment. As a result, the decay rates of the total error estimates τ_ℓ for ML-A, ML-B are still suboptimal.

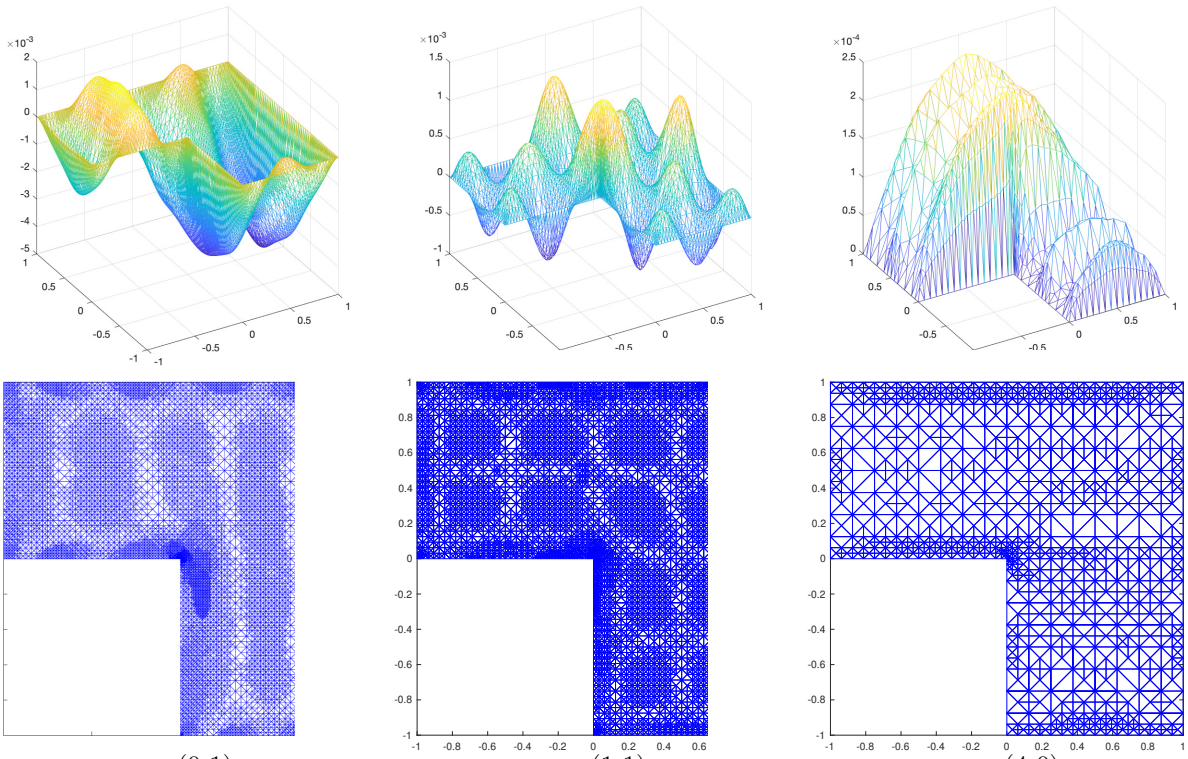
By looking at the plot for the algorithm with combined enrichment (i.e., ML-C) we see a completely different behavior. The balanced enrichment of spatial and parametric components of Galerkin approximations that was inherent to ML-A and ML-B is completely lost in ML-C. Instead, ML-C clearly privileges parametric enrichment by activating significantly more indices than ML-A and ML-B (see also Table 1). This is a consequence of the combined marking strategy (5.5) and the fact that a small number of parametric error indicators are

	ML-A		ML-B		ML-C	
L	28		28		17	
τ_L	$2.32526 \cdot 10^{-3}$		$2.26684 \cdot 10^{-3}$		$2.26429 \cdot 10^{-3}$	
N_L	511 812		569 321		318 897	
$\#\mathfrak{P}_L$	17		17		207	
$\deg \mathfrak{P}_L$	4		4		7	
$M_{\mathfrak{P}_L}$	7		7		17	
$\mathfrak{P}_\ell$	$\ell = 0$	(0 0)	$\ell = 0$	(0 0)	$\ell = 0$	(0 0)
	$\ell = 1$	(1 0)	$\ell = 1$	(1 0)	$\ell = 1$	(1 0)
	$\ell = 7$	(0 1)	$\ell = 7$	(0 1)	$\ell = 2$	(0 1) (2 0)
	$\ell = 10$	(2 0)	$\ell = 9$	(2 0)	$\ell = 3$	(0 0 1) (1 1 0) (3 0 0)
	$\ell = 13$	(0 0 1)	$\ell = 13$	(0 0 1)	$\ell = 4$	(0 0 0 1) (1 0 1 0)
	$\ell = 16$	(1 1 0) (3 0 0)	$\ell = 15$	(1 1 0) (3 0 0)	$\ell = 5$	(0 0 0 0 1) (2 1 0 0 0)
	$\ell = 19$	(0 0 0 1) (1 0 1 0)	$\ell = 18$	(0 0 0 1) (1 0 1 0)	$\ell = 6$	(0 0 0 0 0 1) (1 0 0 1 0 0) (2 0 1 0 0 0) (0 2 0 0 0 0) (4 0 0 0 0 0)
	$\ell = 21$	(0 0 0 0 1) (2 1 0 0 0)	$\ell = 21$	(0 0 0 0 1) (2 1 0 0 0)	$\ell = 7$ $\ell = 8$	5 indices 5 indices
	$\ell = 24$	(0 0 0 0 0 1) (2 0 1 0 0 0) (1 0 0 1 0 0)	$\ell = 24$	(0 0 0 0 0 1) (2 0 1 0 0 0) (1 0 0 1 0 0)	$\ell = 9$ $\ell = 10$ $\ell = 11$	7 indices 8 indices 9 indices
	$\ell = 27$	(0 2 0 0 0 0 0) (0 0 0 0 0 0 1) (4 0 0 0 0 0 0)	$\ell = 26$	(0 2 0 0 0 0 0) (0 0 0 0 0 0 1) (4 0 0 0 0 0 0)	$\ell = 12$ $\ell = 13$ $\ell = 14$ $\ell = 15$ $\ell = 16$ $\ell = 17$	19 indices 16 indices 16 indices 33 indices 38 indices 35 indices

