EFFICIENT ADAPTIVE MULTILEVEL STOCHASTIC GALERKIN APPROXIMATION USING IMPLICIT A POSTERIORI ERROR ESTIMATION*

A.J. CROWDER†, C.E. POWELL‡, AND A. BESPALOV§

Abstract. Partial differential equations (PDEs) with inputs that depend on infinitely many parameters pose serious theoretical and computational challenges. Sophisticated numerical algorithms that automatically determine which parameters need to be activated in the approximation space in order to estimate a quantity of interest to a prescribed error tolerance are needed. For elliptic PDEs with parameter-dependent coefficients, stochastic Galerkin finite element methods (SGFEMs) have been well studied. Under certain assumptions, it can be shown that there exists a sequence of SGFEM approximation spaces for which the energy norm of the error decays to zero at a rate that is independent of the number of input parameters. However, it is not clear how to adaptively construct these spaces in a practical and computationally efficient way. We present a new adaptive SGFEM algorithm that tackles elliptic PDEs with parameter-dependent coefficients quickly and efficiently. We consider approximation spaces with a multilevel structure—where each solution mode is associated with a finite element space on a potentially different mesh—and use an implicit a posteriori error estimation strategy to steer the adaptive enrichment of the space. At each step, the components of the error estimator are used to assess the potential benefits of a variety of enrichment strategies, including whether or not to activate more parameters. No marking or tuning parameters are required. Numerical experiments for a selection of test problems demonstrate that the new method performs optimally in that it generates a sequence of approximations for which the estimated energy error decays to zero at the same rate as the error for the underlying finite element method applied to the associated parameter-free problem.

Key words. adaptivity, finite element methods, stochastic Galerkin approximation, multilevel methods, a posteriori error estimation.

AMS subject classifications. 35R60 , 60H35, 65N30, 65F10

1. Introduction. In many engineering and other real world applications, we frequently encounter models consisting of partial differential equations (PDEs) which have uncertain or parameter-dependent inputs. When the solutions are sufficiently smooth with respect to these parameters, it is known that stochastic Galerkin finite element methods (SGFEMs) [21, 15, 2], also known as intrusive polynomial chaos methods in the statistics and engineering communities, offer a powerful alternative to brute force sampling methods for propagating uncertainty to the model outputs. When the number of input parameters in the PDE model is countably infinite (which may arise, for example, if we represent an uncertain spatially varying coefficient as a Karhunen-Loève expansion), then we encounter significant theoretical and numerical challenges. In general, it is not known a priori which parameters need to be incorporated into discretisations of the model in order to estimate specific quantities of interest to a prescribed error tolerance. Ad hoc selection of a finite subset of parameters prior to applying a standard SGFEM is computationally convenient, but may lead to inaccurate results with no guaranteed error bounds. In this work we consider the steady-state diffusion problem with a spatially varying coefficient that depends on infinitely many parameters, and develop a computationally efficient multilevel SGFEM which uses an a posteriori error estimator to adaptively construct appropriate approximation spaces.

Let the spatial domain $D \subset \mathbb{R}^2$ be bounded with a Lipschitz polygonal boundary ∂D and let

*This work was supported by EPSRC grants EP/P013317/1 and EP/P013791/1. The second author would like to thank the Isaac Newton Institute for Mathematical Sciences, Cambridge, for support and hospitality during the Uncertainty Quantification programme as well as the Simons Foundation. This work was partially also supported by EPSRC grant no EP/K032208/1.

†School of Mathematics, University of Manchester, Oxford Road, Manchester M13 9PL, United Kingdom (adam.crowder@postgrad.manchester.ac.uk).

‡School of Mathematics, University of Manchester, Oxford Road, Manchester M13 9PL, United Kingdom (c.powell@manchester.ac.uk).

§School of Mathematics, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom (a.bespalov@bham.ac.uk).

$y_1, y_2, \ldots$ be a countable sequence of parameters with $y_m \in \Gamma_m = [-1, 1]$, for $m \in \mathbb{N}$. We consider the parametric diffusion problem: find $u(\mathbf{x}, \mathbf{y}) : D \times \Gamma \to \mathbb{R}$ that satisfies

$$-\nabla \cdot (a(\mathbf{x}, \mathbf{y}) \nabla u(\mathbf{x}, \mathbf{y})) = f(\mathbf{x}), \qquad \mathbf{x} \in D,\ \mathbf{y} \in \Gamma, \tag{1.1}$$

$$u(\mathbf{x}, \mathbf{y}) = 0, \qquad \mathbf{x} \in \partial D,\ \mathbf{y} \in \Gamma. \tag{1.2}$$

Here, $\mathbf{y} = [y_1, y_2, \ldots]^\top \in \Gamma$ where $\Gamma = \Pi_{m=1}^{\infty} \Gamma_m$ is the parameter domain. The coefficient $a(\mathbf{x}, \mathbf{y})$ should be positive and bounded on $D \times \Gamma$. We also make the following important assumption.

ASSUMPTION 1.1. The coefficient $a(\mathbf{x}, \mathbf{y})$ admits the decomposition

$$a(\mathbf{x}, \mathbf{y}) = a_0(\mathbf{x}) + \sum_{m=1}^{\infty} a_m(\mathbf{x}) y_m, \tag{1.3}$$

with $a_0(\mathbf{x}),\ a_m(\mathbf{x}) \in L^\infty(D)$ and $||a_m||_{L^\infty(D)} \to 0$ sufficiently quickly as $m \to \infty$ so that

$$\sum_{m=1}^{\infty} ||a_m||_{L^\infty(D)} < \operatorname*{ess\,inf}_{\mathbf{x} \in D} a_0(\mathbf{x}). \tag{1.4}$$

Note that (1.4) helps to ensure the well-posedness of the weak formulation of (1.1)–(1.2). This will be made more rigorous in the next section.

Standard SGFEMs seek approximations to $u(\mathbf{x}, \mathbf{y})$ in (1.1)–(1.2) in a tensor product space X of the form

$$X := H_1 \otimes P, \qquad H_1 := \operatorname{span}\{\phi_i(\mathbf{x})\}_{i=1}^{n}, \qquad P := \operatorname{span}\{\psi_j(\mathbf{y})\}_{j=1}^{s}, \tag{1.5}$$

where H_1 is a finite element space associated with a mesh $\mathcal{T}_h$ on the spatial domain D and P is a set of polynomials on the parameter domain Γ in a *finite* number (say, M) of the parameters y_m. In this case, $u_X \in X$ admits the decomposition

$$u_X(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{s} u_j(\mathbf{x}) \psi_j(\mathbf{y}), \qquad u_j \in H_1.$$

We use the term 'single-level' approximation to mean that X is defined as in (1.5). Here, each coefficient u_j is associated with the *same* finite element space H_1. In contrast, we will work with spaces X which have a 'multilevel' structure, by which we mean that the coefficients u_j may each reside in a *different* finite element space. These finite element spaces will be associated with a sequence of meshes which each have a different 'level' number.

Handling inputs of the form (1.3) is a non-trivial task. Suppose we truncate $a(\mathbf{x}, \mathbf{y})$ in (1.3) after M terms (assuming that $||a_m||_\infty \geq ||a_{m+1}||_\infty$) and define X as in (1.5), where $\mathbf{y} = [y_1, \ldots, y_M]^\top$. A priori error estimates provided in [2] reveal that the rate of convergence of standard SGFEMs deteriorates as $M \to \infty$. This phenomenon is referred to as the *curse of dimensionality*. Many recent works provide a priori error analysis for more sophisticated SGFEMs in the case where we have infinitely many parameters. For example, see [30, 8, 7, 12, 13, 23, 10]. In each of these works, the decay rate, or equivalently, the summability of the sequence $\{\|a_m\|_\infty\}_{m=1}^{\infty}$ plays an important role. Various theoretical results have been established proving the existence of a sequence of SGFEM approximation spaces $X^0, X^1, \ldots$, such that the energy norm of the error decays to zero at a rate that is independent of the number of parameters, as $N_{\text{dof}} = \dim(X) \to \infty$. These results all assume that X has a more complex structure than in (1.5) but demonstrate that SGFEMs *can* be immune to the curse of dimensionality if implemented in the right way.

In [12, 13, 23] a multilevel structure is imposed on X. Theoretical results show that if $\|a_m\|_\infty \to 0$ fast enough, then there exists a sequence of multilevel spaces for which the error decays to zero at the rate afforded to the chosen finite element method for the parameter-free analogue of (1.1)–(1.2). Given a sequence of finite element spaces (with different level numbers), we use an implicit a posteriori error estimation scheme to design an appriopriate sequence of multilevel SGFEM spaces. By implicit, we mean that the approach uses the residual associated with the SGFEM solution indirectly and requires the solution of additional problems. Starting with an initial low-dimensional space X^0, the resulting energy error is estimated. The components of the error estimator are then examined to steer the enrichment of X^0. Adaptive schemes have also been proposed in [18, 22, 17, 19], but using an explicit error estimation strategy which uses the residual directly. Explicit error estimators often lead to less favourable effectivity indices than implicit schemes. Moreover, the algorithms presented in [18, 22, 17, 19] all rely on a Dörfler-like marking strategy [16], and require the selection of multiple tuning or marking parameters. The optimal selection of these is unclear, however, and is problem-dependent. The authors of [4, 6, 28, 5] consider single-level approximation spaces and implement an implicit error estimation strategy. We revisit [4, 6], extend the error estimation strategy considered there to the more complex multilevel setting, and use this to design an accurate and efficient adaptive multilevel SGFEM algorithm.

1.1. Outline. In Section 2 we introduce the weak formulation of (1.1)–(1.2) and review conditions for well-posedness. In Section 3 we describe the multilevel construction of SGFEM approximation spaces and give practical information about how to assemble the matrices associated with the discrete problem in a computationally efficient way. In Section 4 we extend the implicit energy norm a posteriori error estimation strategy developed in [4, 6] for SGFEM approximation spaces X of the form (1.5) to the multilevel setting. In Section 5 we introduce a new adaptive algorithm that uses the error estimation strategy from Section 4 to design problem-dependent multilevel SGFEM approximation spaces. Numerical results are presented in Section 6.

2. Weak Formulation of the Parametric Diffusion Problem. We assume that $y_m \in \Gamma_m := [-1, 1]$ for each $m \in \mathbb{N}$ and that π_m is a measure on $(\Gamma_m, \mathcal{B}(\Gamma_m))$, where $\mathcal{B}(\Gamma_m)$ denotes the Borel σ–algebra on Γ_m. We also assume that

$$\int_{\Gamma_m} y_m \, d\pi_m(y_m) = 0, \quad m \in \mathbb{N}. \tag{2.1}$$

For instance, this is true when y_m is the image of a mean zero random variable and π_m is the associated probability measure. We assume that y_m is the image of a uniform random variable $\xi_m \sim U([-1, 1])$ and so the associated probability measure π_m has density $\rho_m = 1/2$ with respect to Lebesgue measure. We now define the parameter domain $\Gamma = \Pi_{m=1}^{\infty} \Gamma_m$ and the product measure

$$\pi(\mathbf{y}) := \prod_{m=1}^{\infty} \pi_m(y_m).$$

If the parameters y_m are images of independent random variables then the associated probability measure has this separable form.

We are interested in Galerkin approximations of u satisfying (1.1)–(1.2) and thus start by considering its variational formulation:

$$\text{find } u \in V := L^2_\pi(\Gamma, H^1_0(D)) : \quad B(u, v) = F(v), \quad \text{for all } v \in V. \tag{2.2}$$

Here, $H^1_0(D)$ is the usual Hilbert space of functions that vanish on ∂D in the sense of trace and $L^2_\pi(\Gamma)$ is the space of functions that are square integrable with respect to $\pi(\mathbf{y})$ on Γ. That is,

$$L^2_\pi(\Gamma) := \left\{ v(\mathbf{y}) \mid \langle v, v \rangle_{L^2_\pi(\Gamma)} = \int_\Gamma v(\mathbf{y})^2 \, d\pi(\mathbf{y}) < \infty \right\}.$$

The space V is equipped with the norm $||\cdot||_V$, where

$$||v||_V = \left(\int_\Gamma ||v(\cdot,\mathbf{y})||^2_{H^1_0(D)} \, d\pi(\mathbf{y}) \right)^{\frac{1}{2}},$$

and $||v||_{H^1_0(D)} = ||\nabla v||_{L^2(D)}$ for all $v \in H^1_0(D)$. The bilinear form $B : V \times V \to \mathbb{R}$ and the linear functional $F : V \to \mathbb{R}$ are defined by

$$B(u,v) = \int_\Gamma \int_D a(\mathbf{x},\mathbf{y}) \nabla u(\mathbf{x},\mathbf{y}) \cdot \nabla v(\mathbf{x},\mathbf{y}) \, d\mathbf{x} \, d\pi(\mathbf{y}), \tag{2.3}$$

$$F(v) = \int_\Gamma \int_D f(\mathbf{x}) v(\mathbf{x},\mathbf{y}) \, d\mathbf{x} \, d\pi(\mathbf{y}). \tag{2.4}$$

To ensure that (2.2) is well-posed, $B(\cdot,\cdot)$ must be bounded and coercive over V. This is ensured by the following assumption.

