On solving stochastic collocation systems with algebraic multigrid

ANDREW D. GORDON AND CATHERINE E. POWELL*

School of Mathematics, University of Manchester, Oxford Road, Manchester M13 9PL, UK

*Corresponding author: c.powell@manchester.ac.uk

andrew.gordon-2@postgrad.manchester.ac.uk

[Received on 14 October 2010; revised on 21 March 2011]

Stochastic collocation methods facilitate the numerical solution of partial differential equations (PDEs) with random data and give rise to long sequences of similar linear systems. When elliptic PDEs with random diffusion coefficients are discretized with mixed finite element methods in the physical domain we obtain saddle point systems. These are trivial to solve when considered individually; the challenge lies in exploiting their similarities to recycle information and minimize the cost of solving the entire sequence. We apply stochastic collocation to a model stochastic elliptic problem and discretize in physical space using Raviart–Thomas elements. We propose an efficient solution strategy for the resulting linear systems that is more robust than any other in the literature. In particular, we show that it is feasible to use finely-tuned algebraic multigrid preconditioning if key set-up information is reused. The proposed solver is robust with respect to variations in the discretization and statistical parameters for stochastically linear and nonlinear data.

Keywords: stochastic collocation; sparse grids; mixed finite elements; preconditioning; algebraic multigrid.

1. Introduction

Our starting point is the mixed formulation of the steady-state diffusion problem,

$$\begin{aligned} a^{-1}(\mathbf{x}, \omega) \mathbf{u}(\mathbf{x}, \omega) + \nabla p(\mathbf{x}, \omega) &= 0 & \text{in } D \times \Omega, \\ \nabla \cdot \mathbf{u}(\mathbf{x}, \omega) &= f(\mathbf{x}) & \text{in } D \times \Omega, \\ p(\mathbf{x}, \omega) &= 0 & \text{on } \partial D \times \Omega, \end{aligned} \quad (1.1)$$

which arises in models of potential flow when there is uncertainty in a . Here $a : D \times \Omega \rightarrow \mathbb{R}$ is a random field, $D \subset \mathbb{R}^2$ is the bounded physical domain and Ω is a sample space from a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. We assume that $a \in L^2(\Omega, L^2(D))$ and that the mean $\mu(\mathbf{x}) = \mathbb{E}[a(\mathbf{x}, \omega)]$ and covariance function

$$C(\mathbf{x}, \hat{\mathbf{x}}) = \mathbb{E}[(a(\mathbf{x}, \omega) - \mu(\mathbf{x}))(a(\hat{\mathbf{x}}, \omega) - \mu(\hat{\mathbf{x}}))] = \sigma^2 V(\mathbf{x}, \hat{\mathbf{x}}) \quad (1.2)$$

are known. In addition, we assume that $\sigma^2 = C(\mathbf{x}, \mathbf{x})$, which is the variance of $a(\mathbf{x}, \omega)$, is constant. The function $V : D \times D \rightarrow \mathbb{R}$ typically depends on a parameter $\gamma > 0$ called the correlation length, which controls the spatial correlation between the random variables $a(\mathbf{x}, \omega)$ and $a(\hat{\mathbf{x}}, \omega)$. For example,

$$C(\mathbf{x}, \hat{\mathbf{x}}) = \sigma^2 \exp(-\|\mathbf{x} - \hat{\mathbf{x}}\|_1 / \gamma). \quad (1.3)$$

Other commonly used covariance functions can be found in Frauenfelder *et al.* (2005).

We follow the now well-established procedure (see Deb *et al.*, 2001; Babuška *et al.*, 2004) of assuming that $a(\mathbf{x}, \omega)$ can be well approximated by a function of M independent random variables $\xi_k(\omega) \in L^2(\Omega)$. A common choice is a (truncated) Karhunen–Loève (KL) expansion (see Loève, 1978),

$$a_M(\mathbf{x}, \boldsymbol{\xi}) = \mu(\mathbf{x}) + \sigma \sum_{k=1}^M \sqrt{\lambda_k} c_k(\mathbf{x}) \xi_k(\omega), \quad \boldsymbol{\xi} := (\xi_1(\omega), \dots, \xi_M(\omega)). \quad (1.4)$$