Table 1: Experiments in subsection 7.1.2: Final outputs and evolution of the index set for adaptive multilevel algorithms.

larger in magnitude than a significant proportion of spatial error indicators. This results in the parametric error estimates $\tau_{\mathfrak{P}_\ell}$ decaying much faster than their spatial counterparts $\tau_{\mathbb{X}_\ell}$. However, the total error estimate τ_ℓ decays with fully optimal rate $\mathcal{O}(N_\ell^{-0.5})$.

In Table 1, for each multilevel adaptive algorithm, we show the total number of iterations L, the final value of the total error estimate τ_L, the number of degrees of freedom in the final SGFEM approximation, as well as the cardinality of the final index set $\mathfrak{P}_L$, the (total) degree $\deg \mathfrak{P}_L := \max_{\nu \in \mathfrak{P}_L} \sum_{j \geq 1} \nu_j$ of polynomials in the associated polynomial space, and the number of active parameters $M_{\mathfrak{P}_L}$ in $\mathfrak{P}_L$. We also show the evolution of the index set throughout each computation. By looking at these results, we observe that in order to reach the prescribed tolerance, the algorithm with combined enrichment requires significantly less iterations and generates the final Galerkin approximation with significantly less degrees of freedom than either of the algorithms with separate enrichments. In addition to this, these



$\nu = (0\ 1)$
$\#\mathcal{T}_{L\nu} = 24\,664$

$\nu = (1\ 1)$
$\#\mathcal{T}_{L\nu} = 11\,550$

$\nu = (4\ 0)$
$\#\mathcal{T}_{L\nu} = 1809$

Figure 4: Experiments in subsection 7.1.2: Coefficients $u_{L\nu} \in \mathbb{X}_{L\nu} = \mathcal{S}^1(\mathcal{T}_{L\nu})$ of the final SGFEM approximation generated by `ML-C` (top plots) and the associated adaptively refined meshes $\mathcal{T}_{L\nu}$ (bottom plots) for three indices $\nu \in \mathfrak{P}_L$.

two types of multilevel algorithms generate Galerkin approximations with remarkably different distributions of spatial and parametric degrees of freedom. Specifically, while `ML-A` and `ML-B` produce relatively small index sets and fine meshes for most of the indices, `ML-C` generates a much larger index set but very coarse meshes for the majority of indices. The latter feature resembles that of multilevel sampling methods, where very few deterministic PDE solves are performed on fine spatial meshes while the majority of solves use coarse meshes.

In Figure 4, for the final SGFEM approximation

$$\boldsymbol{u}_L = \sum_{\nu \in \mathfrak{P}_L} u_{L\nu} P_\nu \in \mathbb{V}_L$$

generated by `ML-C`, we plot the coefficients $u_{L\nu} \in \mathbb{X}_{L\nu}$ and the associated meshes $\mathcal{T}_{L\nu}$ for $\nu \in \{(0\ 1), (1\ 1), (4\ 0)\} \subset \mathfrak{P}_L$. Meshes with similar patterns were produced by all other multilevel algorithms. We observe that adaptively refined meshes identify the geometric singularity at the reentrant corner (affecting all coefficients in the same way) and the regions with steep

gradient (which are different for each coefficient). All the identified areas exhibit much stronger mesh refinement than elsewhere in the domain. More importantly, finer meshes are produced for those coefficients that are more 'influential' in the Galerkin solution (i.e., the coefficients whose indices are activated earlier); cf. the values of $\#\mathcal{T}_{L\nu}$ in Figure 4. This illustrates how the flexibility in allocating degrees of freedom ensures greater efficiency of multilevel methods, compared to the single-level SGFEM.

7.2. Cookie problem. Our second example of parametric problem (2.1)–(2.2) is the so-called *cookie problem*; cf. [2, 23]. We consider the square domain $D = (0,1)^2$ that contains nine circular inclusions $D_m \subset D$ $(m = 1, \dots, 9)$. For all $i, j \in \{1, 2, 3\}$, the subdomain $D_{i+3(j-1)}$ is the disk with center at the point $((2i-1)/6, (2j-1)/6)$ and radius $r = 1/8$. We set $\boldsymbol{f} \equiv 1$ in (2.1) and select the expansion coefficients in (2.2) as follows:

$$a_m(x) = \begin{cases} 1 & \text{for } m = 0, \\ 0.5\,\chi_{D_m}(x) & \text{for } m = 1,3,7,9, \\ 0.7\,\chi_{D_m}(x) & \text{for } m = 2,4,6,8, \\ 0.9\,\chi_{D_m}(x) & \text{for } m = 5, \\ 0 & \text{for } m > 9 \end{cases} \qquad \text{for all } x \in D, \tag{7.1}$$

where χ_{D_m} denotes the characteristic function of the subdomain D_m. Thus, the diffusion coefficient $\boldsymbol{a}(x, \mathbf{y})$ in this example depends on finitely many parameters $y_1, \dots, y_9 \in [-1, 1]$; furthermore, assumptions (2.3)–(2.4) are satisfied (with $a_0^{\min} = a_0^{\max} = 1$ and $\tau = 0.9$).