ASSUMPTION 2.1. There exist real positive constants $a_{\min}$ and $a_{\max}$ such that

$$0 < a_{\min} \leq a(\mathbf{x},\mathbf{y}) \leq a_{\max} < \infty, \quad a.e. \text{ in } D \times \Gamma.$$

Note that (1.4) is a sufficient condition for Assumption 2.1 to hold. If Assumption 2.1 holds, the bilinear form (2.3) induces a norm (the so-called energy norm),

$$||v||_B = B(v,v)^{1/2}, \qquad \text{for all } v \in V.$$

In addition, to ensure that $F(\cdot)$ is bounded on V we assume $f(\mathbf{x}) \in L^2(D)$. We will also make the following assumption.

ASSUMPTION 2.2. There exist real positive constants $a^0_{\min}$ and $a^0_{\max}$ such that

$$0 < a^0_{\min} \leq a_0(\mathbf{x}) \leq a^0_{\max} < \infty, \quad a.e. \text{ in } D.$$

Due to (1.3), we have the decomposition,

$$B(u,v) = B_0(u,v) + \sum_{m=1}^{\infty} B_m(u,v), \quad \text{for all } u,v \in V, \tag{2.5}$$

where the component bilinear forms are given by

$$B_0(u,v) = \int_\Gamma \int_D a_0(\mathbf{x}) \nabla u(\mathbf{x},\mathbf{y}) \cdot \nabla v(\mathbf{x},\mathbf{y}) \, d\mathbf{x} \, d\pi(\mathbf{y}), \tag{2.6}$$

$$B_m(u,v) = \int_\Gamma \int_D a_m y_m(\mathbf{x}) \nabla u(\mathbf{x},\mathbf{y}) \cdot \nabla v(\mathbf{x},\mathbf{y}) \, d\mathbf{x} \, d\pi(\mathbf{y}). \tag{2.7}$$

If Assumption 2.2 holds, the bilinear form (2.6) also induces the norm $||v||_{B_0} = B_0(v,v)^{1/2}$ on V, associated with the coefficient a_0. It is then straightforward to show that

$$\lambda ||v||^2_B \leq ||v||^2_{B_0} \leq \Lambda ||v||^2_B, \quad \text{for all } v \in V,$$

where $0 < \lambda < 1 < \Lambda < \infty$ and

$$\lambda := a^0_{\min} a^{-1}_{\max}, \qquad \lambda := a^0_{\max} a^{-1}_{\min}, \tag{2.8}$$

and so the norms $||\cdot||_B$ and $||\cdot||_{B_0}$ are equivalent.

3. Multilevel SGFEM Approximation. We can compute a Galerkin approximation to $u \in V$ by projecting (2.2) onto a finite-dimensional subspace $X \subset V$. The best known rates of convergence with respect to $N_{\rm dof} = \dim(X)$ (see [10, 12, 13, 23]) are achieved for approximation spaces that have a multilevel structure, which we now describe. As usual, we exploit the fact that $V \cong H_0^1(D) \otimes L_\pi^2(\Gamma)$ and construct X by tensorising separate subspaces of $H_0^1(D)$ and $L_\pi^2(\Gamma)$.

For the parameter domain, we first introduce families of univariate polynomials $\{\psi_n(y_m)\}_{n\in\mathbb{N}_0}$ on Γ_m for each $m = 1, 2, \dots$ that are orthonormal with respect to the inner product

$$\langle v, w\rangle_{L^2_{\pi_m}(\Gamma_m)} = \int_{\Gamma_m} v(y_m)w(y_m)d\pi_m(y_m).$$

Here, n denotes the polynomial degree and $\psi_0(y_m) = 1$. Now we define the set of finitely supported multi-indices $J := \{\mu = (\mu_1, \mu_2, \dots) \in \mathbb{N}_0^{\mathbb{N}};\ \#{\rm supp}(\mu) < \infty\}$ where ${\rm supp}(\mu) := \{m \in \mathbb{N};\ \mu_m \neq 0\}$ and consider multivariate tensor product polynomials of the form

$$\psi_\mu(\mathbf{y}) = \prod_{m=1}^{\infty} \psi_{\mu_m}(y_m) = \prod_{m\in{\rm supp}(\mu)} \psi_{\mu_m}(y_m), \quad \mu \in J. \tag{3.1}$$

The countable set $\{\psi_\mu(\mathbf{y})\}_{\mu\in J}$ is an orthonormal basis of $L_\pi^2(\Gamma)$ with respect to the inner product $\langle\cdot,\cdot\rangle_{L_\pi^2(\Gamma)}$. Orthonormality comes from the separability of $\pi(\mathbf{y})$ and the construction (3.1) since

$$\langle\psi_\mu(\mathbf{y}), \psi_\nu(\mathbf{y})\rangle_{L_\pi^2(\Gamma)} = \prod_{m=1}^{\infty} \langle\psi_{\mu_m}(y_m), \psi_{\nu_m}(y_m)\rangle_{L^2_{\pi_m}(\Gamma_m)} = \prod_{m=1}^{\infty} \delta_{\mu_m\nu_m} = \delta_{\mu\nu}, \tag{3.2}$$

for all $\mu, \nu \in J$. Now, given any finite set $J_P \subset J$ (which we assume always contains the multi-index $\mu = (0, 0, \dots)$) we can construct a finite-dimensional set $P := \{\psi_\mu(\mathbf{y}), \mu \in J_P\} \subset L_\pi^2(\Gamma)$ of multivariate polynomials on Γ. Note that we can also write

$$P = \bigoplus_{\mu\in J_P} P^\mu, \quad P^\mu = {\rm span}\{\psi_\mu(\mathbf{y})\}, \quad \mu \in J_P.$$

Given a set of multi-indices J_P, we will construct approximation spaces of the form

$$X := \bigoplus_{\mu\in J_P} X^\mu := \bigoplus_{\mu\in J_P} H_1^\mu \otimes P^\mu \subset V, \tag{3.3}$$

where each $H_1^\mu \subset H_0^1(D)$ is a finite element space associated with the spatial domain D and

$$H_1^\mu := {\rm span}\Big\{\phi_i^\mu(\mathbf{x});\ i = 1, 2, \dots, N_1^\mu\Big\}, \quad \text{for all } \mu \in J_P.$$

For each $\mu \in J_P$ we may use a potentially different space H_1^μ. Compare X in (3.3) to X in (1.5). The latter can be written as $X := \oplus_{\mu\in J_P} H_1 \otimes P^\mu$. To work with spaces of the form (3.3), we need to select an appropriate set $\mathbf{H}_1 := \{H_1^\mu\}_{\mu\in J_P}$ of finite element spaces. To this end, we assume that we can construct a nested sequence of meshes $\mathcal{T}_i$, $i = 0, 1, \dots$ (of rectangular or triangular elements) that give rise to a sequence of conforming finite element spaces $H^{(0)} \subset H^{(1)} \subset \cdots H^{(i)} \cdots \subset H_0^1(D)$. In this setting, the index i denotes the mesh 'level number'. We will assume that the polynomial degree is fixed in the definition of the finite element spaces, and only the mesh is changing as we change the level. If $j > i$, then $\mathcal{T}_j$ can be obtained from $\mathcal{T}_i$ by one or more mesh refinements.

For notational convenience, we collect the meshes into a set

$$\boldsymbol{\mathcal{T}} := \big\{\mathcal{T}_i;\ i = 0, 1, 2, \dots\big\}. \tag{3.4}$$

For each $\mu \in J_P$, the space H_1^μ is constructed using one of the meshes from $\boldsymbol{\mathcal{T}}$. That is, to each $\mu \in J_P$ we assign a mesh level number $\ell^\mu = i$ (for some $i \in \mathbb{N}_0$) and set $H_1^\mu = H^{(i)}$. If $\ell^\mu = \ell^\nu$ for some $\mu, \nu \in J_P$, then $H_1^\mu = H_1^\nu$. We collect the chosen levels ℓ^μ in the set $\boldsymbol{\ell} := \{\ell^\mu\}_{\mu \in J_P}$. Now, any space X of the form (3.3) is determined by choosing a finite set J_P of multi-indices and a set $\boldsymbol{\ell}$ of associated mesh level numbers. Clearly, $\mathrm{card}(\boldsymbol{\ell}) = \mathrm{card}(J_P) < \infty$.

Once J_P and $\boldsymbol{\ell}$ have been chosen, our SGFEM approximation $u_X \in X$ to $u \in V$ is found by solving the discrete problem:

$$\text{find } u_X \in X: \quad B(u_X, v) = F(v), \quad \text{for all } v \in X. \tag{3.5}$$

For u_X to be computable, it is essential that the sum in (2.5) has a finite number of nonzero terms. Let $M \in \mathbb{N}$ be the smallest integer such that $\mu_m = 0$ for all $m > M$ and for all $\mu \in J_P$. That is, let M be the number of parameters y_m that are 'active' in the definition of J_P. Then, provided (2.1) holds, $B_m(u_X, v) = 0$ for $u_X, v \in X$ for all $m > M$ (e.g. see [4]). In other words, the choice of J_P implicitly truncates the sum after M terms; we do not have to truncate $a(\mathbf{x}, \mathbf{y})$ a priori. Expanding the Galerkin approximation as

$$u_X = \sum_{\mu \in J_P} u_X^\mu(\mathbf{x})\psi_\mu(\mathbf{y}), \qquad u_X^\mu = \sum_{i=1}^{N_1^\mu} u_i^\mu \phi_i^\mu(\mathbf{x}), \qquad u_i^\mu \in \mathbb{R}, \tag{3.6}$$

and taking test functions $v = \psi_\nu(\mathbf{y})\phi_j^\nu(\mathbf{x})$ for all $\nu \in J_P$ and $j = 1, 2, \ldots, N_1^\nu$ yields a system of N_{dof} equations $A\mathbf{u} = \mathbf{b}$ for the unknown coefficients u_i^μ that define u_X, where

$$N_{\mathrm{dof}} = \sum_{\mu \in J_P} \dim(X^\mu) = \sum_{\mu \in J_P} N_1^\mu.$$

If multilevel SGFEMs are to be useful in practice, we have to be able to assemble the components of this linear system and solve it efficiently. We discuss this next.

3.1. Multilevel SGFEM Matrices. The matrix A and the vectors $\mathbf{b}$ and $\mathbf{u}$ each have a block structure, with the blocks indexed by the elements (multi-indices) of J_P, namely

$$\begin{aligned} [A_{\mu\nu}]_{ij} = [A_{\nu\mu}]_{ji} &= B\big(\psi_\mu\phi_i^\mu, \psi_\nu\phi_j^\nu\big) \qquad (A \text{ is symmetric}), \\ [\mathbf{b}_\nu]_j &= F\big(\psi_\nu\phi_j^\nu\big), \\ [\mathbf{u}_\mu]_i &= u_i^\mu, \end{aligned}$$

for $i = 1, 2, \ldots, N_1^\mu$ and $j = 1, 2, \ldots, N_1^\nu$. For single-level methods, the resulting system matrix admits the Kronecker product structure (e.g., see [27]) $K_0 \otimes G_0 + \sum_{m=1}^M K_m \otimes G_m$, where $\{K_m\}_{m=0}^M$ are stiffness matrices associated with the *same* finite element space and

$$[G_0]_{\mu\nu} = [G_0]_{\nu\mu} = \delta_{\nu\mu}, \qquad [G_m]_{\mu\nu} = [G_m]_{\nu\mu} = \int_\Gamma y_m \psi_\mu(\mathbf{y})\psi_\nu(\mathbf{y})\, d\pi(\mathbf{y}), \qquad m = 1, 2, \ldots, M.$$

In the multilevel approach, there is no such Kronecker structure. The $\nu\mu^{\text{th}}$ block of A is given by

$$A_{\nu\mu} = \sum_{m=0}^M [G_m]_{\nu\mu} K_{\nu\mu}^m, \qquad [K_{\nu\mu}^m]_{ji} = \int_D a_m(\mathbf{x}) \nabla\phi_i^\mu(\mathbf{x}) \cdot \nabla\phi_j^\nu(\mathbf{x})\, d\mathbf{x}, \tag{3.7}$$

for $i = 1, 2, \ldots, N_1^\mu$ and $j = 1, 2, \ldots, N_1^\nu$. The entries of the stiffness matrix $K_{\nu\mu}^m$ in (3.7) depend on basis functions associated with a pair of meshes $\mathcal{T}_{\ell^\mu}$ and $\mathcal{T}_{\ell^\nu}$, which may be different. Consequently, $K_{\nu\mu}^m$ is non-square if $\ell^\mu \neq \ell^\nu$ for any $\mu, \nu \in J_P$.