Here $\mathbb{E}[\xi_k] = 0$, $\mathbb{E}[\xi_k \xi_l] = \delta_{k,l}$ and $\{\lambda_k, c_k(\mathbf{x})\}_{k=1}^M$ are the leading eigenpairs of the integral operator associated with $V(\mathbf{x}, \hat{\mathbf{x}})$ in (1.2). We note that $\lambda_k \rightarrow 0$ as $k \rightarrow \infty$ at a rate that depends on the regularity of $C(\mathbf{x}, \hat{\mathbf{x}})$ (see Frauenfelder *et al.*, 2005). Speaking generally, the smaller the correlation length γ , the larger the number of terms needed to obtain a good approximation to $a(\mathbf{x}, \omega)$. Alternatively, a statistical distribution may be assumed for $b(\mathbf{x}, \omega) = \log(a(\mathbf{x}, \omega))$. If the mean $\mu(\mathbf{x})$ and the covariance function $C(\mathbf{x}, \hat{\mathbf{x}})$ of $b(\mathbf{x}, \omega)$ are available, $a(\mathbf{x}, \omega)$ may also be approximated via

$$a_M(\mathbf{x}, \boldsymbol{\xi}) = \exp \left(\mu(\mathbf{x}) + \sigma \sum_{k=1}^M \sqrt{\lambda_k} c_k(\mathbf{x}) \xi_k(\omega) \right). \quad (1.5)$$

We consider (1.1) with a approximated by either (1.4) or (1.5), which we will refer to as the stochastically linear and nonlinear model problems, respectively.

Let $\xi_k(\Omega) = \Gamma_k \subseteq \mathbb{R}$ and denote the probability density function of ξ_k by ρ_k . For example, if ξ_k is uniformly distributed then $\Gamma_k = [-\sqrt{3}, \sqrt{3}]$. We assume the ξ_k are independent so that the joint density function of $\boldsymbol{\xi}$ is $\rho(\mathbf{y}) = \prod_{k=1}^M \rho_k(y_k)$, where $y_k = \xi_k(\omega) \in \Gamma_k$ and $\mathbf{y} \in \Gamma = \Gamma_1 \times \dots \times \Gamma_M \subseteq \mathbb{R}^M$. Recall now that for any measurable function f of the multivariate random variable $\boldsymbol{\xi}$,

$$\mathbb{E}[f(\boldsymbol{\xi})] = \int_{\Omega} f(\boldsymbol{\xi}(\omega)) \, d\mathbb{P}(\omega) = \int_{\Gamma} f(\mathbf{y}) \rho(\mathbf{y}) \, d\mathbf{y}.$$

Replacing $a(\mathbf{x}, \omega)$ by $a_M(\mathbf{x}, \boldsymbol{\xi})$ in (1.1), writing down the weak form, and taking expectations then results in an $(M + 2)$ -dimensional deterministic problem (Ganis *et al.*, 2008; Ernst *et al.*, 2009; Elman *et al.*, 2010).

We seek $p(\mathbf{x}, \mathbf{y}) \in Q$ and $\mathbf{u}(\mathbf{x}, \mathbf{y}) \in \mathbf{V}$ satisfying

$$\begin{aligned} \int_{\Gamma} \rho(\mathbf{y}) \int_D a_M^{-1}(\mathbf{x}, \mathbf{y}) \mathbf{u} \cdot \mathbf{v} \, d\mathbf{x} \, d\mathbf{y} - \int_{\Gamma} \rho(\mathbf{y}) \int_D p \nabla \cdot \mathbf{v} \, d\mathbf{x} \, d\mathbf{y} &= 0, \\ - \int_{\Gamma} \rho(\mathbf{y}) \int_D q \nabla \cdot \mathbf{u} \, d\mathbf{x} \, d\mathbf{y} &= - \int_{\Gamma} \rho(\mathbf{y}) \int_D f q \, d\mathbf{x} \, d\mathbf{y} \end{aligned}$$

for all $\mathbf{v}(\mathbf{x}, \mathbf{y}) \in \mathbf{V}$ and $q(\mathbf{x}, \mathbf{y}) \in Q$, where

$$Q = L^2_{\rho}(\Gamma, L^2(D)) := \left\{ q : D \times \Gamma \rightarrow \mathbb{R} \left| \int_{\Gamma} \rho(\mathbf{y}) \|q\|_{L^2(D)}^2 \, d\mathbf{y} < \infty \right. \right\},$$

$$\mathbf{V} = L^2_{\rho}(\Gamma, \mathbf{H}_{\text{div}}(D)) := \left\{ \mathbf{v} : D \times \Gamma \rightarrow \mathbb{R}^2 \left| \int_{\Gamma} \rho(\mathbf{y}) \left(\|\mathbf{v}\|_{L^2(D)}^2 + \|\nabla \cdot \mathbf{v}\|_{L^2(D)}^2 \right) \, d\mathbf{y} < \infty \right. \right\}.$$