We emphasize that, in contrast to the benchmark problem in subsection 7.1, where the amplitude of the coefficient a_m in the expansion (2.2) decays as m increases, which induces a hierarchy of the parameters (with y_m being more 'important' than y_ℓ if $m < \ell$), in this example the 'importance' of the parameters cannot be directly inferred from the ordering of the terms in expansion (2.2). Hence, one should not *a priori* prescribe any specific order in which the parameters are activated. That is why, when running adaptive algorithms for the cookie problem, we set $\overline{M} = 9$ in (3.8) (note that in this example $\mathfrak{I} := \mathbb{N}_0^9$). This way, when it comes to the first parametric enrichment, all parameters are available for activation, and the order in which they are activated is determined by the associated parametric indicators.

In computations with all five adaptive algorithms for this problem, we set the stopping tolerance $\texttt{tol} = 8.0 \cdot 10^{-4}$ and use the same initial mesh $\mathcal{T}_0$ as in subsection 7.1.1.

In Figure 5, for all adaptive algorithms, we plot the error estimates τ_ℓ against the number of degrees of freedom N_ℓ. The results are in agreement with those presented in subsection 7.1: (i) For single-level approximations, the error estimates decay with suboptimal rate $\mathcal{O}(N_\ell^{-0.33})$; (ii) The decay rates for the multilevel approximations generated by `ML-A` and `ML-B` are faster than $\mathcal{O}(N_\ell^{-0.33})$ but not optimal; (iii) For the multilevel approximations generated by `ML-C`, the error estimates decay with fully optimal rate $\mathcal{O}(N_\ell^{-0.5})$.

In Figure 6, for algorithms `ML-A`, `ML-B` and `ML-C`, we plot the total error estimates τ_ℓ along with their spatial and parametric components $\tau_{\mathbb{X}_\ell}$ and $\tau_{\mathfrak{P}_\ell}$, as well as the reference energy error $|||\, \boldsymbol{u}_{\mathrm{ref}} - \boldsymbol{u}_\ell \,|||$, where $\boldsymbol{u}_{\mathrm{ref}}$ denotes a reference solution computed by running the algorithm `ML-C` to a lower tolerance ($\texttt{tol} = 2.0 \cdot 10^{-4}$).

In Table 2, we show the outputs for the multilevel algorithms. Each algorithm activates all

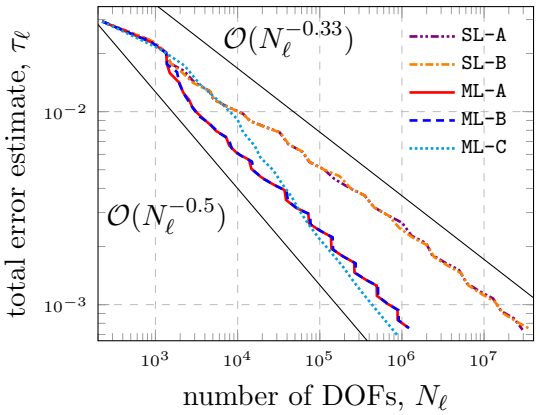


Figure 5: Experiments in subsection 7.2: Total error estimates τ_ℓ versus the number of degrees of freedom N_ℓ for all adaptive algorithms.

	ML-A	ML-B	ML-C
L	37	37	21
τ_L	$7.60064 \cdot 10^{-4}$	$7.55016 \cdot 10^{-4}$	$6.86986 \cdot 10^{-4}$
N_L	1 188 953	1 223 401	897 023
$\#\mathfrak{P}_L$	73	73	629
$\deg \mathfrak{P}_L$	8	8	17
$M_{\mathfrak{P}_L}$	9	9	9

Table 2: Experiments in subsection 7.2: Final outputs for adaptive multilevel algorithms.

nine relevant parameters $y_1, \ldots, y_9$. While we do not observe significant differences between ML-A and ML-B, we see that ML-C reaches the prescribed tolerance with less iterations, a smaller number of degrees of freedom, a richer index set, and a higher polynomial degree than the two other algorithms (see also Figure 7, where we show the evolution of $\#\mathfrak{P}_\ell$). This is again in agreement with the results presented in subsection 7.1.

In Figure 8, we consider an intermediate SGFEM approximation $\boldsymbol{u}_\ell \in \mathbb{V}_\ell$ generated by ML-C ($\ell = 16$). For five indices in $\mathfrak{P}_\ell$, namely $\nu = \mathbf{0}$, three unit indices $\nu = \varepsilon_1, \varepsilon_2, \varepsilon_5$, and $\nu = (1\ 0\ 0\ 1)$, we plot the coefficients $u_{\ell\nu} \in \mathbb{X}_{\ell\nu}$ and the associated adaptively refined meshes $\mathcal{T}_{\ell\nu}$. Note that the coefficient associated with $\nu = \mathbf{0}$ represents the expectation of the SGFEM approximation. Looking at the mesh associated with $\nu = \mathbf{0}$, we observe that the intensity of local mesh refinement at the boundary of each subdomain reflects the 'importance' of the corresponding parameter (cf. (7.1)). Moreover, we observe that for each $m = 1, 2, 5$, the subdomain D_m is identified by the mesh associated with the index ε_m. In the same way, the mesh associated with $\nu = (1\ 0\ 0\ 1)$ identifies the subdomains D_1 and D_4.