TABLE 3.1
Naive upper bound for the number of matrices $K^m_{\nu\mu}$ that need computing for the test problems (TP.1–TP.4) outlined in Section 6, and the actual number required. The set J_P and the mesh level numbers $\boldsymbol{\ell}$ are selected automatically using Algorithm 1 in Section 5. See Sections 6.1 and 6.2 for more details.

Test Problem	card(J_P)	M	$(1+2M)$card(J_P)	actual
TP.1	169	93	31,603	616
TP.2	36	13	972	96
TP.3	17	3	119	35
TP.4	21	8	357	54

The key to a fast and efficient multilevel SGFEM algorithm is to first determine what, and what does not, need computing. If we use iterative solvers, then we only need to compute the action of A on vectors. Here, $\mathbf{v} = A\mathbf{x}$ can be computed blockwise via

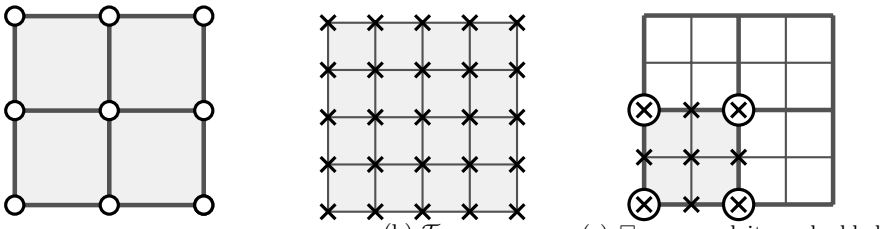
$$[\mathbf{v}]_\nu = [A\mathbf{x}]_\nu = \sum_{\mu\in J_P} A_{\nu\mu}[\mathbf{x}]_\mu = \sum_{\mu\in J_P}\sum_{m=0}^{M} [G_m]_{\nu\mu} K^m_{\nu\mu}[\mathbf{x}]_\mu, \qquad \nu \in J_P. \tag{3.8}$$

We need only compute $K^m_{\nu\mu}$ for all *distinct* triplets (m, ℓ^ν, ℓ^μ) where the corresponding entry $[G_m]_{\nu\mu}$ is *non-zero*. Due to the orthonormality of the polynomials $\{\psi_\mu(\mathbf{y})\}_{\mu\in J_P}$, the matrices $\{G_m\}_{m=0}^M$ are very sparse (in fact $G_0 = I$). Indeed, if the density ρ_m associated with π_m on Γ_m is an even function (symmetric about zero), then the matrices $\{G_m\}_{m=1}^M$ have at most two nonzero entries per row, see [27, 20]. Hence, a naive upper bound for the number of required stiffness matrices is $(1+2M)$card(J_P). This takes the sparsity of G_m into account, but does not exploit the fact that the *same* mesh may be assigned to several multi-indices $\mu \in J_P$. An adaptive algorithm for automatically selecting J_P and the associated set of mesh level numbers $\boldsymbol{\ell}$ is developed in Section 5. In Table 3.1 we record card(J_P) and the number of matrices $K^m_{\nu\mu}$ that are required at the final step of that algorithm (when the error tolerance is set to $\epsilon = 2\times 10^{-3}$), for the test problems outlined in Section 6 (see also Table 6.2). Since the same mesh level number is assigned to many multi-indices in J_P, the number of matrices computed is significantly lower than the bound.

Adaptive multilevel SGFEMs have been considered in [22, 18]. Those works use an explicit a posteriori error estimation strategy to drive the enrichment of the approximation space. In [18], all stiffness matrices $K^m_{\nu\mu}$ that are non-square ($\ell^\nu \neq \ell^\mu$) are approximated using a projection technique involving only the square matrices $K^m_{\mu\mu}$ that feature in the diagonal blocks $A_{\mu\mu}$ of A. Even with this approximation, the multilevel approach considered in [18] is reported to be computationally expensive. In the next section, we describe how the matrices $K^m_{\nu\mu}$ can be computed quickly and efficiently, without the need for the approximation used in [18].

3.2. Assembly of Stiffness Matrices. We describe the construction of $K^m_{\nu\mu}$ for two multi-indices $\mu, \nu \in J_P$, with $\ell^\mu \neq \ell^\nu$ (m is not important here) for a simple example. For clarity of presentation, we consider uniform meshes of square elements. However, the procedure is applicable to any conforming FEM spaces H^μ_1 and H^ν_1 for which $\mathcal{T}_{\ell^\nu}$ is nested in $\mathcal{T}_{\ell^\mu}$, or equivalently, when $\mathcal{T}_{\ell^\nu}$ is obtained from a conforming (without introducing hanging nodes) refinement of $\mathcal{T}_{\ell^\mu}$.

EXAMPLE 3.1. For simplicity, assume that $D \subset \mathbb{R}^2$ is a square and H^μ_1 and H^ν_1 are spaces of continuous piecewise bilinear functions associated with two uniform meshes of square elements ($\mathbb{Q}_1$ elements). In particular, let $\mathcal{T}_{\ell^\mu}$ denote a uniform 2×2 square partition of D with mesh level number ℓ^μ and let $\mathcal{T}_{\ell^\nu}$ be a uniform 4×4 square partition of D with $\ell^\nu := \ell^\mu + 1$ (representing, in this case, a uniform refinement of $\mathcal{T}_{\ell^\mu}$). For now, we retain the boundary nodes so that $N^\mu_1 := \dim(H^\mu_1) = 9$ and $N^\nu_1 := \dim(H^\nu_1) = 25$. See Figures 3.1(a) and 3.1(b). To construct $K^m_{\nu\mu} \in \mathbb{R}^{25\times 9}$, we compute a *coarse-element* matrix for each element $\square_{\text{coarse}}$ in $\mathcal{T}_{\ell^\mu}$. In Figure 3.1(c) we highlight



(a) $\mathcal{T}_{\ell^\mu}$. (b) $\mathcal{T}_{\ell^\nu}$. (c) $\square_{\text{coarse}}$ and its embedded elements.

FIG. 3.1. *Example meshes with (a) $N_1^\mu = 9$ and level number ℓ^μ and (b) $N_1^\nu = 25$ and level number $\ell^\nu = \ell^\mu + 1$.*

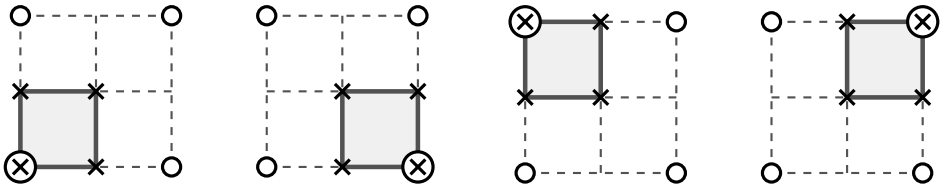


FIG. 3.2. *The four embedded elements in Figure 3.1(c) on which we construct four 4×4 local matrices.*

one such element, and the four (fine) elements $\square_{\text{fine}}$ in $\mathcal{T}_{\ell^\nu}$ that are embedded within it. The associated coarse-element matrix $K^m_{\nu\mu,c} \in \mathbb{R}^{9\times 4}$ has entries

$$[K^m_{\nu\mu,c}]_{ji} = \int_{\square_{\text{coarse}}} a_m(\mathbf{x}) \nabla \phi_i^{\mu,c}(\mathbf{x}) \cdot \nabla \phi_j^{\nu,c}(\mathbf{x}) \, d\mathbf{x}, \qquad i = 1,2,3,4, \qquad j = 1,2,\ldots,9,$$

where $\{\phi_i^{\mu,c}\}_{i=1}^4$ and $\{\phi_j^{\nu,c}\}_{j=1}^9$ are basis functions associated with the round and cross markers, with support on $\square_{\text{coarse}}$ and patches of $\square_{\text{coarse}}$, respectively. To construct $K^m_{\nu\mu,c}$, we concatenate four *fine-element* matrices $K^m_{\nu\mu,f} \in \mathbb{R}^{4\times 4}$ defined by

$$[K^m_{\nu\mu,f}]_{ji} = \int_{\square_{\text{fine}}} a_m(\mathbf{x}) \nabla \phi_i^{\mu,c}(\mathbf{x}) \cdot \nabla \phi_j^{\nu,f}(\mathbf{x}) \, d\mathbf{x}, \qquad i,j = 1,2,3,4,$$

where $\square_{\text{fine}}$ is one of the four elements embedded in $\square_{\text{coarse}}$. Here, $\{\phi_j^{\nu,f}\}_{j=1}^4$ are the basis functions defined with respect to the crosses in Figure 3.2, that are supported only on $\square_{\text{fine}}$ (shaded region).

For $\mathbb{Q}_1$ elements, constructing $K^m_{\nu\mu}$ boils down to the assembly of 4×4 fine-element matrices $K^m_{\nu\mu,f}$. Similarly, for $\mathbb{Q}_2$ elements (continuous piecewise biquadratic approximation), the procedure requires the assembly of 9×9 fine-element matrices $K^m_{\nu\mu,f}$. If $K^m_{\nu\mu}$ is square ($\ell^\mu = \ell^\nu$), we can use the traditional element construction. In either case, we only need to perform integration on elements in the fine mesh, for which we implement an exact quadrature rule.

REMARK 3.1. *When the meshes $\mathcal{T}_{\ell^\mu}$ and $\mathcal{T}_{\ell^\nu}$ are uniform, as in Example 3.1, the computation of the fine-element matrices can be vectorised over all the coarse elements.*

4. Energy Norm A Posteriori Error Estimation. Given an approximation space X of the form (3.3) and an SGFEM approximation $u_X \in X$ satisfying (3.5), we want to estimate the energy error $||u - u_X||_B$. We now extend the implicit strategy developed in [4, 6].

Computing the error $e = u - u_X \in V$ is a non-trivial task. Due to the bilinearity of $B(\cdot,\cdot)$ it is clear that e satisfies

$$B(e,v) = B(u,v) - B(u_X,v) = F(v) - B(u_X,v), \quad \text{for all } v \in V.$$

We look for an approximation to e in an SGFEM space $W \subset V$ that is richer than X, i.e., $W \supset X$. The quality of the resulting approximation is closely related to the quality of the Galerkin approximation $u_W \in W$ satisfying

$$\text{find } u_W \in W: \quad B(u_W, v) = F(v), \quad \text{for all } v \in W. \tag{4.1}$$

By letting $e_W = u_W - u_X$ we see that

$$B(e_W, v) = B(u_W, v) - B(u_X, v) = F(v) - B(u_X, v), \quad \text{for all } v \in W, \tag{4.2}$$

and thus $e_W \in W$ satisfying (4.2) estimates the true error $e \in V$. Clearly, since e_W estimates e, SGFEM spaces W that contain significantly improved approximations u_W to u (compared to u_X), also contain good estimates e_W to e. To analyse the quality of the error estimate $||e_W||_B$, for a given choice of W, we require the following assumption.

ASSUMPTION 4.1. Let the functions u, u_X and u_W satisfy (2.2), (3.5) and (4.1) respectively. There exists a constant $\beta \in [0, 1)$ (the saturation constant) such that

$$||u - u_W||_B \leq \beta ||u - u_X||_B. \tag{4.3}$$

We will also assume that $W := X \oplus Y$ for some space $Y \subset V$ (the 'detail' space) such that $X \cap Y = \{0\}$. Since computing $e_W \in W$ satisfying (4.2) is usually too expensive we instead exploit the decomposition of W and solve:

$$\text{find } e_Y \in Y: \quad B_0(e_Y, v) = F(v) - B(u_X, v), \quad \text{for all } v \in Y. \tag{4.4}$$

Notice the use of the parameter-free $B_0(\cdot, \cdot)$ bilinear form from (2.6) on the left-hand side of (4.4). To analyse the quality of the approximation $||e_Y||_{B_0} \approx ||e_W||_B$ we require the following result. Since X and Y are disjoint, and $B_0(\cdot, \cdot)$ induces a norm on the Hilbert space V in (2.2), there exists a constant $\gamma \in [0, 1)$ such that

$$|B_0(u, v)| \leq \gamma ||u||_{B_0} ||v||_{B_0}, \quad \text{for all } u \in X, \quad \text{for all } v \in Y, \tag{4.5}$$

see [1, Theorem 5.4]. Utilising (4.3) and (4.5) yields the following result [14, 6].