If we assume $0 < a_1 \leq a_M(\mathbf{x}, \mathbf{y}) \leq a_2 < \infty$ a.e. in $D \times \Gamma$ then a unique solution exists. Implementing a mixed finite element discretization on D then leads to the semidiscrete saddle point problem: find $p_h(\mathbf{x}, \mathbf{y}) \in Q_h = L^2_\rho(\Gamma, \Psi_h) \subset Q$ and $\mathbf{u}_h(\mathbf{x}, \mathbf{y}) \in \mathbf{V}_h = L^2_\rho(\Gamma, \Upsilon_h) \subset \mathbf{V}$ satisfying

$$\int_\Gamma \rho(\mathbf{y}) \int_D a_M^{-1}(\mathbf{x}, \mathbf{y}) \mathbf{u}_h \cdot \mathbf{v} \, d\mathbf{x} \, d\mathbf{y} - \int_\Gamma \rho(\mathbf{y}) \int_D p_h \nabla \cdot \mathbf{v} \, d\mathbf{x} \, d\mathbf{y} = 0, \quad (1.6)$$

$$- \int_\Gamma \rho(\mathbf{y}) \int_D q \nabla \cdot \mathbf{u}_h \, d\mathbf{x} \, d\mathbf{y} = - \int_\Gamma \rho(\mathbf{y}) \int_D f q \, d\mathbf{x} \, d\mathbf{y} \quad (1.7)$$

for all $\mathbf{v}(\mathbf{x}, \mathbf{y}) \in \mathbf{V}_h$ and $q(\mathbf{x}, \mathbf{y}) \in Q_h$. Here $\Psi_h \subset L^2(D)$ with $\dim(\Psi_h) = m_h$ and $\Upsilon_h \subset \mathbf{H}_{\text{div}}(D)$ with $\dim(\Upsilon_h) = l_h$ are any pair of finite element spaces satisfying the standard inf-sup condition on $\mathbf{H}_{\text{div}}(D) \times L^2(D)$.

After fixing a spatial discretization, we can tackle (1.6)–(1.7) with Monte Carlo methods (MCMs), stochastic Galerkin methods (SGMs) (Ghanem & Spanos, 1991; Deb *et al.*, 2001; Babuška *et al.*, 2004) or stochastic collocation methods (SCMs) (Xiu & Hesthaven, 2005; Babuška *et al.*, 2007; Nobile *et al.*, 2008). Standard MCMs approximate $\mathbb{E}[p_h]$ and $\mathbb{E}[\mathbf{u}_h]$ by the sample average at randomly chosen points $\mathbf{y}_r \in \Gamma$. For each sample $a_M^r(\mathbf{x}) = a_M(\mathbf{x}, \mathbf{y}_r)$, $p_h^r(\mathbf{x}) = p_h(\mathbf{x}, \mathbf{y}_r) \in \Psi_h$ and $\mathbf{u}_h^r(\mathbf{x}) = \mathbf{u}_h(\mathbf{x}, \mathbf{y}_r) \in \Upsilon_h$ satisfy

$$\int_D a_M^r(\mathbf{x})^{-1} \mathbf{u}_h^r \cdot \mathbf{v} \, d\mathbf{x} - \int_D p_h^r \nabla \cdot \mathbf{v} \, d\mathbf{x} = 0 \quad \forall \mathbf{v}(\mathbf{x}) \in \Upsilon_h, \quad (1.8)$$

$$- \int_D q \nabla \cdot \mathbf{u}_h^r \, d\mathbf{x} = - \int_D f q \, d\mathbf{x} \quad \forall q(\mathbf{x}) \in \Psi_h, \quad (1.9)$$

leading to a sequence of decoupled deterministic saddle point problems

$$\underbrace{\begin{pmatrix} K_r & B^T \\ B & 0 \end{pmatrix}}_{A_r} \underbrace{\begin{pmatrix} \mathbf{u}_r \\ \mathbf{p}_r \end{pmatrix}}_{\mathbf{q}_r} = \underbrace{\begin{pmatrix} \mathbf{0} \\ f \end{pmatrix}}_{\mathbf{g}}, \quad r = 1, 2, \dots, \quad (1.10)$$

which can be solved with state-of-the-art deterministic solvers. However, the sample average is slow to converge, and alternative discretization schemes have gained much attention recently.

SGMs (Ernst *et al.*, 2009; Elman *et al.*, 2010) seek $p_{hd} \in \Psi_h \otimes S_d$ and $\mathbf{u}_{hd} \in \Upsilon_h \otimes S_d$ with $S_d \subset L^2_\rho(\Gamma)$. They have favourable convergence rates for low values of M (Babuška *et al.*, 2004), but yield a single linear system of dimension $(l_h + m_h) \times \dim(S_d)$ that often cannot be decoupled. Fully robust cheap solvers for these systems are lacking (Powell & Ullmann, 2010). If S_d consists of complete polynomials, the Galerkin equations must be solved simultaneously (Ernst & Ullmann, 2010). At the expense of the convergence rate, tensor product polynomials can also be used. In that case, if $a_M(\mathbf{x}, \boldsymbol{\xi})$ is linear in the variables ξ_k , as in (1.4), then S_d possesses a basis of so-called doubly orthogonal polynomials that makes the Galerkin matrix block-diagonal, yielding a set of decoupled problems as in (1.10). SCMs provide a happy medium. Like MCMs they sample the finite element solution $(\mathbf{u}_h, p_h)$ and lead to a sequence of decoupled systems (1.10) for both the stochastically linear and nonlinear model problems. Furthermore, their convergence is comparable to that of SGMs (Bäck *et al.*, 2011;

Elman *et al.*, 2011). In the stochastic Galerkin approach, however, it is necessary to assume that a_M^{-1} is a truncated KL expansion and that a covariance function is known for a^{-1} . SCMs do not impose this restriction.