Next, we consider the final index set $\mathfrak{P}_L$ generated by ML-C ($L = 21$) and assess the

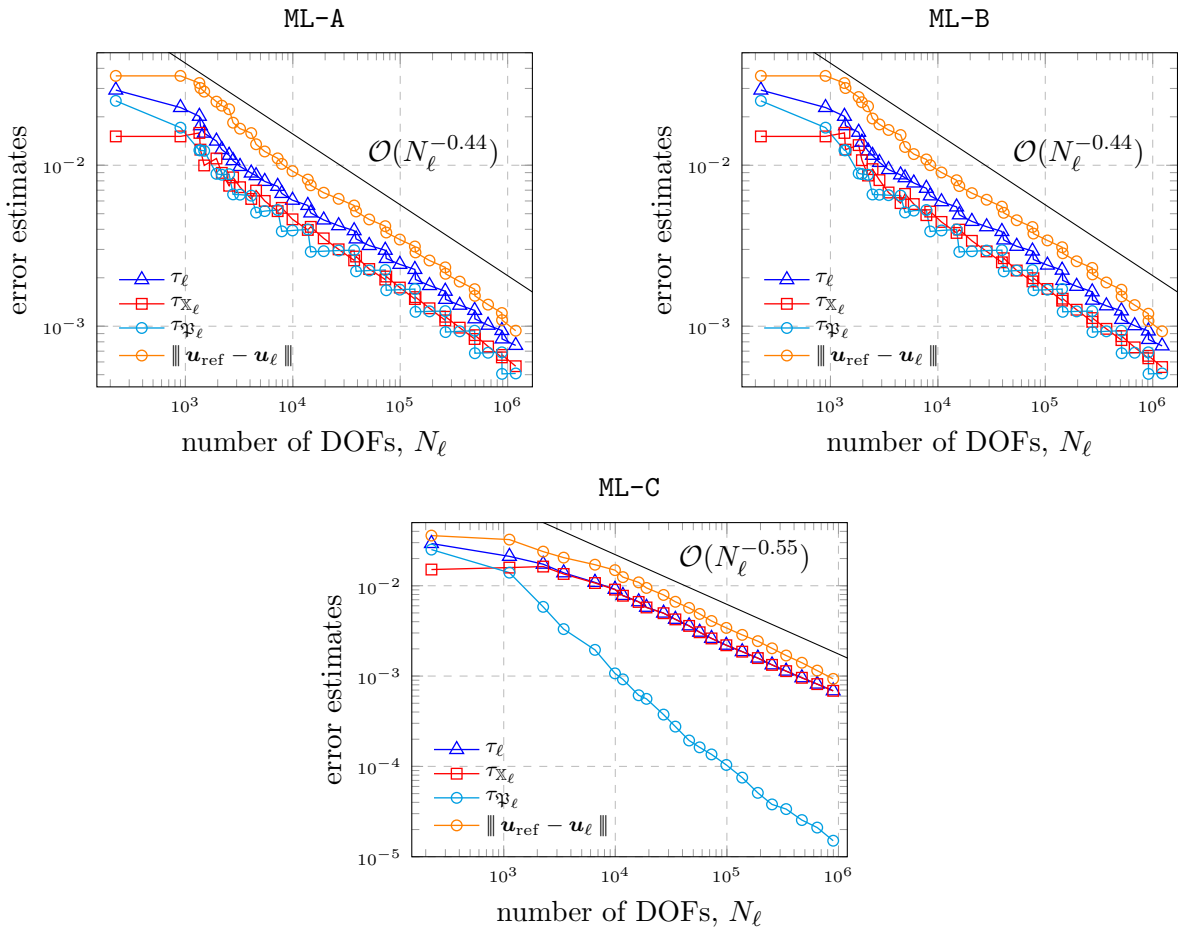


Figure 6: Experiments in subsection 7.2: Decay of the error estimates (total, spatial, parametric) and the reference errors computed at each iteration of the adaptive multilevel algorithm.

maximum polynomial degree activated for each parameter y_m $(m = 1, \dots, 9)$:

$$\max_{\nu \in \mathfrak{P}_L} \nu_m = \begin{cases} 6 & \text{for } m = 1, 3, 7, 9, \\ 9 & \text{for } m = 2, 4, 6, 8, \\ 17 & \text{for } m = 5. \end{cases}$$

We see that the maximum polynomial degrees assigned to the parameters mirror the hierarchy of the parameters induced by the coefficients (cf. (7.1)). This result, together with those reported in Figure 8, illustrate the capability of our multilevel fully adaptive algorithm to capture the anisotropy of the inclusions and allocate degrees of freedom according to the 'importance' of both the individual parameters and the gPC expansion modes.