THEOREM 4.1. *Let $u \in V = H_0^1(D) \otimes L_\pi^2(\Gamma)$ satisfy the variational problem (2.2) associated with the parametric diffusion problem (1.1)–(1.2) and let $u_X \in X$ satisfy (3.5) for X in (3.3). Choose $Y \subset V$ such that $X \cap Y = \{0\}$ and let $e_Y \in Y$ satisfy (4.4). If Assumption 4.1 holds, as well as Assumptions 2.1 and 2.2, then $\eta := ||e_Y||_{B_0}$ satisfies*

$$\sqrt{\lambda}\, \eta \leq ||u - u_X||_B \leq \frac{\sqrt{\Lambda}}{\sqrt{1 - \gamma^2}\sqrt{1 - \beta^2}}\, \eta, \tag{4.6}$$

where λ and Λ are defined in (2.8), $\gamma \in [0, 1)$ satisfies (4.5), and $\beta \in [0, 1)$ satisfies (4.3).

The quality of the error estimate $\eta \approx ||e||_B$ depends on our choice of Y because the constants γ and β in (4.6) depend on Y. In the next section we describe a suitable structure for Y when X has the multilevel structure in (3.3).

4.1. Choice of Detail Space Y. In order to compute $\eta = ||e_Y||_{B_0}$ by solving (4.4), we need to choose the space Y. Note that in an adaptive SGFEM algorithm, Y must vary with X, which is enriched at each step as we reduce $||u - u_X||_B$. Suppose that X has the form (3.3), where J_P and the set of finite element spaces $\mathbf{H}_1$ are given. As stated in [6, Remark 4.3], one possibility is to choose a second set of multi-indices $J_Q \subset J$ that satisfy $J_Q \cap J_P = \emptyset$ and construct

$$Y := \Big(\bigoplus_{\mu \in J_P} H_2^\mu \otimes P^\mu \Big) \oplus \Big(\bigoplus_{\nu \in J_Q} H \otimes P^\nu \Big), \tag{4.7}$$

where $H_2^\mu \subset H_0^1(D)$ are FEM spaces satisfying $H_1^\mu \cap H_2^\mu = \{0\}$ for all $\mu \in J_P$ and $H \subset H_0^1(D)$ is some other finite element space (to be defined later). Clearly, we have

$$Y := Y_1 \oplus Y_2 := \Big(\bigoplus_{\mu\in J_P} Y_1^\mu \Big) \oplus \Big(\bigoplus_{\nu\in J_Q} Y_2^\nu \Big), \qquad Y_1^\mu := H_2^\mu \otimes P^\mu, \qquad Y_2^\nu := H \otimes P^\nu, \tag{4.8}$$

which in turn leads to the following decomposition of $e_Y \in Y$,

$$e_Y = e_{Y_1} + e_{Y_2} = \sum_{\mu\in J_P} e_{Y_1}^\mu + \sum_{\nu\in J_Q} e_{Y_2}^\nu, \qquad e_{Y_1}^\mu \in Y_1^\mu, \qquad e_{Y_2}^\nu \in Y_2^\nu.$$

Since $B_0(\cdot,\cdot)$ is parameter-free and $J_P \cap J_Q = \emptyset$, then, as a consequence of the orthogonality property (3.2), problem (4.4) decouples into $\mathrm{card}(J_P \cup J_Q) = \mathrm{card}(J_P) + \mathrm{card}(J_Q)$ smaller problems:

$$\text{find } e_{Y_1}^\mu \in Y_1^\mu: \qquad B_0(e_{Y_1}^\mu, v) = F(v) - B(u_X, v), \qquad \text{for all } v \in Y_1^\mu, \qquad \mu \in J_P, \tag{4.9}$$

$$\text{find } e_{Y_2}^\nu \in Y_2^\nu: \qquad B_0(e_{Y_2}^\nu, v) = F(v) - B(u_X, v), \qquad \text{for all } v \in Y_2^\nu, \qquad \nu \in J_Q. \tag{4.10}$$

In addition, the error estimate η in (4.6) admits the decomposition

$$\eta = ||e_Y||_{B_0} = \big(||e_{Y_1}||_{B_0}^2 + ||e_{Y_2}||_{B_0}^2 \big)^{\frac{1}{2}} = \Bigg(\sum_{\mu\in J_P} ||e_{Y_1}^\mu||_{B_0}^2 + \sum_{\nu\in J_Q} ||e_{Y_2}^\nu||_{B_0}^2 \Bigg)^{\frac{1}{2}}. \tag{4.11}$$

For each $\mu \in J_P$ in (4.9) we solve a problem of size $N_{Y_1}^\mu := \dim(H_2^\mu \otimes P^\mu) = \dim(H_2^\mu)$. For each $\nu \in J_Q$ in (4.10), we solve a problem of size $N_{Y_2}^\nu := \dim(H \otimes P^\nu) = \dim(H)$. We refer to $||e_{Y_1}||_{B_0}$ as the *spatial error estimate*, and to $||e_{Y_2}||_{B_0}$ as the *parametric error estimate*. For the adaptive algorithm in Section 5, it will be beneficial to define the set $\mathbf{H}_2 = \{H_2^\mu\}_{\mu\in J_P}$ as well as the sets

$$\mathbf{N}_{Y_1} = \{N_{Y_1}^\mu\}_{\mu\in J_P}, \qquad \mathbf{N}_{Y_2} = \{N_{Y_2}^\nu\}_{\nu\in J_Q}.$$

The quality of the error estimate η depends on our choice of J_Q and $\mathbf{H}_2$ as well as the finite element space H appearing in the definition of Y_2, since they affect the constants γ and β appearing in (4.6). The error bound is sharp when β and γ are close to zero.

If Assumption 2.2 holds, then $H_0^1(D)$ is a Hilbert space with respect to the inner product

$$\langle a_0 u, v\rangle = \int_D a_0(\mathbf{x}) \nabla u(\mathbf{x}) \cdot \nabla v(\mathbf{x})\, d\mathbf{x}, \qquad u, v \in H_0^1(D).$$

Furthermore, since $H_1^\mu \cap H_2^\mu = \{0\}$ for all $\mu \in J_P$, there exists a constant $\gamma^\mu \in [0,1)$ such that

$$|\langle a_0 u, v\rangle| \le \gamma^\mu \langle a_0 u, u\rangle^{1/2} \langle a_0 v, v\rangle^{1/2}, \quad \text{for all } u \in H_1^\mu, \quad \text{for all } v \in H_2^\mu, \tag{4.12}$$

for all $\mu \in J_P$ (again, see [1, Theorem 5.4]). We denote the smallest such constant (known as the CBS constant) by $\gamma_{\min}^\mu$. Note that this constant only depends on the chosen finite element spaces H_1^μ and H_2^μ and is known explicitly in many cases, see [14]. It is then straightforward to prove, using the mutual orthogonality of the sets $\{\psi_\mu(\mathbf{y})\}_{\mu\in J_P}$ and $\{\psi_\nu(\mathbf{y})\}_{\nu\in J_Q}$ and the definition of $B_0(\cdot,\cdot)$ that with Y chosen as in (4.7), the bound (4.5) holds with

$$\gamma := \max_{\mu\in J_P} \{\gamma_{\min}^\mu\}. \tag{4.13}$$

See also [6, Remark 4.3].

REMARK 4.1. *Since H in* (4.7) *does not depend on $\nu \in J_Q$, the matrix that characterises the linear systems associated with* (4.10) *is the same for all $\nu \in J_Q$. Only the right-hand side changes. Consequently, we can vectorise the system solves associated with* (4.10) *over the multi-indices J_Q.*

REMARK 4.2. *For two FEM spaces H_1^μ and H_2^μ, there often exists a sharp upper bound for the associated CBS constant $\gamma_{\min}^\mu$ that is independent of the mesh level number ℓ^μ, see [14].*

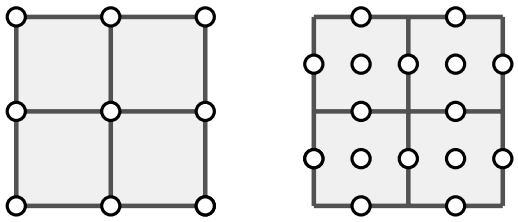


FIG. 4.1. *The nodes associated with H_1^μ (left) and H_2^μ (right), when H_1^μ is chosen to be a $\mathbb{Q}_1$ space and H_2^μ is chosen to be a 'broken' $\mathbb{Q}_2$ space associated with the same mesh $\mathcal{T}_{\ell^\mu}$ as H_1^μ.*

4.2. The Spatial Error Estimator. We now briefly discuss possible choices of the FEM spaces $\mathbf{H}_2 = \{H_2^\mu\}_{\mu\in J_P}$ that define the tensor spaces $\mathbf{Y}_1 := \{Y_1^\mu\}_{\mu\in J_P}$ in (4.8). Recall that each FEM space H_1^μ is associated with a mesh $\mathcal{T}_{\ell^\mu} = \mathcal{T}_i$ for some $i \in \mathbb{N}_0$. One option is to construct a basis for H_2^μ with respect to the same mesh $\mathcal{T}_{\ell^\mu}$ but using polynomials of a higher degree. In order to ensure that $H_1^\mu \cap H_2^\mu = \{0\}$, we exclude basis functions associated with nodes associated with H_1^μ. For example, if the spaces $\mathbf{H}_1$ are $\mathbb{Q}_1$ FEM spaces, we may choose the spaces $\mathbf{H}_2$ to be 'broken' $\mathbb{Q}_2$ FEM spaces (see Figure 4.1). Another option is to use polynomials of the same degree, but introduce basis functions associated with the new nodes that would be introduced by performing the mesh refinement $\mathcal{T}_{\ell^\mu} \to \mathcal{T}_{i+1}$ (i.e., by increasing the level number by one).

4.3. The Parametric Error Estimator. It remains to explain how to choose the multi-indices J_Q and the space $H \subset H_0^1(D)$ that define the tensor spaces $\mathbf{Y}_2 := \{Y_2^\mu\}_{\mu\in J_Q}$ in (4.8). It was proven in [6] that $||e_{Y_2}^\nu||_{B_0} = 0$ for considerably many multi-indices $\nu \in J\backslash J_P$. In order to avoid unnecessary computations, it is essential that we first identify the set of multi-indices $J^* \subset J$ that result in non-zero contributions. Indeed, this set is given by

$$J^* = \big\{\mu \in J\backslash J_P;\ \mu = \nu \pm \epsilon^m\ \forall\ \nu \in J_P,\ \forall\ m \in \mathbb{N}\big\},$$

where $\epsilon^m := (\epsilon_1^m, \epsilon_2^m, \dots)$ is the Kronecker delta sequence such that $\epsilon_j^m = \delta_{mj}$ for all $j \in \mathbb{N}$. Since J^* is an infinite set, we need to choose a finite subset $J_Q \subset J^*$. We call J^* the set of 'neighbouring indices' to J_P and choose

$$J_Q = \big\{\nu \in J^*;\ \max\{\mathrm{supp}(\nu)\} \le M + \Delta_M\big\}, \tag{4.14}$$

where $\Delta_M \in \mathbb{N}$ is the number of additional parameters we wish to activate.

We now turn our attention to $H \subset H_0^1(D)$. Recall that $W = X \oplus Y$ in (4.1). The space Y (and hence $Y_2^\nu = H \otimes P^\nu$) should be chosen so that W contains functions that would result in an improved approximation $u_W \in W$ to u. We clearly want to choose Y so that we have an accurate energy error estimate η for the current approximation u_X. However, since we want to perform adaptivity, the functions in Y serve as candidates to be added to X at the next approximation step. Since X may be augmented with $H \otimes P^\nu$ for some $\nu \in J_Q$, we should choose H such that the structure of Y in (4.7) is maintained and the error estimator is straightforward to compute at each step. For this reason, we choose $H = H_1^{\bar\mu}$ for some $\bar\mu \in J_P$. That is, we choose H to be one of the FEM spaces already used in the construction of X.

When choosing $\bar\mu \in J_P$ we must consider the fact that through our choice of Y in (4.7), β in (4.6) depends on $\bar\mu$. We have to balance the accuracy of the estimate η against the cost to compute it. If we choose $\bar\mu$ such that $\ell^{\bar\mu} = \max_{\mu\in J_P} \boldsymbol{\ell}$ (i.e., choose the richest FEM space used so far), then $\dim(X)$ will grow too quickly when we augment X with functions in $\mathbf{Y}_2$. Similarly, if $\ell^{\bar\mu} = \min_{\mu\in J_P} \boldsymbol{\ell}$, the error reduction may be negligible if X is augmented with functions from $\mathbf{Y}_2$. To strike a balance, we will choose $\bar\mu$ to correspond to the FEM space H_1^μ with the smallest mesh

level number ℓ^μ such that the number of spaces with level number ℓ^μ or less is greater than or equal to $\lceil \frac{1}{2}\text{card}(J_P) \rceil$. We denote this choice by $\bar{\mu} = \arg\text{avg}_{\mu \in J_P} \boldsymbol{\ell}$.