1.1 Solution strategies

Algebraic multigrid (AMG) (Ruge & Stüben, 1987; Stüben, 2000) is a widely-used solver for positive-definite linear systems arising from discretizations of elliptic partial differential equations (PDEs), that is highly robust with respect to variations in diffusion coefficients. In Powell & Silvester (2004) AMG is used as a building block for a robust preconditioner for indefinite systems of the form (1.10). When the number of systems is large it may be infeasible to fine-tune a preconditioner to the individual matrices A_r ; the one-preconditioner-fits-all approach has merit. This reduces set-up costs but the preconditioner may be so weak for some systems that no savings are made overall. If the systems are similar there is scope for saving work by recycling preconditioner components using previous solutions as initial guesses and/or recycling Krylov subspace vectors (Parks *et al.*, 2006).

Jin *et al.* (2007) and Ullmann (2008) study sequences of *positive-definite* systems that arise when the primal version of the model problem is discretized with an SGM based on doubly orthogonal polynomials and apply Krylov subspace recycling. Only stochastically *linear* coefficients (1.4) are considered, which leads to fairly well-conditioned highly similar matrices. Most recycled Krylov subspace solvers are suboptimal for individual systems but have benefits when applied to a long similar sequence if a cheap but *weak* preconditioner is applied. The domain decomposition preconditioner in Jin *et al.* (2007) is optimal for most systems but weak for a certain subset. In Ullmann (2008) one V-cycle of AMG applied to the ‘mean’ stiffness matrix is used to precondition all systems but efficiency deteriorates significantly when σ in (1.2) is large relative to $\mu(\mathbf{x})$. Ullmann (2008) and Ernst *et al.* (2009) discuss saddle point systems that arise when the mixed problem (but with a_M^{-1} as a KL expansion) is discretized with an SGM based on doubly orthogonal polynomials. Again, the matrices are fairly well conditioned in that case. A weak preconditioner based on ideas in Powell & Silvester (2004) is considered and in Ullmann (2008) the benefits of Krylov subspace recycling are investigated. Stochastically nonlinear data and strong preconditioners are not considered.

We focus on SCMs and emphasize strong, cheap preconditioners and the reuse of preconditioner information. We propose an efficient way to solve the entire sequence (1.10) with the minimal residual method (MINRES) (Paige & Saunders, 1975) using AMG-based preconditioners. Unlike previous SGM studies our method handles (1.4) and (1.5) well, treats the diffusion coefficient properly, and is robust with respect to variations in the discretization and statistical parameters.

2. Stochastic collocation methods

SCMs collocate the semidiscrete problem (1.6)–(1.7) on a set of points $\mathbf{y}_1, \dots, \mathbf{y}_{n_c}$ in the hypercube Γ . A global approximation is then obtained by performing Lagrange interpolation. We obtain

$$\mathbf{u}_{hd}(\mathbf{x}, \mathbf{y}) = \sum_{r=1}^{n_c} \mathbf{u}_h^r(\mathbf{x}) L_r(\mathbf{y}), \quad p_{hd}(\mathbf{x}, \mathbf{y}) = \sum_{r=1}^{n_c} p_h^r(\mathbf{x}) L_r(\mathbf{y}), \quad (2.1)$$

where each $(\mathbf{u}_h^r(\mathbf{x}), p_h^r(\mathbf{x})) \in \mathcal{T}_h \times \Psi_h$ satisfies (1.8)–(1.9) at $\mathbf{y}_r \in \Gamma$, and $L_r(\mathbf{y})$ is a multivariate Lagrange polynomial. By construction, $\mathbf{u}_{hd}(\mathbf{x}, \mathbf{y}) \in \mathcal{T}_h \otimes S_d$ and $p_{hd}(\mathbf{x}, \mathbf{y}) \in \Psi_h \otimes S_d$, where $S_d = \text{span}\{L_1(\mathbf{y}), \dots, L_{n_c}(\mathbf{y})\} \subset L_\rho^2(\Gamma)$ and $\dim(S_d) = n_c$. Unlike standard MCMs, which choose points

$\mathbf{y}_r \in \Gamma$ randomly, good SCMs choose as few points as possible, in a structured way, to minimize the interpolation error.