In our final experiment, we investigate whether appropriately selecting the parameter $\vartheta > 0$, which modulates the choice between mesh refinement and parametric enrichment, can

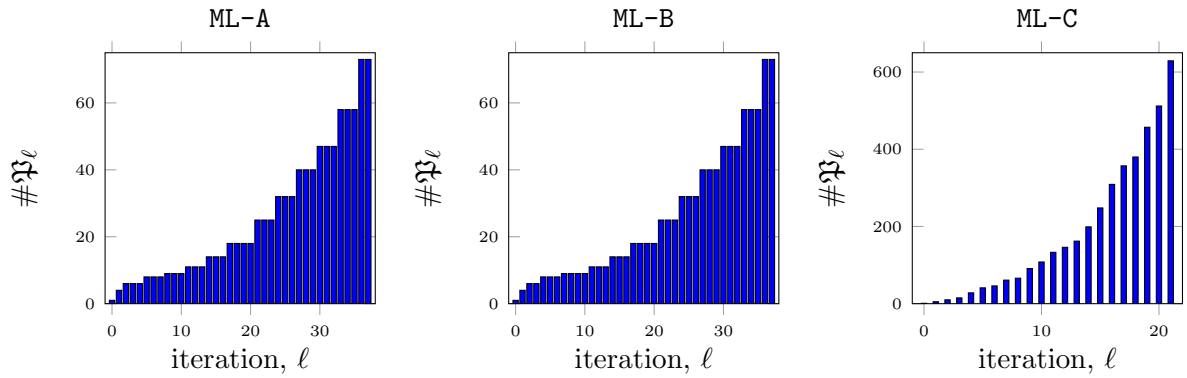


Figure 7: Experiments in subsection 7.2: Evolution of the cardinality of the index set $\mathfrak{P}_\ell$.

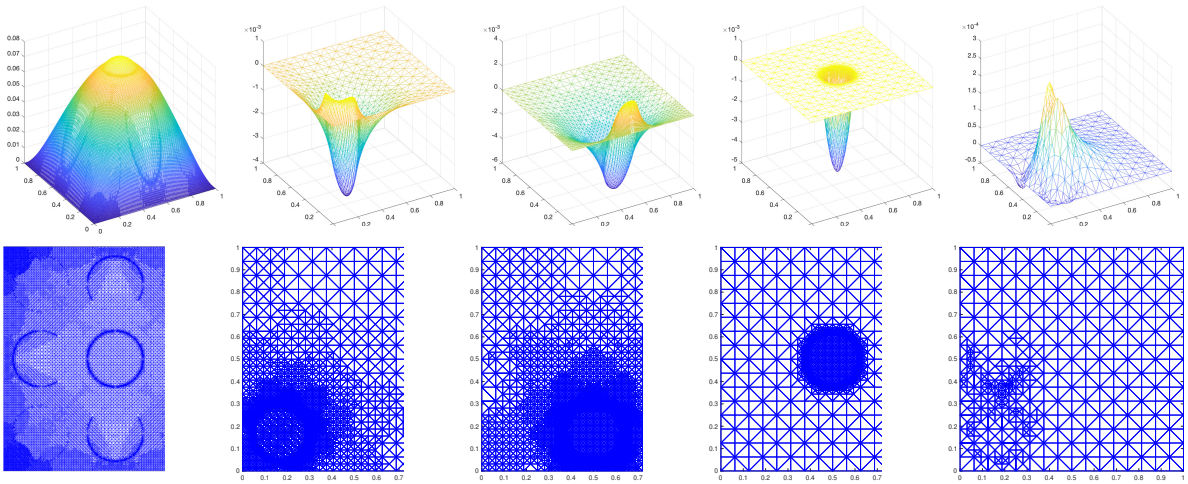


Figure 8: Experiments in subsection 7.2: Coefficients $u_{\ell\nu} \in \mathbb{X}_{\ell\nu} = \mathcal{S}^1(\mathcal{T}_{\ell\nu})$ of an intermediate SGFEM approximation ($\ell = 16$) generated by `ML-C` (top plots) and the associated adaptively refined meshes $\mathcal{T}_{\ell\nu}$ (bottom plots) for five indices $\nu \in \mathfrak{P}_\ell$.

lead to a decay of the error estimate with fully optimal rate $\mathcal{O}(N_\ell^{-0.5})$ also for `ML-A` and `ML-B`. In Figure 9, we compare the decay of the error estimates τ_ℓ obtained for $\vartheta = 1, 2, 4, 8$. We observe that each choice $\vartheta > 1$ leads to a significant improvement of the convergence rate, which is optimal for $\vartheta = 4, 8$. This behavior is in agreement with the results obtained for `ML-C` presented in Figure 3 and Figure 6, where we see that the combined marking strategy automatically favors parametric enrichments over spatial refinements.