EXAMPLE 4.1. Suppose $\text{card}(J_P) = 5$ and $\boldsymbol{\ell} = \{2, 3, 3, 2, 1\}$, then $\ell^{\bar{\mu}} = 2$. Similarly, if $\text{card}(J_P) = 3$ and $\boldsymbol{\ell} = \{4, 3, 2\}$, then $\ell^{\bar{\mu}} = 3$.

The choice $\bar{\mu} = \arg\text{avg}_{\mu \in J_P} \boldsymbol{\ell}$ ensures that the dimensions of the spaces in $\mathbf{Y}_2$ are always modest in comparison to those of the spaces in $\mathbf{X} = \{X^\mu\}_{\mu \in J_P}$ in (3.3).

5. Adaptive Multilevel SGFEM. Suppose that X and Y in (3.3) and (4.7) have been chosen (and so the sets of multi-indices $J_P, J_Q \subset J$ have also been chosen) and that the corresponding approximations $u_X \in X$ and $e_Y \in Y$ satisfying (3.5) and (4.4) have been computed. If $\eta = ||e_Y||_{B_0}$ is too large, we want to augment X with some of the functions in Y and compute a (hopefully) improved approximation to $u \in V$ satisfying (2.2). Of course, we could augment X with the full space Y to ensure it is sufficiently rich. However, we must also ensure that the total number of additional degrees of freedom (DOFs) introduced is balanced against the reduction in the energy error that is achieved. We should only augment X with functions that result in significant error reductions. Below, we demonstrate that using the sets of component estimates

$$\mathbf{E}_{Y_1} := \{||e^\mu_{Y_1}||_{B_0}\}_{\mu \in J_P}, \qquad \mathbf{E}_{Y_2} := \{||e^\mu_{Y_2}||_{B_0}\}_{\mu \in J_Q}, \tag{5.1}$$

(which are computed to determine η), we can estimate the error reduction that would be achieved by performing certain enrichment strategies at the next approximation step.

5.1. Estimated Error Reductions. Consider the discrete problems:

$$\text{find } u_{W_1} \in W_1 : \qquad B(u_{W_1}, v) = F(v), \qquad \text{for all } v \in W_1, \tag{5.2}$$

$$\text{find } u_{W_2} \in W_2 : \qquad B(u_{W_2}, v) = F(v), \qquad \text{for all } v \in W_2, \tag{5.3}$$

where W_1 and W_2 are 'enhanced' SGFEM approximation spaces given by

$$\begin{aligned} W_1 &:= X \oplus Y_{W_1} := X \oplus \Big(\bigoplus_{\mu \in \bar{J}_P} Y_1^\mu \Big), \quad \bar{J}_P \subset J_P, \\ W_2 &:= X \oplus Y_{W_2} := X \oplus \Big(\bigoplus_{\nu \in \bar{J}_Q} Y_2^\nu \Big), \quad \bar{J}_Q \subset J_Q. \end{aligned} \tag{5.4}$$

That is, u_{W_1} and u_{W_2} are SGFEM approximations to $u \in V$ computed in W_1 and W_2, respectively. Note that if $\bar{J}_P = J_P$ then $Y_{W_1} = Y_1$ and if $\bar{J}_Q = J_Q$ then $Y_{W_2} = Y_2$. However, we want to consider enrichment strategies associated with only important subsets of the multi-indices. The space W_1 corresponds to refining the finite element meshes associated with a subset of the multi-indices $\mu \in J_P$ used in the definition of X, whereas W_2 corresponds to adding new basis polynomials on the parameter domain. We want to estimate the potential pay-offs of these two strategies.

Let $e_{W_1} = u - u_{W_1}$ denote the error corresponding to the enhanced approximation u_{W_1}. Due to the orthogonality of e_{W_1} with functions in W_1 ($(u_{W_1} - u_X) \in W_1$ in particular) with respect to $B(\cdot, \cdot)$ (Galerkin-orthogonality), and the symmetry of $B(\cdot, \cdot)$, we find that

$$||e_{W_1}||^2_B = ||u - u_X||^2_B - ||u_{W_1} - u_X||^2_B.$$

Hence, $||u_{W_1} - u_X||^2_B$ characterises the reduction in $||u - u_X||^2_B$ (the square of the energy error) that would be achieved by augmenting X with Y_{W_1}, for a suitably chosen set $\bar{J}_P \subset J_P$, and computing an enhanced approximation $u_{W_1} \in W_1$ satisfying (5.2). Similarly, $||u_{W_2} - u_X||^2_B$ characterises the reduction in $||u - u_X||^2_B$ that would be achieved by augmenting X with Y_{W_2} for a suitably chosen

set $\bar{J}_Q \subset J_Q$ and computing $u_{W_2} \in W_2$ satisfying (5.3). The following result provides estimates for these quantities. This is a simple extension of a result proved in [4, 6]; the proof is very similar.

THEOREM 5.1. Let $u_X \in X$ be the SGFEM approximation satisfying (3.5) and let $u_{W_1} \in W_1$ and $u_{W_2} \in W_2$ satisfy problems (5.2) and (5.3). Define the quantities

$$\zeta_{W_1} := \sum_{\mu \in \bar{J}_P} ||e^{\mu}_{Y_1}||^2_{B_0}, \qquad \zeta_{W_2} := \sum_{\nu \in \bar{J}_Q} ||e^{\nu}_{Y_2}||^2_{B_0},$$

for some $\bar{J}_P \subset J_P$ and $\bar{J}_Q \subset J_Q$. Then the following estimates hold:

$$\lambda \zeta_{W_1} \leq ||u_{W_1} - u_X||^2_B \leq \frac{\Lambda}{1-\gamma^2} \zeta_{W_1}, \tag{5.5}$$

$$\lambda \zeta_{W_2} \leq ||u_{W_2} - u_X||^2_B \leq \Lambda \zeta_{W_2}, \tag{5.6}$$

where λ and Λ are the constants in (2.8), and $\gamma \in [0, 1)$ is the constant satisfying (4.13).

Given two sets of multi-indices $\bar{J}_P$ and $\bar{J}_Q$, we now determine an appropriate enrichment strategy for X by considering the bounds (5.5)–(5.6). One option would be to perform the enrichment strategy that corresponds to $\max\{\zeta_{W_1}, \zeta_{W_2}\}$. Whilst this may lead to a large reduction of $||u - u_X||^2_B$ (and hence of $||u - u_X||_B$), it doesn't take into account the computational cost incurred. We want to construct sequences of SGFEM spaces X for which the energy error converges to zero at the best possible rate with respect to $N_{\text{dof}} = \dim(X)$ for the chosen set of finite element spaces. Hence, the number of DOFs should be taken into account. Recall the definitions

$$N^{\mu}_{Y_1} := \dim(Y^{\mu}_1),\ \mu \in J_P, \qquad N^{\nu}_{Y_2} := \dim(Y^{\nu}_2),\ \nu \in J_Q. \tag{5.7}$$

The number of additional DOFs (compared to the current space X) associated with the spaces W_1 and W_2 in (5.4) is given by

$$N_{W_1} := \sum_{\mu \in \bar{J}_P} N^{\mu}_{Y_1}, \qquad N_{W_2} := \sum_{\nu \in \bar{J}_Q} N^{\nu}_{Y_2},$$

respectively. Due to Theorem 5.1, the ratios

$$R_{W_1} := \frac{\zeta_{W_1}}{N_{W_1}}, \qquad R_{W_2} := \frac{\zeta_{W_2}}{N_{W_2}}, \tag{5.8}$$

provide approximations to $||u_{W_1} - u_X||^2_B / N_{W_1}$ and $||u_{W_2} - u_X||^2_B / N_{W_2}$, respectively. Once we have chosen $\bar{J}_P$ and $\bar{J}_Q$, we augment X with the space Y_{W_1} or Y_{W_2}, that corresponds to $\max\{R_{W_1}, R_{W_2}\}$. In the next section we propose an adaptive multilevel SGFEM algorithm for the numerical solution of (1.1)–(1.2) as well as two methods for the selection of the sets of multi-indices $\bar{J}_P$ and $\bar{J}_Q$.

5.2. An Adaptive Algorithm. Using the a posteriori error estimation strategy discussed in Section 4.1, and the estimated error reductions described in Section 5.1, we now propose an adaptive algorithm that generates a sequence of multilevel SGFEM spaces

$$X^0 \subset X^1 \cdots \subset X^k \cdots \subset X^K \subset V,$$

and terminates at step $k = K$ when the SGFEM approximation $u^K_X \in X^K$ to u satisfies a prescribed error tolerance ϵ. We start by selecting an initial low-dimensional SGFEM space X^0 of the form (3.3) and compute an initial approximation $u^0_X \in X^0$ to $u \in V$ satisfying (3.5). Assuming that the polynomial degree of the FEM approximation on D has been fixed, we only need to supply an initial set of multi-indices J^0_P, as well as a set of mesh level numbers $\boldsymbol{\ell}^0 = \{\ell^{\mu}_0\}_{\mu \in J^0_P}$. We then

Algorithm 1: Adaptive multilevel SGFEM

Input : Problem data $a(\mathbf{x},\mathbf{y})$, $f(\mathbf{x})$; initial index set J_P^0 and mesh level numbers $\boldsymbol{\ell}^0$; energy error tolerance ϵ.
Output: Final SGFEM approximation u_X^K and energy error estimate η^K.

Choose `version` (1 or 2)
for $k = 0, 1, 2, \dots$ **do**
  $u_X^k \leftarrow \texttt{SOLVE}\big[a, f, J_P^k, \boldsymbol{\ell}^k\big]$
  $J_Q^k \leftarrow \texttt{PARAMETRIC\_INDICES}\big[J_P^k\big]$ see: (4.14)
  $\mathbf{E}_{Y_1}^k \leftarrow \texttt{COMPONENT\_SPATIAL\_ERRORS}\big[u_X^k, J_P^k, \boldsymbol{\ell}^k\big]$ (5.1)
  $\mathbf{E}_{Y_2}^k \leftarrow \texttt{COMPONENT\_PARAMETRIC\_ERRORS}\big[u_X^k, J_Q^k, \boldsymbol{\ell}^k\big]$
  $\eta^k = \big[\sum_{\mu\in J_P^k} ||e_{Y_1}^{\mu,k}||_{B_0}^2 + \sum_{\nu\in J_Q^k} ||e_{Y_2}^{\nu,k}||_{B_0}^2\big]^{\frac{1}{2}}$ (4.11)
  if $\eta^k < \epsilon$ **then**
    return u_X^k, η^k
  else
    $[\texttt{refinement\_type}, \bar{J}^k] \leftarrow \texttt{ENRICHMENT\_INDICES}\big[\texttt{version}, \mathbf{E}_{Y_1}^k, \mathbf{E}_{Y_2}^k, J_P^k, J_Q^k\big]$
    if `refinement_type` $=$ `spatial` **then**
      $J_P^{k+1} = J_P^k$
      $\boldsymbol{\ell}^{k+1} = \big\{\ell_k^{\mu+};\ \mu\in\bar{J}^k\big\} \cup \big\{\ell_k^{\mu};\ \mu\in J_P^k\backslash\bar{J}^k\big\}$ (5.9)
    else
      $J_P^{k+1} = J_P^k \cup \bar{J}^k$
      $\boldsymbol{\ell}^{k+1} = \boldsymbol{\ell}^k \cup \big\{\ell_k^{\bar{\mu}};\ \nu\in\bar{J}^k\big\}$
    end
  end
end

consider two enrichment strategies. The first option is to refine certain meshes associated with the spaces $\mathbf{H}_1^0$ and produce a new set $\boldsymbol{\ell}^1$. If $\ell_0^\mu = i$ for some $\mu \in J_P$, and we want to perform a refinement, we set $\ell_1^\mu = i+1$ or equivalently replace $\mathcal{T}_{\ell_0^\mu}$ with the next mesh in the sequence $\boldsymbol{\mathcal{T}}$ in (3.4). In our adaptive algorithm we write

$$\ell_0^\mu \to \ell_0^{\mu+} =: \ell_1^\mu. \tag{5.9}$$

The second option is to add multi-indices to J_P^0 to give a new set J_P^1. In this case, we must also update $\boldsymbol{\ell}^0$ with new mesh parameters to maintain the relationship $\mathrm{card}(J_P) = \mathrm{card}(\boldsymbol{\ell})$. Specifically, we add a copy of $\ell_0^{\bar{\mu}}$ to $\boldsymbol{\ell}^0$, for every multi-index added to J_P^0 (see Section 4.3 for the definition of $\bar{\mu}$). Once J_P^1 and $\boldsymbol{\ell}^1$ are defined, and $u_X^1 \in X^1$ is computed, the process is repeated.