Full tensor SCMs (Xiu & Hesthaven, 2005; Babuška *et al.*, 2007) use Cartesian products of M sets of interpolation points on the one-dimensional intervals Γ_k . Possibilities include Clenshaw–Curtis (CC) points (Clenshaw & Curtis, 1960) and Gauss points. If $d_k + 1$ points are selected on Γ_k , $n_c = \prod_{k=1}^M (d_k + 1)$ and this quickly becomes intractable as M increases. More sophisticated sparse grid SCMs (Xiu & Hesthaven, 2005; Nobile *et al.*, 2008) are based on interpolation and cubature rules for high-dimensional problems (Novak & Ritter, 1996; Gerstner & Griebel, 1998; Barthelmann *et al.*, 2000) derived from the work Smolyak (1963).

Let Z_i be a given set of one-dimensional interpolation points on Γ_k of size $m_i + 1$ where for $i \in \mathbb{N}$,

$$m_i = \begin{cases} 0 & \text{if } i = 1, \\ 2^{i-1} & \text{if } i > 1. \end{cases}$$

Writing $\mathbf{i} = (i_1, \dots, i_M) \in \mathbb{N}^M$, and given an approximation level l , the sparse grid on Γ is defined via

$$H(l, M) = \bigcup_{l \leq \|\mathbf{i}\|_1 \leq l+M} Z_{i_1} \times \dots \times Z_{i_M}. \quad (2.2)$$

If $Z_i \subset Z_{i+1}$, as is the case for CC points (see Fig. 1) then we require only $\|\mathbf{i}\|_1 = l + M - 1$ in (2.2). The convergence rate of any multivariate interpolation scheme depends on the largest value d for which the rule is exact for all polynomials of total degree d and sparse grid SCMs achieve total degree d accuracy with $l = d + 1$ (Barthelmann *et al.*, 2000, Theorem 4) using far fewer points than full tensor SCMs (see Table 1).

In Ganis *et al.* (2008) basic full tensor collocation is applied to (1.6)–(1.7), and standard spatial error estimates for Raviart–Thomas finite elements are combined with interpolation error estimates from Babuška *et al.* (2007) to obtain *a priori* error estimates for $\mathbf{u} - \mathbf{u}_{hd}$ and $p - p_{hd}$. Sparse grid interpolation error estimates from Nobile *et al.* (2008) can be applied in the same way. Our concern is the cost of computing (2.1).

FIG. 1. Sparse grid CC points on $[-\sqrt{3}, \sqrt{3}]^2$ with $l = 3$, $n_c = 13$ (left), $l = 4$, $n_c = 29$ (centre), $l = 5$, $n_c = 65$ (right).

TABLE 1 *Dimension of S_d (or n_c) for varying M and d using CC points*

M	5		10		20	
d	1	3	1	3	1	3
Sparse grid	11	241	21	1581	41	11561
Full tensor	32	1024	1024	10^6	10^6	10^{12}

3. AMG-based preconditioning

Each $(\mathbf{u}_h^r(\mathbf{x}), p_h^r(\mathbf{x}))$ in (2.1) solves (1.8)–(1.9), where $a_M^r(\mathbf{x})$ is (1.4) or (1.5) sampled at $\mathbf{y}_r$. For the spatial discretization we choose lowest-order Raviart–Thomas elements. Then $\Psi_h = \text{span}\{\psi_1(\mathbf{x}), \dots, \psi_{m_h}(\mathbf{x})\}$ and $\mathcal{T}_h = \text{span}\{\boldsymbol{\varphi}_1(\mathbf{x}), \dots, \boldsymbol{\varphi}_{l_h}(\mathbf{x})\}$ consist of piecewise constant functions, and vector functions with piecewise linear components. We assume that the finite element mesh is shape-regular and quasi-uniform, with characteristic mesh width h . This yields n_c sparse linear systems (1.10) where, for $i, j = 1, \dots, l_h$ and $k = 1, \dots, m_h$,

$$[K_r]_{ij} = \int_D a_M^r(\mathbf{x})^{-1} \boldsymbol{\varphi}_i(\mathbf{x}) \cdot \boldsymbol{\varphi}_j(\mathbf{x}) \, d\mathbf{x}, \quad [B]_{ki} = - \int_D \psi_k(\mathbf{x}) \nabla \cdot \boldsymbol{\varphi}_i(\mathbf{x}) \, d\mathbf{x}. \quad (3.1)$$

The properties of $a_M^r(\mathbf{x})$, which appears in K_r , are important for our solution strategy. We assume that

$$0 < a_{1,r} \leq a_M^r(\mathbf{x}) \leq a_{2,r} < \infty \quad \text{a.e. in } D, \text{ for } r = 1, 2, \dots, n_c. \quad (3.2)$$