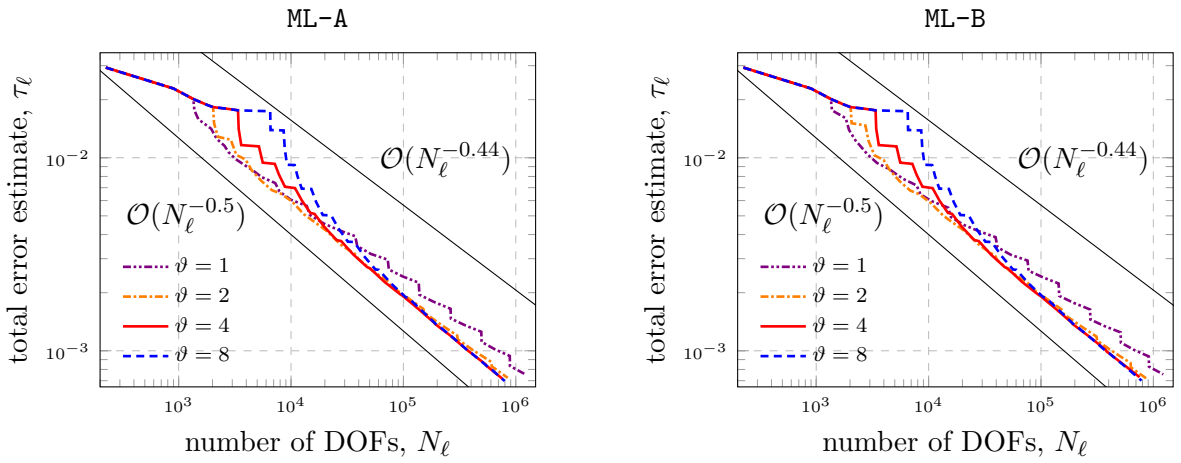


Figure 9: Experiments in subsection 7.2: Total error estimates τ_ℓ versus the number of degrees of freedom N_ℓ for all `ML-A` and `ML-B` and different values of ϑ.

7.3. Conclusions on numerical experiments. Overall, the reported results of numerical experiments indicate that:

- the proposed error estimation strategy in the context of the multilevel SGFEM is as effective as the error estimators for single-level and multilevel SGFEMs investigated in [5] and [16], respectively;
- for the considered test problems, adaptive multilevel SGFEM outperforms its single-level counterpart in terms of convergence rates and in terms of the number of degrees of freedom required to reach the prescribed tolerance; this is a consequence of a greater flexibility of the multilevel SGFEM in allocating degrees of freedom compared to the single-level SGFEM;
- the error estimates for multilevel SFGEM approximations generated by the algorithm with combined marking/enrichment (Algorithm 7.C) decay with the optimal rate; on the other hand, the optimal decay rate for approximations generated by the algorithms with separate marking/enrichment (Algorithm 7.A and Algorithm 7.B) can be ensured by prioritizing parametric enrichments over spatial refinements (by setting $\vartheta > 1$ in the associated marking criterion);
- all adaptive algorithms proposed in this paper are effective in identifying the most 'important' modes in the gPC expansion of the solution to the parametric problem, including the case of infinitely many parameters (as in the test problem in subsection 7.1) and the case when 'importance' of parameters cannot be directly inferred from the ordering of terms in the coefficient expansion (as in the test problem in subsection 7.2).

The application of our algorithms to other classes of parametric PDE problems (e.g., the problems with non-affine coefficient expansions in terms of a finite number of bounded parameters) is possible (see, e.g., [11] for adaptive single-level SGFEM). However, for more challenging problems (e.g., the problems with lognormal parametric coefficients), the efficiency of the algorithms will significantly benefit from combining adaptivity with compression techniques (e.g., low-rank tensor methods [17]), as developed recently in [21] in the context of

the single-level SGFEM. The extension of this methodology to adaptive multilevel SGFEM approximations will be considered in future research.

Acknowledgments. The authors are grateful to David Silvester (University of Manchester) for useful discussions and advice on the implementation of iterative solvers in multilevel stochastic Galerkin FEM.

REFERENCES

[1] J. Bäck, F. Nobile, L. Tamellini, and 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 Lect. Notes Comput. Sci. Eng., Springer, Heidelberg, 2011, pp. 43–62, https://doi.org/10.1007/978-3-642-15337-2_3.

[2] J. Ballani and L. Grasedyck, *Hierarchical tensor approximation of output quantities of parameter-dependent PDEs*, SIAM/ASA J. Uncertain. Quantif., 3 (2015), pp. 852–872, https://doi.org/10.1137/140960980.

[3] A. Bespalov, D. Loghin, and R. Youngnoi, *Truncation preconditioners for stochastic Galerkin finite element discretizations*, SIAM J. Sci. Comput., (2021). (to appear; available as preprint at arXiv:2006.06428).

[4] A. Bespalov, C. E. Powell, and D. Silvester, *Energy norm a posteriori error estimation for parametric operator equations*, SIAM J. Sci. Comput., 36 (2014), pp. A339–A363, https://doi.org/10.1137/130916849, http://dx.doi.org/10.1137/130916849.

[5] A. Bespalov, D. Praetorius, L. Rocchi, and M. Ruggeri, *Convergence of adaptive stochastic Galerkin FEM*, SIAM J. Numer. Anal., 57 (2019), pp. 2359–2382, https://doi.org/10.1137/18M1229560, https://doi.org/10.1137/18M1229560.

[6] A. Bespalov, D. Praetorius, L. Rocchi, and M. Ruggeri, *Goal-oriented error estimation and adaptivity for elliptic PDEs with parametric or uncertain inputs*, Comput. Methods Appl. Mech. Engrg., 345 (2019), pp. 951–982, https://doi.org/10.1016/j.cma.2018.10.041, https://doi.org/10.1016/j.cma.2018.10.041.

[7] A. Bespalov and L. Rocchi, *Efficient adaptive algorithms for elliptic PDEs with random data*, SIAM/ASA J. Uncertain. Quantif., 6 (2018), pp. 243–272, https://doi.org/10.1137/17M1139928, https://doi.org/10.1137/17M1139928.