The general process is outlined in Algorithm 1. At a given step k:

- `SOLVE` computes an SGFEM approximation $u_X \in X$ to $u \in V$ satisfying (3.5).
- `PARAMETRIC_INDICES` uses (4.14) to determine a subset J_Q of the neighbouring indices to J_P for a prescribed choice of Δ_M.
- `COMPONENT_SPATIAL_ERRORS` and `COMPONENT_PARAMETRIC_ERRORS` compute the sets of error estimates $\mathbf{E}_{Y_1}$ and $\mathbf{E}_{Y_2}$ in (5.1), respectively, by solving (4.9) and (4.10).
- `ENRICHMENT_INDICES` analyses the sets $\mathbf{E}_{Y_1}$ and $\mathbf{E}_{Y_2}$ in conjunction with the formulae in (5.8) to determine how to enrich the current SGFEM space X.

A key part of `ENRICHMENT_INDICES` is the determination of suitable sets $\bar{J}_P \subset J_P$ and $\bar{J}_Q \subset J_Q$, which we describe in the next section. Algorithm 1 subsequently performs either a spatial or

Algorithm 2: `ENRICHMENT_INDICES` versions 1 and 2

Input : `version`; $\mathbf{E}_{Y_1}^k$; $\mathbf{E}_{Y_2}^k$; J_P^k; J_Q^k.
Output: `refinement_type`, $\bar{J}^k$.

$\delta_{Y_1}^k = \max_{\mu \in J_P^k} \mathbf{R}_{Y_1}^k$, $\delta_{Y_2}^k = \max_{\nu \in J_Q^k} \mathbf{R}_{Y_2}^k$
if $\delta_{Y_1}^k > \delta_{Y_2}^k$ **then**
  $\bar{J}_Q^k = \{\nu \in J_Q^k;\ R_{Y_2}^{\nu,k} = \delta_{Y_2}^k\}$
  if `version` $= 1$ **then**
    $\bar{J}_P^k = \{\mu \in J_P^k;\ R_{Y_1}^{\mu,k} > \delta_{Y_2}^k\}$
  else
    $\bar{J}_P^k \leftarrow$ `MARK`$[\mathbf{E}_{Y_1}^k, \mathbf{N}_{Y_1}^k, \delta_{Y_2}^k]$
  end
else
  $\bar{J}_P^k = \{\mu \in J_P^k;\ R_{Y_1}^{\mu,k} = \delta_{Y_1}^k\}$
  if `version` $= 1$ **then**
    $\bar{J}_Q^k = \{\nu \in J_Q^k;\ R_{Y_2}^{\nu,k} > \delta_{Y_1}^k\}$
  else
    $\bar{J}_Q^k \leftarrow$ `MARK`$[\mathbf{E}_{Y_2}^k, \mathbf{N}_{Y_2}^k, \delta_{Y_1}^k]$
  end
end
if $R_{W_1}^k > R_{W_2}^k$ **then**
  `refinement_type` $=$ `spatial`, $\bar{J}^k = \bar{J}_P^k$
else
  `refinement_type` $=$ `parametric`, $\bar{J}^k = \bar{J}_Q^k$
end
return [`refinement_type`, $\bar{J}^k$]

parametric refinement associated with the set of multi-indices $\bar{J} := \bar{J}_P$ or $\bar{J} := \bar{J}_Q$, respectively.

5.3. Selection of the Enrichment Multi-indices. We introduce two versions of the module `ENRICHMENT_INDICES`, which are outlined in Algorithm 2. To begin, define the sets

$$\mathbf{R}_{Y_1} := \left\{R_{Y_1}^{\mu}\right\}_{\mu \in J_P} := \left\{\frac{||e_{Y_1}^{\mu}||_{B_0}^2}{N_{Y_1}^{\mu}}\right\}_{\mu \in J_P}, \quad \mathbf{R}_{Y_2} := \left\{R_{Y_2}^{\nu}\right\}_{\nu \in J_Q} := \left\{\frac{||e_{Y_2}^{\nu}||_{B_0}^2}{N_{Y_2}^{\nu}}\right\}_{\nu \in J_Q},$$

of estimated error reduction ratios and consider the quantities

$$\delta_{Y_1} := \max_{\mu \in J_P} \mathbf{R}_{Y_1}, \qquad \delta_{Y_2} := \max_{\nu \in J_Q} \mathbf{R}_{Y_2}.$$

Version 1 of Algorithm 2 is simple. If $\delta_{Y_1} > \delta_{Y_2}$, we define $\bar{J}_P$ to be the set of multi-indices $\mu \in J_P$ such that $R_{Y_1}^{\mu} > \delta_{Y_2}$ and we define $\bar{J}_Q$ to be the set of multi-indices $\nu \in J_Q$ such that $R_{Y_2}^{\nu} = \delta_{Y_2}$. Similarly, if $\delta_{Y_2} > \delta_{Y_1}$, we define $\bar{J}_Q$ to be the set of multi-indices in J_Q such that $R_{Y_2}^{\nu} > \delta_{Y_1}$ and $\bar{J}_P$ is the set of multi-indices in J_P such that $R_{Y_1}^{\mu} = \delta_{Y_1}$. The refinement type is then determined by computing R_{W_1} and R_{W_2} in (5.8). If $R_{W_1} > R_{W_2}$ we perform *spatial* refinement and set $\bar{J} = \bar{J}_P$. Otherwise, we enrich the *parametric* part, and set $\bar{J} = \bar{J}_Q$.

Version 2 is similar. However, if $\delta_{Y_1} > \delta_{Y_2}$, we choose $\bar{J}_P$ to be the largest subset of J_P such that $R_{W_1} > \delta_{Y_2}$ (recall R_{W_1} depends on $\bar{J}_P$). Similarly, if $\delta_{Y_2} > \delta_{Y_1}$, we choose $\bar{J}_Q$ to be

the largest subset of J_Q such that $R_{W_2} > \delta_{Y_1}$. As before, the refinement type chosen is the one associated with $\max\{R_{W_1}, R_{W_2}\}$. Version 2 is reminiscent of a Dörfler marking strategy [16] and so the module that generates $\bar{J}_P$ (if $\delta_{Y_1} > \delta_{Y_2}$) and $\bar{J}_Q$ (if $\delta_{Y_2} > \delta_{Y_1}$) is called `MARK`.

REMARK 5.1. *A key feature of both versions of `ENRICHMENT_INDICES` is that no marking or tuning parameters are required. The user only needs to choose Δ_M in the definition of J_Q in* (4.14). *This fixes an upper bound on the number of new parameters y_m that may be activated.*

6. Numerical Experiments. We now investigate the performance of Algorithms 1 and 2 in computing approximate solutions to (1.1)–(1.2). First, we describe four test problems. These differ, in particular, in the choice of $a(\mathbf{x}, \mathbf{y})$, and give rise to sequences of coefficients $\{\|a_m\|_\infty\}_{m=1}^\infty$ that decay at different rates. Recall, $y_m \in \Gamma_m = [-1, 1]$ is the image of a uniform random variable and $\pi_m(y_m)$ is the associated probability measure, for $m \in \mathbb{N}$.

Test Problem 1 (TP.1). First, we consider a problem from [4, 14]. Let $f(\mathbf{x}) = \frac{1}{8}(2-x_1^2-x_2^2)$ for $\mathbf{x} = (x_1, x_2)^\top \in D := [-1, 1]^2$ and assume that

$$a(\mathbf{x}, \mathbf{y}) = 1 + \sigma\sqrt{3} \sum_{m=1}^{\infty} \sqrt{\lambda_m}\phi_m(\mathbf{x})y_m, \tag{6.1}$$

where (λ_m, ϕ_m) are the eigenpairs of the operator associated with the covariance function

$$C[a](\mathbf{x}, \mathbf{x}') = \exp\left(- \frac{|x_1 - x_1'|}{l_1} - \frac{|x_2 - x_2'|}{l_2}\right), \qquad \mathbf{x}, \mathbf{x}' \in D.$$

As in [14] we choose $\sigma = 0.15$ (the standard deviation) and $l_1 = l_2 = 2$ (the correlation lengths). It can be shown that asymptotically (as $m \to \infty$), λ_m is $\mathcal{O}(m^{-2})$, see [26].

Test Problem 2 (TP.2). Next, we consider a problem from [18, 6]. Let $f(\mathbf{x}) = 1$ for $\mathbf{x} = (x_1, x_2)^\top \in D := [0, 1]^2$ and assume that

$$a(\mathbf{x}, \mathbf{y}) = 1 + \sum_{m=1}^{\infty} \alpha_m \cos(2\pi\beta_m^1 x_1) \cos(2\pi\beta_m^2 x_2)y_m,$$

where $\beta_m^1 = m - k_m(k_m + 1)/2$, $\beta_m^2 = k_m - \beta_m^1$ and $k_m = \lfloor -1/2 + (1/4 + 2m)^{1/2} \rfloor$ for $m \in \mathbb{N}$. In this test problem, we select the amplitude coefficients $\alpha_m = 0.547m^{-2}$.

Test Problem 3 (TP.3). This is the same as TP.2 but we now choose $\alpha_m = 0.832m^{-4}$, so that the terms in the expansion of $a(\mathbf{x}, \mathbf{y})$ decay more quickly.

Test Problem 4 (TP.4). Finally, we consider a problem from [26]. Let $f(\mathbf{x})$ and D be as in TP.2 and assume that

$$a(\mathbf{x}, \mathbf{y}) = 2 + \sqrt{3} \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sqrt{\nu_{ij}}\phi_{ij}(\mathbf{x})y_{ij}, \qquad y_{ij} \in [-1, 1] \tag{6.2}$$

where $\phi_{00} = 1$, $\nu_{00} = \frac{1}{4}$ and

$$\phi_{ij} = 2\cos(i\pi x_1)\cos(j\pi x_2), \quad \nu_{ij} = \frac{1}{4}\exp(-\pi(i^2 + j^2)l^{-2}).$$

We choose the correlation length $l = 0.65$ and rewrite the sum (6.2) in terms of a single index m to mimic the form (1.3), with the sequence $\{\nu_m\}_{m=1}^\infty$ ordered descendingly.

TABLE 6.1
Reference energies $||u^{ref}||_B$ for the four test problems TP.1–TP.4 presented in Section 6.

Test Problem	Reference Energy $||u^{\text{ref}}||_B$
TP.1	1.50342524×10^{-1}
TP.2	1.90117000×10^{-1}
TP.3	1.94142000×10^{-1}
TP.4	1.34570405×10^{-1}

6.1. Experimental Setup. To begin, we select an appropriate set of finite element spaces $\mathbf{H}_1$. Since D is square in all cases we choose a sequence $\boldsymbol{\mathcal{T}}$ of uniform meshes of square elements, with $\mathcal{T}_i$ representing a $2^i \times 2^i$ grid over D (thus $\mathcal{T}_{i+1}$ represents a uniform refinement of $\mathcal{T}_i$) with element width $h(i) = 2^{1-i}$ for TP.1 and $h(i) = 2^{-i}$ for TP.2–TP.4. We then choose $\mathbf{H}_1$ to be the set of $\mathbb{Q}_1$ finite element spaces associated with $\boldsymbol{\mathcal{T}}$. We initialise Algorithm 1 with

$$J_P^0 = \{(0,0,\dots),(1,0,\dots)\}, \qquad \boldsymbol{\ell}^0 = \{4,4\} \qquad (16 \times 16 \text{ grids}).$$

To compute the error estimator η defined in Section 4.1, the FEM spaces $\mathbf{H}_2 = \{H_2^\mu\}_{\mu\in J_P}$ are chosen to be broken $\mathbb{Q}_2$ spaces (see Figure 4.1) defined with respect to the same meshes as the spaces $\mathbf{H}_1$, as described in Section 4.2. Note that for this setup, if a_0 in (1.3) is a constant, we have $\gamma \leq \sqrt{5/11}$ in (4.5); c.f. Remark 4.2 and see [14] for a proof. We also fix $\Delta_M = 5$ in the definition of J_Q in (4.14). Due to Galerkin orthogonality, the exact energy error $||u - u_X^k||_B$ at step k admits the representation

$$||u - u_X^k||_B = \left(||u||_B^2 - ||u_X^k||_B^2\right)^{\frac{1}{2}}. \tag{6.3}$$

To examine the *effectivity index* $\theta^k = \eta^k/||u - u^k||_B$ we approximate u in (6.3) with an accurate 'reference' solution $u^{\text{ref}} \in X^{\text{ref}}$. The space X^{ref} is generated by applying Algorithm 1 with a much smaller error tolerance ϵ than the one used to generate η^k, $k = 1,\dots,K$. The reference energies $||u^{\text{ref}}||_B$ required for the approximation of (6.3) are provided in Table 6.1.