If the covariance $C(\mathbf{x}, \hat{\mathbf{x}}) \in C(\bar{D} \times \bar{D})$ then the eigenfunctions $c_k(\mathbf{x})$ of the associated integral operator belong to $C(\bar{D})$ (Riesz & Sz. Nagy, 1990, pp. 245). Hence, if $\mu(\mathbf{x}) \in C(\bar{D})$ then for M fixed we can assume that our samples $a_M^r(\mathbf{x})$ are continuous. To simplify presentation, unless otherwise stated, we assume $\mu(\mathbf{x}) = \mu \in \mathbb{R}$. If (3.2) holds then each K_r is positive definite and each saddle point system can be solved iteratively with MINRES. However, each A_r is ill conditioned with respect to h and $a_M^r(\mathbf{x})$ (e.g. see Powell & Silvester, 2004) and preconditioning is essential. The dependence on $a_M^r(\mathbf{x})$ comes from K_r and $\kappa(K_r) \lesssim a_{2,r} a_{1,r}^{-1}$. This varies with the choice of collocation point $\mathbf{y}_r := (y_1^r, \dots, y_M^r)$ and some matrices are ‘nicer’ than others. Consider $\mathbf{y}_r = \mathbf{0}$, which sits at the centre of the hypercube Γ (when each density ρ_k is symmetric about zero). The associated matrix K_r , which we denote K_μ , is the so-called *mean mass matrix* and $\kappa(K_\mu) = \mathcal{O}(1)$. For an arbitrary $\mathbf{y}_r$ if a_M is given by (1.4), $a_{1,r} = \mu + \sigma m_r$ and $a_{2,r} = \mu + \sigma M_r$ in (3.2) where

$$m_r := \inf_{\mathbf{x} \in D} \sum_{k=1}^M \sqrt{\lambda_k} c_k(\mathbf{x}) y_k^r, \quad M_r := \sup_{\mathbf{x} \in D} \sum_{k=1}^M \sqrt{\lambda_k} c_k(\mathbf{x}) y_k^r.$$

Similarly, if a_M is given by (1.5), $a_{1,r} = \exp(\mu) \exp(\sigma m_r)$ and $a_{2,r} = \exp(\mu) \exp(\sigma M_r)$. In both cases the smaller m_r and the larger M_r , the more ill conditioned is the matrix K_r (and hence A_r).

3.1 Finely-tuned AMG preconditioning

For each saddle point matrix A_r we start by considering individual block-diagonal preconditioners

$$P_r = \begin{pmatrix} D_r & 0 \\ 0 & V_r \end{pmatrix}, \quad r = 1, 2, \dots, n_c. \quad (3.3)$$

Here D_r is any diagonal approximation to K_r with positive entries, and V_r is any symmetric positive-definite approximation to $S_r = BD_r^{-1}B^T$. Corollary 3.1, which follows from Powell & Silvester (2004, Corollary 3.4), provides bounds for the eigenvalues of the preconditioned matrix $P_r^{-1}A_r$, for each r .

COROLLARY 3.1 Denote by $\alpha_{1,r}$ and $\alpha_{2,r}$ the minimum and maximum eigenvalues of $D_r^{-1}K_r$, respectively, and by θ_r and Θ_r the minimum and maximum eigenvalues of $V_r^{-1}S_r$. The eigenvalues of $P_r^{-1}A_r$, for each $r = 1, 2, \dots, n_c$, lie in the intervals

$$\left[\frac{1}{2} \left(\alpha_{1,r} - \sqrt{\alpha_{1,r}^2 + 4\Theta_r} \right), \frac{1}{2} \left(\alpha_{2,r} - \sqrt{\alpha_{2,r}^2 + 4\Theta_r} \right) \right] \cup \left[\alpha_{1,r}, \frac{1}{2} \left(\alpha_{2,r} + \sqrt{\alpha_{2,r}^2 + 4\Theta_r} \right) \right]. \quad (3.4)$$

For lowest-order Raviart–Thomas elements, if (3.2) holds, each $S_r = BD_r^{-1}B^T$ is an M -matrix (Powell, 2003) and the action of S_r^{-1} can be approximated using a V-cycle of AMG (Stüben, 2000).

Briefly, AMG is an iterative solver for sparse linear systems that starts from an initial guess and eliminates error by successively combining smoothing (Gauss–Seidel iteration) and coarse grid correction. To focus ideas consider $S^1 \mathbf{u} = \mathbf{b}$ with $\mathbf{u} \in \mathbb{R}^{n_1}$. Before iteration begins the following set-up information is generated.

1. Sequence of grids: $C_l \subset C_{l-1} \subset \dots \subset C_1 = \{1, \dots, n_1\}$ with $|C_k| = n_k$ for $k = 2, \dots, l$.
2. Prolongation matrices: $P_k^{k-1} \in \mathbb{R}^{n_{k-1} \times n_k}$, for $k = 2, \dots, l$.
3. Coarse grid matrices: $S^k = R_{k-1}^k S^{k-1} P_k^{k-1}$, $R_{k-1}^k = (P_k^{k-1})^T$ for $k = 2, \dots, l$.