[8] A. Bespalov and L. Rocchi, *Stochastic T-IFISS*, February 2019. Available online at http://web.mat.bham.ac.uk/A.Bespalov/software/index.html#stoch_tifiss.

[9] A. Bespalov, L. Rocchi, and D. Silvester, *T-IFISS: a toolbox for adaptive FEM computation*, Comput. Math. Appl., 81 (2021), pp. 373–390, https://doi.org/10.1016/j.camwa.2020.03.005.

[10] A. Bespalov and D. Silvester, *Efficient adaptive stochastic Galerkin methods for parametric operator equations*, SIAM J. Sci. Comput., 38 (2016), pp. A2118–A2140, https://doi.org/10.1137/15M1027048, http://dx.doi.org/10.1137/15M1027048.

[11] A. Bespalov and F. Xu, *A posteriori error estimation and adaptivity in stochastic Galerkin FEM for parametric elliptic PDEs: beyond the affine case*, Comput. Math. Appl., 80 (2020), pp. 1084–1103, https://doi.org/10.1016/j.camwa.2020.05.023, https://doi.org/10.1016/j.camwa.2020.05.023.

[12] F. A. Bornemann, B. Erdmann, and R. Kornhuber, *A posteriori error estimates for elliptic problems in two and three space dimensions*, SIAM J. Numer. Anal., 33 (1996), pp. 1188–1204, https://doi.org/10.1137/0733059, https://doi.org/10.1137/0733059.

[13] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup, *Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients*, Comput. Vis. Sci., 14 (2011), pp. 3–15, https://doi.org/10.1007/s00791-011-0160-x.

[14] A. Cohen, R. DeVore, and C. Schwab, *Convergence rates of best N-term Galerkin approximations for a class of elliptic sPDEs*, Found. Comput. Math., 10 (2010), pp. 615–646, https://doi.org/10.1007/s10208-010-9072-2.

[15] A. Cohen, R. DeVore, and C. Schwab, *Analytic regularity and polynomial approximation of parametric*

and stochastic elliptic PDE's, Anal. Appl., 9 (2011), pp. 11–47.

[16] A. J. Crowder, C. E. Powell, and A. Bespalov, *Efficient adaptive multilevel stochastic Galerkin approximation using implicit a posteriori error estimation*, SIAM J. Sci. Comput., 41 (2019), pp. A1681–A1705.

[17] S. Dolgov, B. N. Khoromskij, A. Litvinenko, and H. G. Matthies, *Polynomial chaos expansion of random coefficients and the solution of stochastic partial differential equations in the tensor train format*, SIAM/ASA J. Uncertain. Quantif., 3 (2015), pp. 1109–1135, https://doi.org/10.1137/140972536.

[18] W. Dörfler, *A convergent adaptive algorithm for Poisson's equation*, SIAM J. Numer. Anal., 33 (1996), pp. 1106–1124, https://doi.org/10.1137/0733054, https://doi.org/10.1137/0733054.

[19] M. Eigel, C. J. Gittelson, C. Schwab, and E. Zander, *Adaptive stochastic Galerkin FEM*, Comput. Methods Appl. Mech. Engrg., 270 (2014), pp. 247–269, https://doi.org/10.1016/j.cma.2013.11.015, http://dx.doi.org/10.1016/j.cma.2013.11.015.

[20] M. Eigel, C. J. Gittelson, C. Schwab, and E. Zander, *A convergent adaptive stochastic Galerkin finite element method with quasi-optimal spatial meshes*, ESAIM Math. Model. Numer. Anal., 49 (2015), pp. 1367–1398, https://doi.org/10.1051/m2an/2015017, http://dx.doi.org/10.1051/m2an/2015017.

[21] M. Eigel, M. Marschall, M. Pfeffer, and R. Schneider, *Adaptive stochastic Galerkin FEM for lognormal coefficients in hierarchical tensor representations*, Numer. Math., 145 (2020), pp. 655–692, https://doi.org/10.1007/s00211-020-01123-1, https://doi.org/10.1007/s00211-020-01123-1.

[22] M. Eigel and C. Merdon, *Local equilibration error estimators for guaranteed error control in adaptive stochastic higher-order Galerkin finite element methods*, SIAM/ASA J. Uncertain. Quantif., 4 (2016), pp. 1372–1397, https://doi.org/10.1137/15M102188X, http://dx.doi.org/10.1137/15M102188X.

[23] M. Eigel, J. Neumann, R. Schneider, and S. Wolf, *Non-intrusive tensor reconstruction for high-dimensional random PDEs*, Comput. Meth. Appl. Mat., 19 (2019), pp. 39–53.

[24] M. Eigel and E. Zander, *ALEA – A python framework for spectral methods and low-rank approximations in uncertainty quantification*. https://bitbucket.org/aleadev/alea.

[25] C. Erath, G. Gantner, and D. Praetorius, *Optimal convergence behavior of adaptive FEM driven by simple $(h - h/2)$-type error estimators*, Comput. Math. Appl., 79 (2020), pp. 623–642, https://doi.org/10.1016/j.camwa.2019.07.014, https://doi.org/10.1016/j.camwa.2019.07.014.

[26] O. G. Ernst and E. Ullmann, *Stochastic Galerkin matrices*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 1848–1872, https://doi.org/10.1137/080742282.

[27] M. Espig, W. Hackbusch, A. Litvinenko, H. G. Matthies, and P. Wähnert, *Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats*, Comput. Math. Appl., 67 (2014), pp. 818–829, https://doi.org/10.1016/j.camwa.2012.10.008.