6.2. Experiment 1 (convergence rates). In our first experiment we solve test problems TP.1–TP.4 using Algorithms 1 and 2 (version 1) with tolerance $\epsilon = 2\times10^{-3}$. In Figure 6.1 we plot the evolution of the estimated error η^k against $\dim(X^k)$ (left plots) over each step of the iteration, as well as estimates of the effectivity indices θ^k (right plots). For test problems TP.2–TP.4, we observe that the estimated error behaves like $N_{\text{dof}}^{-0.5}$. Note that this is an improvement on the convergence rates obtained in [6, 5] for the same test problems, where single-level SGFEM spaces of the form (1.5) are employed. Due to our choice of FEM spaces $\mathbf{H}_1$ (bilinear approximation), and the spatial regularity of the solution, this is the optimal rate of convergence. That is, we achieve the rate afforded to the analogous parameter-free problem when employing $\mathbb{Q}_1$ approximation over uniform square meshes, and performing uniform mesh refinements. As proven in [12, 13, 23], the optimal achievable rate is a consequence of the fact that the sequence $\{||a_m||_\infty\}_{m=1}^\infty$ decays sufficiently quickly, and the error attributed to the choice of spatial discretisation dominates. Conversely, for test problem TP.1 the associated sequence $\{||a_m||_\infty\}_{m=1}^\infty$ decays too slowly, and the error attributed to the parametric part of the approximation dominates. For this reason, test problem TP.1 is particularly challenging. Nevertheless, for moderate error tolerances, our adaptive algorithm can tackle it efficiently. For all test problems considered, the effectivity indices are close to one, meaning that the error estimate is highly accurate.

Figure 6.1 provides no information about the structure of the multilevel SGFEM spaces X^K constructed. To illustrate the qualitative differences between the four cases, in Table 6.2 we record the number of activated parameters M, the cardinality of the final set J_P^K and the number

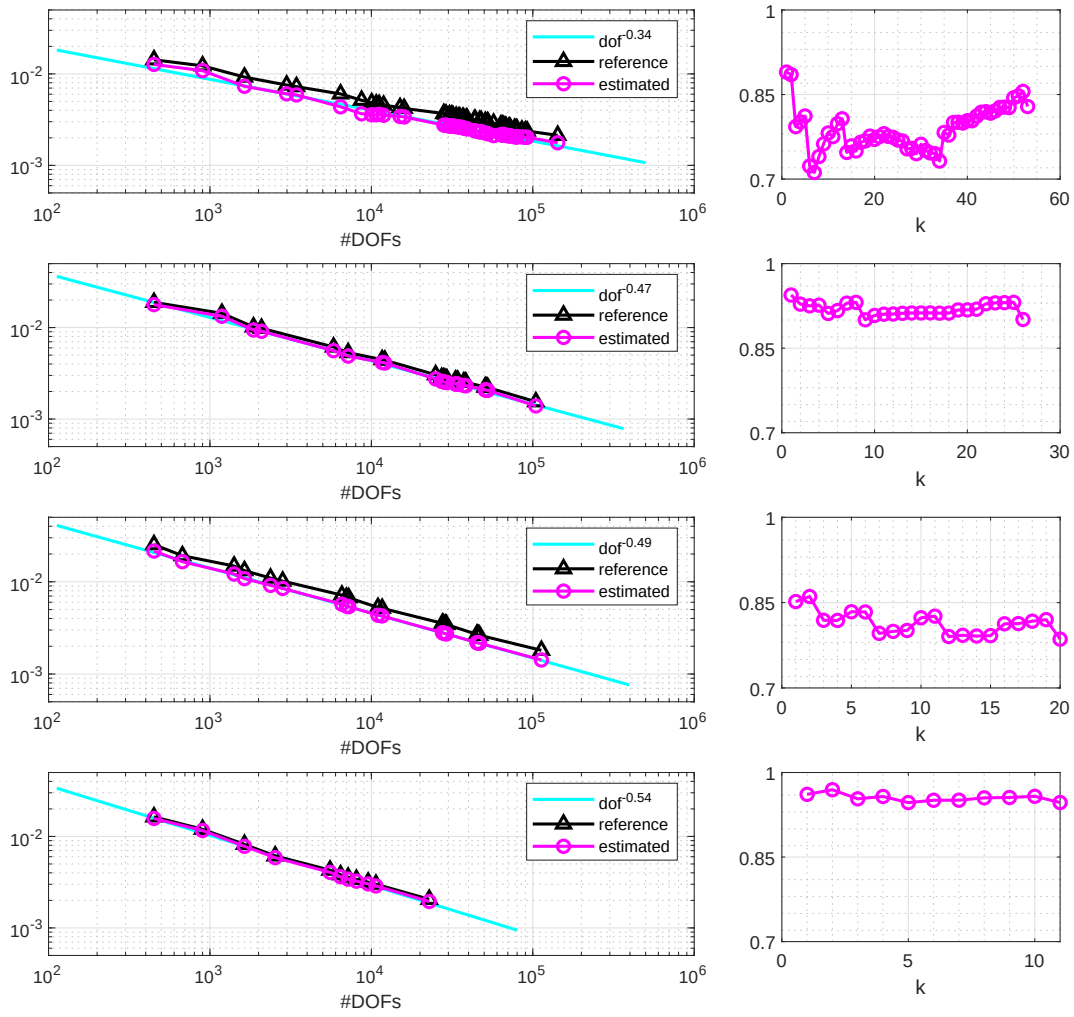


FIG. 6.1. *Plots of the estimated errors η^k versus number of degrees of freedom N_{dof} (left) at steps $k = 0, 1, \ldots$ and effectivity indices θ^k (right) when solving TP.1–TP.4 (top-to-bottom) using Algorithms 1 and 2 (version 1).*

TABLE 6.2
Number of solution modes assigned the same element width $h(\ell_K^\mu)$ (correspondong to a mesh level number ℓ_K^μ in $\boldsymbol{\ell}^K$) for test problems TP.1–TP.4.

Test Problem	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	card(J_P^K)	M
TP.1	118	49	1	0	1	0	169	93
TP.2	–	25	6	3	1	1	36	13
TP.3	–	5	7	2	2	1	17	3
TP.4	–	17	3	0	1	0	21	8

of multi-indices within that set that are assigned the same finite element space (i.e., the same mesh level number from the set $\boldsymbol{\ell}^K$). In each case, we observe that fine meshes are required to estimate very few solution modes (polynomial coefficients), whereas higher numbers of modes are assigned coarse meshes. This is reminiscent of multilevel sampling methods. While multilevel

TABLE 6.3
A subset of 12 *multi-indices from the set* J_P^K *generated by Algorithm 1 and the associated element widths* $h(\ell_K^\mu)$ *assigned to those multi-indices at the final step for test problems TP.1–TP4.*

TP.1		TP.2		TP.3		TP.4	
μ	$h(\ell_K^\mu)$	μ	$h(\ell_K^\mu)$	μ	$h(\ell_K^\mu)$	μ	$h(\ell_K^\mu)$
(0 0 0 0 0 0 0 0 0 0)	2^{-7}	(0 0 0 0 0 0)	2^{-8}	(0 0 0)	2^{-8}	(0 0 0 0 0 0)	2^{-7}
(**1** 0 0 0 0 0 0 0 0 0)	2^{-5}	(**1** 0 0 0 0 0)	2^{-7}	(**1** 0 0)	2^{-7}	(**1** 0 0 0 0 0)	2^{-5}
(0 0 **1** 0 0 0 0 0 0 0)	2^{-4}	(0 0 **1** 0 0 0)	2^{-6}	(**2** 0 0)	2^{-7}	(0 0 **1** 0 0 0)	2^{-5}
(0 **1** 0 0 0 0 0 0 0 0)	2^{-4}	(0 **1** 0 0 0 0)	2^{-6}	(**3** 0 0)	2^{-6}	(0 **1** 0 0 0 0)	2^{-5}
(0 0 0 0 0 **1** 0 0 0 0)	2^{-4}	(**2** 0 0 0 0 0)	2^{-6}	(0 **1** 0)	2^{-5}	(0 0 0 **1** 0 0)	2^{-4}
(0 0 0 0 **1** 0 0 0 0 0)	2^{-4}	(**1** **1** 0 0 0 0)	2^{-5}	(**4** 0 0)	2^{-6}	(**1** 0 **1** 0 0 0)	2^{-4}
(0 0 0 **1** 0 0 0 0 0 0)	2^{-4}	(0 0 0 0 0 **1**)	2^{-5}	(**1** **1** 0)	2^{-5}	(**1** **1** 0 0 0 0)	2^{-4}
(**2** 0 0 0 0 0 0 0 0 0)	2^{-3}	(0 0 0 0 **1** 0)	2^{-5}	(**5** 0 0)	2^{-5}	(**2** 0 0 0 0 0)	2^{-4}
(0 0 0 0 0 0 0 **1** 0 0)	2^{-4}	(0 0 0 **1** 0 0)	2^{-5}	(**2** **1** 0)	2^{-5}	(0 0 0 0 0 **1**)	2^{-4}
(0 0 0 0 0 0 **1** 0 0 0)	2^{-4}	(**1** 0 **1** 0 0 0)	2^{-5}	(0 0 **1**)	2^{-5}	(0 0 0 0 **1** 0)	2^{-4}
(0 0 0 0 0 0 0 0 0 **1**)	2^{-4}	(**2** **1** 0 0 0 0)	2^{-4}	(**3** **1** 0)	2^{-5}	(**1** 0 0 **1** 0 0)	2^{-4}
(0 0 0 0 0 0 0 0 0 **1** 0)	2^{-4}	(**3** 0 0 0 0 0)	2^{-5}	(**6** 0 0)	2^{-5}	(0 **1** **1** 0 0 0)	2^{-4}

Monte Carlo and multilevel and multi-index stochastic collocation methods [11, 9, 29, 25, 24] also typically require few deterministic PDE solves using fine finite element meshes and larger numbers using coarser meshes, there are some differences. Multilevel sampling methods typically require the number of parameters to be fixed a priori. We stress that our algorithm requires no sampling and learns which are the important parameters to activate as part of the solution process itself. The decision about which meshes to use is based on an a rigorous a posteriori error estimate. For TP.1, we observe that many more parameters are activated ($M = 93$) and the number of polynomials required ($\text{card}(J_P^K) = 169$) is much higher than in test problems TP.2–TP.4. This is due to the slow decay of the eigenvalues λ_m in (6.1). Although many more polynomials are needed in TP.1, the majority of the corresponding meshes are coarse. Conversely, test problem TP.3 has the lowest number of activated parameters ($M = 3$) and requires the smallest number of polynomials ($\text{card}(J_P^K) = 17$). Compared to TP.1, however, a larger proportion of the meshes associated with the selected multi-indices are finer. For TP.2, the number of activated parameters is higher than in TP.3, as expected.

In Table 6.3 we display twelve of the multi-indices in the set J_P^K that are selected by Algorithm 1 for each test problem, as well as the associated element widths $h(\ell_K^\mu)$ assigned to those multi-indices, at the final step. Note that it is not possible to list all the multi-indices generated for all four test problems. The twelve shown in each case are selected in the first few iterations. For TP.1, these mostly correspond to univariate polynomials of degree one. In the early stages, Algorithm 1 selects multi-indices that activate more terms in the expansion (6.1), rather than multi-indices that correspond to polynomials of higher degree in the currently active parameters. Again, this is due to the slow decay of the λ_m in (6.1). In contrast, when solving TP.3, Algorithm 1 first selects multi-indices that correspond to polynomials of higher degree in the currently active parameters, before activating new parameters. For all test problems, the multi-indices that are selected in the early stages (corresponding to the most important solution modes, with respect to the energy error), are assigned the finest meshes. In particular, the *mean* solution mode is the coefficient of the polynomial associated with $\mu = (0, 0, \dots)$. This is always allocated the finest mesh.

6.3. Experiment 2 (timings). We now investigate the computational efficiency of the new method. All computations were performed in MATLAB using new software developed from components of the S-IFISS toolbox [3] on an Intel Core i7 4770k 3.50GHz CPU with 24GB of RAM.