‘Grids’ are index sets of unknowns. The coarse grids and prolongation matrices are designed so that error not eliminated by smoothing lies in the range of the resulting interpolation operator. Although full convergence proofs are lacking, it is well established that AMG works best for M -matrices. For such matrices it is easy to write down algebraic equations that characterize the error that cannot be eliminated by smoothing. The concept of strong dependence is then key to designing C_k and P_k^{k-1} . Smoothed error can be shown to vary slowly in the direction of strong dependence and for successful interpolation, unknowns that strongly influence the i th unknown at level k should be represented in the coarse grid C_{k+1} .

DEFINITION 3.2 (Strong influence/dependence). The j th unknown strongly influences the i th unknown if for a given $\alpha > 0$, $|[S]_{ij}| \geq \alpha \max_{k \neq i} |[S]_{ik}|$. Then we say that i strongly depends on j .

Strong dependence for the i th unknown is gauged by the relative size of the matrix entries in the i th row. Here we have a sequence of matrices S_r , with diagonal and off-diagonal entries

$$[S_r]_{ii} = \sum_j [K_r]_{l_j, l_j}^{-1}, \quad [S_r]_{ik} = -[K_r]_{l_i, l_k}^{-1}, \quad i \neq k. \quad (3.5)$$

In (3.5) $\{l_j\}$ are the labels of the edges of the i th element and l is the label of the edge connecting the i th and k th elements. Thus, the entries of S_r depend on a_M^r ; strong connections for the i th element are determined relative to the maximum value of a_M^r on the patch of elements sharing an edge with it. If we apply AMG as a black box to solve systems with each S_r separately then the strong connections vary from matrix to matrix, and the grids C_k and interpolation matrices P_k^{k-1} produced in the set-up phase are different for each one.

DEFINITION 3.3 (Optimal preconditioner). An optimal preconditioner for $S \in \mathbb{R}^{n \times n}$ is a matrix P for which the action of P^{-1} can be computed in $\mathcal{O}(n)$ work and the eigenvalues of $P^{-1}S$ are contained in the interval $[\theta, \Theta]$ with constants $\theta, \Theta > 0$ independent of h .

FIG. 2. Multigrid constants θ_r for Example 3.4 appearing in the spectral bounds (3.4) for the finely-tuned AMG preconditioner.

For M -matrices, one V-cycle (iteration) of AMG is known to be an effective preconditioner (e.g. see Ruge & Stüben, 1987). If P is the matrix for which $P^{-1}\mathbf{v}$ denotes the application of one AMG V-cycle to $S\mathbf{u} = \mathbf{v}$, with set-up information (coarse grids, prolongation matrices and coarse grid matrices) generated from S then P is expected to be optimal in the sense of Definition 3.3. Returning to (3.3), choose $D_r := \text{diag}(K_r)$ and set V_r to be the matrix for which $V_r^{-1}\mathbf{v}$ denotes the application of one AMG V-cycle to $S_r\mathbf{u} = \mathbf{v}$ with set-up information generated using S_r . We call this *finely-tuned* AMG preconditioning. Since the M -matrix property is satisfied we expect θ_r and Θ_r in (3.4) to be independent of h and also quite insensitive to $a_M^r(\mathbf{x})$.

EXAMPLE 3.4 Consider (1.1) on $D = (-1, 1) \times (-1, 1)$ with $f = 0$ and boundary conditions $p = 1$ on $\{-1\} \times [-1, 1]$, $p = 0$ on $\{1\} \times [-1, 1]$ and $\mathbf{n} \cdot \mathbf{v} = 0$ on $(-1, 1) \times \{-1, 1\}$. Suppose a_M is given by (1.5), with Gaussian random variables ξ_k , $\mu = 1$, and covariance function (1.3) with $\gamma = 1$ and $\sigma^2 = 5$. For the spatial discretization consider a uniform mesh of 16^2 triangles. Now for $M = 40$ and sparse grid level $l = 3$ we have $n_c = 3361$ linear systems. The multigrid constants θ_r from (3.4) are plotted in Fig. 2 for each system. The average value is 0.677 and the range is $[0.631, 0.681]$. For each r we also have $\Theta_r = 1$, $\alpha_{1,r} \geq 1/2$ and $\alpha_{2,r} \leq 3/2$, so the spectral bounds are indeed insensitive to the choice of collocation point.