[28] R. G. Ghanem and R. M. Kruger, *Numerical solution of spectral stochastic finite element systems*, Comput. Methods Appl. Mech. Engrg., 129 (1996), pp. 289–303, https://doi.org/10.1016/0045-7825(95)00909-4, https://doi.org/10.1016/0045-7825(95)00909-4.

[29] L. Giraldi, A. Litvinenko, D. Liu, H. G. Matthies, and A. Nouy, *To be or not to be intrusive? The solution of parametric and stochastic equations—the "plain vanilla" Galerkin case*, SIAM J. Sci. Comput., 36 (2014), pp. A2720–A2744, https://doi.org/10.1137/130942802.

[30] C. J. Gittelson, *An adaptive stochastic Galerkin method for random elliptic operators*, Math. Comp., 82 (2013), pp. 1515–1541, https://doi.org/10.1090/S0025-5718-2013-02654-3.

[31] C. J. Gittelson, *Convergence rates of multilevel and sparse tensor approximations for a random elliptic PDE*, SIAM J. Numer. Anal., 51 (2013), pp. 2426–2447, https://doi.org/10.1137/110826539, https://doi.org/10.1137/110826539.

[32] M. D. Gunzburger, C. G. Webster, and G. Zhang, *Stochastic finite element methods for partial differential equations with random input data*, Acta Numer., 23 (2014), pp. 521–650, https://doi.org/10.1017/S0962492914000075.

[33] M. Karkulik, D. Pavlicek, and D. Praetorius, *On 2D newest vertex bisection: Optimality of mesh-closure and H^1-stability of L_2-projection*, Constr. Approx., 38 (2013), pp. 213–234.

[34] R. Kornhuber and E. Youett, *Adaptive multilevel Monte Carlo methods for stochastic variational inequalities*, SIAM J. Numer. Anal., 56 (2018), pp. 1987–2007, https://doi.org/10.1137/16M1104986, https://doi.org/10.1137/16M1104986.

[35] F. Y. Kuo, C. Schwab, and I. H. Sloan, *Multi-level quasi-Monte Carlo finite element methods for a class of elliptic PDEs with random coefficients*, Found. Comput. Math., 15 (2015), pp. 411–449,

https://doi.org/10.1007/s10208-014-9237-5.

[36] J. Lang, R. Scheichl, and D. Silvester, *A fully adaptive multilevel stochastic collocation strategy for solving elliptic PDEs with random data*, J. Comput. Phys., 419 (2020), pp. 109692, 17, https://doi.org/10.1016/j.jcp.2020.109692, https://doi.org/10.1016/j.jcp.2020.109692.

[37] G. J. Lord, C. E. Powell, and T. Shardlow, *An introduction to computational stochastic PDEs*, Cambridge Texts in Applied Mathematics, Cambridge University Press, 2014, https://doi.org/10.1017/CBO9781139017329.

[38] C. E. Powell and H. C. Elman, *Block-diagonal preconditioning for spectral stochastic finite-element systems*, IMA J. Numer. Anal., 29 (2009), pp. 350–375, https://doi.org/10.1093/imanum/drn014, https://doi.org/10.1093/imanum/drn014.

[39] D. Praetorius, M. Ruggeri, and E. P. Stephan, *The saturation assumption yields optimal convergence of two-level adaptive BEM*, Appl. Numer. Math., 152 (2020), pp. 105–124.

[40] C. Schwab and C. J. Gittelson, *Sparse tensor discretizations of high-dimensional parametric and stochastic PDEs*, Acta Numer., 20 (2011), pp. 291–467, https://doi.org/10.1017/S0962492911000055, http://dx.doi.org/10.1017/S0962492911000055.

[41] D. J. Silvester, A. Bespalov, Q. Liao, and L. Rocchi, *Triangular IFISS (T-IFISS)*. Available online at http://www.manchester.ac.uk/ifiss/tifiss, February 2019.

[42] D. J. Silvester and V. Simoncini, *An optimal iterative solver for symmetric indefinite systems stemming from mixed approximation*, ACM Trans. Math. Software, 37 (2011), pp. 42/1–42/22, https://doi.org/10.1145/1916461.1916466, https://doi.org/10.1145/1916461.1916466.

[43] B. Sousedík and R. G. Ghanem, *Truncated hierarchical preconditioning for the stochastic Galerkin FEM*, Int. J. Uncertain. Quantif., 4 (2014), pp. 333–348, https://doi.org/10.1615/Int.J.UncertaintyQuantification.2014007353.

[44] R. Stevenson, *The completion of locally refined simplicial partitions created by bisection*, Math. Comp., 77 (2008), pp. 227–241, https://doi.org/10.1090/S0025-5718-07-01959-X, http://dx.doi.org/10.1090/S0025-5718-07-01959-X.

[45] A. L. Teckentrup, P. Jantsch, C. G. Webster, and M. Gunzburger, *A multilevel stochastic collocation method for partial differential equations with random input data*, SIAM/ASA J. Uncertain. Quantif., 3 (2015), pp. 1046–1074, https://doi.org/10.1137/140969002.

[46] E. Ullmann, *A Kronecker product preconditioner for stochastic Galerkin finite element discretizations*, SIAM J. Sci. Comput., 32 (2010), pp. 923–946, https://doi.org/10.1137/080742853.