TABLE 6.4

Solution times T (in seconds) and adaptive step counts K required to solve test problems TP.1–TP.4 using Algorithms 1 and 2 (versions 1 and 2) with various choices of the error tolerances ϵ. The symbol '–' denotes that the estimated error at the previous step is already below the tolerance and the preceeding T and K are applicable.

-	TP.1				TP.2				TP.3				TP.4			
	ver. 1		ver. 2		ver. 1		ver. 2		ver. 1		ver. 2		ver. 1		ver. 2	
ϵ	T	K	T	K	T	K	T	K	T	K	T	K	T	K	T	K
$4.5 \cdot 10^{-3}$	2	6	2	6	1	7	5	6	1	10	1	7	1	5	2	5
$3.0 \cdot 10^{-3}$	13	14	3	8	4	9	–	–	3	12	3	9	2	10	–	–
$1.5 \cdot 10^{-3}$	311	83	325	34	27	26	29	10	16	20	11	11	7	19	5	7
$9.0 \cdot 10^{-4}$	out of memory				236	70	167	13	87	36	62	15	23	29	22	8
$7.5 \cdot 10^{-4}$					–	–	–	–	100	38	–	–	36	38	–	–
$6.0 \cdot 10^{-4}$					881	147	–	–	147	44	92	18	110	48	80	9
$4.5 \cdot 10^{-4}$					2197	177	1306	19	484	61	340	22	158	59	95	10

In Table 6.4 we record timings (T) in seconds and the number of adaptive steps (K) taken by Algorithm 1 (using both versions of Algorithm 2 now), as we decrease the error tolerance ϵ. We observe that for TP.2–TP.4, for smaller error tolerances, using version 2 of Algorithm 2 results in a quicker solution time and a lower adaptive step count. The lower step count is due to the fact that the sets of multi-indices $\bar{J}_k$ that are produced by version 2 are usually richer than the ones produced by version 1. Note that because of this, a single step of version 2 is more expensive than a single step of version 1. Time savings are only made when enough steps are saved. We use the preconditioned conjugate gradient method with a mean–based preconditioner [27] to solve (3.5). Fewer adaptive steps means that fewer SGFEM linear systems have to be solved and hence fewer matrix–vector products (3.8) are required. For TP.1 with $\epsilon = 1.5 \times 10^{-3}$, the difference in step count between version 1 and 2 is not large enough for time savings to be made. We note also that asymptotically, both versions of Algorithm 2 result in the same rates of convergence (illustrated by the blue lines in Figure 6.1). However, due to the larger associated sets $\bar{J}_k$, version 2 requires more adaptive steps before this rate is realised.

In Figure 6.2 we plot the total computational time (T) against the the number of degrees of freedom (N_{dof}) when employing version 2 of Algorithm 2. The total number of markers, each reflecting a single step of Algorithm 1, is equal to the value of K corresponding to the smallest value of ϵ in Table 6.4. We observe that for all four test problems, the computational time behaves at most like $N_{\text{dof}}^{1.35}$. For TP.3 and TP.4, where M is smaller, T behaves almost linearly with respect to N_{dof}. We also plot the ratio r of the cumulative time taken to estimate the energy error (by executing the modules `COMPONENT_SPATIAL_ERRORS` and `PARAMETRIC_SPATIAL_ERRORS` in Algorithm 1) to the time taken to compute the SGFEM approximation u_X (by executing the `SOLVE` module in Algorithm 1). We observe that r does not grow with N_{dof} (indeed, $1.2 < r < 2.5$ at the final step for all four problems). Hence, the cost of estimating the error is proportional to the cost of computing the SGFEM approximation itself.

7. Summary. We presented a novel adaptive multilevel SGFEM algorithm for the numerical solution of elliptic PDEs with coefficients that depend on countably many parameters y_m in an affine way. A key feature is the use of an implicit a posteriori error estimation strategy to drive the adaptive enrichment of the approximation space. We demonstrated how to extend the error estimation strategy used in [4, 6] to the new multilevel setting and described new ways to utilise the distinct components of the error estimator to determine how to best enrich the spaces associated with the spatial and parameter domains. Through numerical experiments we demonstrated that the error estimate is accurate and that the resulting adaptive algorithm achieves the optimal rate

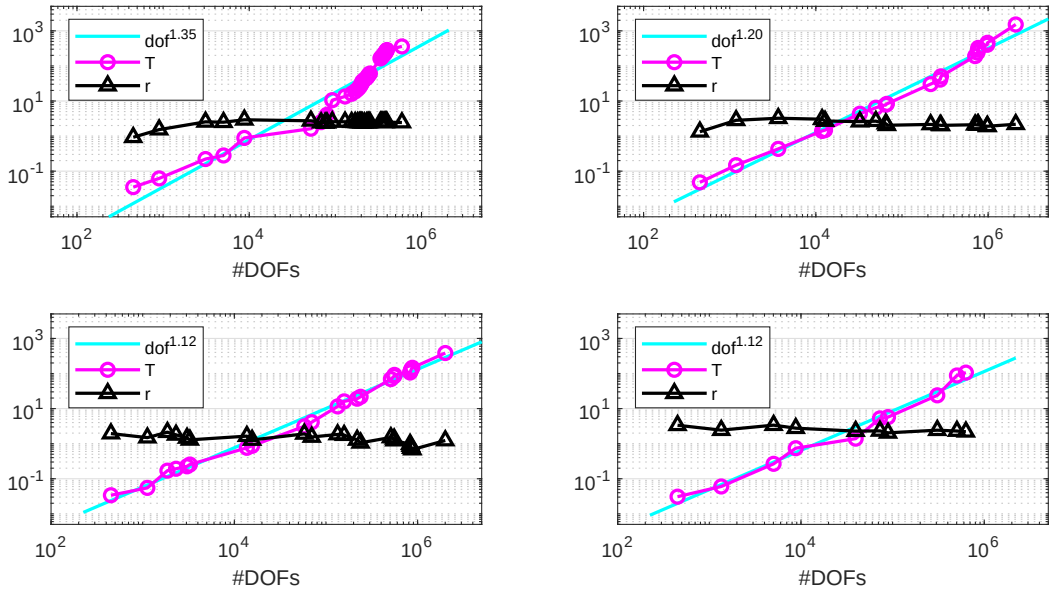


FIG. 6.2. *Plots of the total computational time T (round markers) in seconds accumulated over all refinement steps and the error estimation–solve time ratio r (triangular markers), versus the number of degrees of freedom N_{dof} when solving TP.1–TP.4 (left-to-right, top-to-bottom) using Algorithm 1 with version 2 of Algorithm 2.*

of convergence with respect to the dimension of the approximation space. That is, we achieve the convergence rate associated with the chosen finite element method for the associated parameter-free problems. Unlike other methods, our numerical scheme uses no marking or tuning parameters. Finally, we demonstrated that our multilevel algorithm is computationally efficient. Indeed, for some test problems (where the number M of parameters that need to be activated is not too high), the solution time scales almost linearly with respect to the dimension of the approximation space.

REFERENCES

[1] Mark Ainsworth and J. Tinsley Oden. *A posteriori error estimation in finite element analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2000.

[2] Ivo M. Babuška, Raúl Tempone, and Georgios E. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM J. Numer. Anal.*, 42(2):800–825, 2004.

[3] Alex Bespalov, Catherine E. Powell, and David Silvester. Stochastic IFISS (S-IFISS) version 1.1, 2016. Available online at `http://www.manchester.ac.uk/ifiss/s-ifiss1.0.tar.gz`.

[4] Alex Bespalov, Catherine E. Powell, and David Silvester. Energy norm a posteriori error estimation for parametric operator equations. *SIAM J. Sci. Comput.*, 36(2):A339–A363, 2014.

[5] Alex Bespalov and Leonardo Rocchi. Efficient adaptive algorithms for elliptic PDEs with random data. *SIAM/ASA J. Uncertain. Quantif.*, 6(1):243–272, 2018.

[6] Alex Bespalov and David Silvester. Efficient adaptive stochastic Galerkin methods for parametric operator equations. *SIAM J. Sci. Comput.*, 38(4):A2118–A2140, 2016.

[7] Marcel Bieri, Roman Andreev, and Christoph Schwab. Sparse tensor discretization of elliptic SPDEs. *SIAM J. Sci. Comput.*, 31(6):4281–4304, 2009/10.

[8] Marcel Bieri and Christoph Schwab. Sparse high order FEM for elliptic sPDEs. *Comput. Methods Appl. Mech. Engrg.*, 198(13-14):1149–1170, 2009.

[9] J. Charrier, R. Scheichl, and A. L. Teckentrup. Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods. *SIAM J. Numer. Anal.*, 51(1):322–352, 2013.

[10] Abdellah Chkifa, Albert Cohen, and Christoph Schwab. Breaking the curse of dimensionality in sparse polynomial approximation of parametric PDEs. *J. Math. Pures Appl. (9)*, 103(2):400–428, 2015.

[11] 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(1):3–15, 2011.

[12] Albert Cohen, Ronald DeVore, and Christoph Schwab. Convergence rates of best N-term Galerkin approximations for a class of elliptic sPDEs. *Found. Comput. Math.*, 10(6):615–646, 2010.

[13] Albert Cohen, Ronald Devore, and Christoph Schwab. Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's. *Anal. Appl. (Singap.)*, 9(1):11–47, 2011.

[14] Adam J. Crowder and Catherine E. Powell. CBS constants and their role in error estimation for stochastic Galerkin finite element methods. Manchester Institute for Mathematical Sciences, The University of Manchester, Manchester, UK. Electronically published at `http://eprints.ma.man.ac.uk/2549/`.

[15] Manas K. Deb, Ivo M. Babuška, and J. Tinsley Oden. Solution of stochastic partial differential equations using Galerkin finite element techniques. *Comput. Methods Appl. Mech. Engrg.*, 190(48):6359–6372, 2001.

[16] Willy Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM J. Numer. Anal.*, 33(3):1106–1124, 1996.

[17] Martin Eigel, Claude J. Gittelson, Christoph Schwab, and Elmar Zander. A convergent adaptive stochastic Galerkin finite element method with quasi-optimal spatial meshes. *ESAIM Math. Model. Numer. Anal.*, 49(5):1367–1398, 2015.

[18] Martin Eigel, Claude Jeffrey Gittelson, Christoph Schwab, and Elmar Zander. Adaptive stochastic Galerkin FEM. *Comput. Methods Appl. Mech. Engrg.*, 270:247–269, 2014.

[19] Martin Eigel and Christian Merdon. Local equilibration error estimators for guaranteed error control in adaptive stochastic higher-order Galerkin finite element methods. *SIAM/ASA J. Uncertain. Quantif.*, 4(1):1372–1397, 2016.

[20] Oliver G. Ernst and Elisabeth Ullmann. Stochastic Galerkin matrices. *SIAM J. Matrix Anal. Appl.*, 31(4):1848–1872, 2009/10.

[21] Roger G. Ghanem and Pol D. Spanos. *Stochastic finite elements: a spectral approach*. Springer-Verlag, New York, 1991.

[22] Claude J. Gittelson. An adaptive stochastic Galerkin method for random elliptic operators. *Math. Comp.*, 82(283):1515–1541, 2013.

[23] Claude Jeffrey Gittelson. Convergence rates of multilevel and sparse tensor approximations for a random elliptic PDE. *SIAM J. Numer. Anal.*, 51(4):2426–2447, 2013.

[24] Abdul-Lateef Haji-Ali, Fabio Nobile, Lorenzo Tamellini, and Raúl Tempone. Multi-index stochastic collocation convergence rates for random PDEs with parametric regularity. *Found. Comput. Math.*, 16(6):1555–1605, 2016.

[25] Abdul-Lateef Haji-Ali, Fabio Nobile, Lorenzo Tamellini, and Raúl Tempone. Multi-index stochastic collocation for random PDEs. *Comput. Methods Appl. Mech. Engrg.*, 306:95–122, 2016.

[26] Gabriel J. Lord, Catherine E. Powell, and Tony Shardlow. *An introduction to computational stochastic PDEs*. Cambridge Texts in Applied Mathematics. Cambridge University Press, New York, 2014.

[27] Catherine E. Powell and Howard C. Elman. Block-diagonal preconditioning for spectral stochastic finite-element systems. *IMA J. Numer. Anal.*, 29(2):350–375, 2009.

[28] Ivana Pultarová. Adaptive algorithm for stochastic Galerkin method. *Appl. Math.*, 60(5):551–571, 2015.

[29] Aretha L. Teckentrup, Peter Jantsch, Clayton G. Webster, and M. Gunzburger. A multilevel stochastic collocation method for partial differential equations with random input data. *SIAM/ASA J. Uncertain. Quantif.*, 3(1):1046–1074, 2015.

[30] Radu A. Todor and Christoph Schwab. Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients. *IMA J. Numer. Anal.*, 27(2):232–261, 2007.