If the samples $a_M^r(\mathbf{x})$ are highly oscillatory then we expect some degradation in the performance of the preconditioner as $\theta_r \rightarrow 0$ (Stüben, 2000). The regularity of the samples depends on the regularity of the specified covariance function. Samples of a random field with covariance function (1.3), for example, are spatially rough as $C(\mathbf{x}, \hat{\mathbf{x}})$ is not differentiable at $\mathbf{x} = \hat{\mathbf{x}}$. However, samples of the truncated KL expansion $a_M^r(\mathbf{x})$ are smoother. For fixed M , $a_M^r(\mathbf{x})$ becomes more oscillatory for decreasing γ and for fixed γ becomes more oscillatory for increasing M . If $\gamma \rightarrow 0$ then we require $M \rightarrow \infty$ in (1.4) and (1.5). It is well known, however, that MCMs have a more favourable convergence rate when M is large. Finely-tuned AMG preconditioning works well within the regime of statistical parameters for which SCMs are recommended.

3.2 Mean-based AMG preconditioning

The major disadvantage of finely-tuned AMG preconditioning is that set-up information has to be generated for n_c distinct matrices. Since the matrices are similar there is lots of scope for computational savings. A simple idea is to use one generic preconditioner for all systems. To that end, consider so-called *mean-based* AMG preconditioning. Let K_μ be the mean mass matrix associated with $\mathbf{y}_r = \mathbf{0}$, set

$D_\mu := \text{diag}(K_\mu)$ and let V_μ be the matrix for which $V_\mu^{-1}\mathbf{v}$ denotes the application of one AMG V-cycle to $S_\mu \mathbf{u} = \mathbf{v}$, with set-up information generated using $S_\mu = BD_\mu^{-1}B^T$. The preconditioner

$$P_\mu = \begin{pmatrix} D_\mu & 0 \\ 0 & V_\mu \end{pmatrix} \quad (3.6)$$

is a finely-tuned preconditioner only for the saddle point matrix A_μ . Its efficiency, for the distinct saddle point matrices A_r , is summarized in Lemma 3.5.

LEMMA 3.5 Denote by $\alpha_{1,\mu,r}$ and $\alpha_{2,\mu,r}$ the minimum and maximum eigenvalues of $D_\mu^{-1}K_r$, respectively, and by θ_μ and Θ_μ the minimum and maximum eigenvalues of $V_\mu^{-1}S_\mu$. The eigenvalues of $P_\mu^{-1}A_r$ lie in

$$\begin{aligned} & \left[\frac{1}{2} \left(\alpha_{1,\mu,r} - \sqrt{\alpha_{1,\mu,r}^2 + 4\Theta_\mu} \right), \frac{1}{2} \left(\alpha_{2,\mu,r} - \sqrt{\alpha_{2,\mu,r}^2 + 4\Theta_\mu} \right) \right] \\ & \cup \left[\alpha_{1,\mu,r}, \frac{1}{2} \left(\alpha_{2,\mu,r} + \sqrt{\alpha_{2,\mu,r}^2 + 4\Theta_\mu} \right) \right]. \end{aligned} \quad (3.7)$$

Furthermore, the eigenvalues of $D_\mu^{-1}K_r$ are contained in the interval

$$\left[\alpha_{1,\mu} \left(\frac{\mu}{\mu + \sigma M_r} \right), \alpha_{2,\mu} \left(\frac{\mu}{\mu + \sigma m_r} \right) \right] \quad (3.8)$$

if a_M is given by (1.4), and in

$$[\alpha_{1,\mu} e^{-\sigma M_r}, \alpha_{2,\mu} e^{-\sigma m_r}] \quad (3.9)$$

if a_M is given by (1.5), where $\alpha_{1,\mu}$ and $\alpha_{2,\mu}$ are the minimum and maximum eigenvalues of $D_\mu^{-1}K_\mu$, which are independent of h , and m_r and M_r are defined in (3.3).

Proof. For the case (1.4) the proof follows in a very similar way to Ernst *et al.* (2009, Lemma 4.9) but with a_M replaced by a_M^{-1} in the mass matrices K_r . The bounds for (1.5) are similarly proved. $\square$

Since S_μ is an M -matrix we may assume that θ_μ and Θ_μ do not depend on h and so neither do the spectral inclusion bounds for $P_\mu^{-1}A_r$ in (3.7). Because of (3.8) and (3.9), however, the bounds do depend on σ and $\|\mathbf{y}_r\|_\infty$ (via m_r and M_r), which depend on d if unbounded random variables are used. Note that if $\mathbf{y}_r = \mathbf{0}$ then $m_r = M_r = 0$ and we recover the bound (3.4). If a_M is given by (1.4) we can only ensure (3.2) is satisfied if σ is small and $\|\mathbf{y}_r\|_\infty$ is bounded for each r (i.e. the random variables ξ_k are bounded). Mean-based preconditioning is generally successful in that case. However, for large σ and unbounded random variables, which is permitted for (1.5), we see that a much stronger preconditioner is required.

3.3 AMG preconditioning with recycled set-up

Ideally, we want a preconditioner that is strong, i.e. has spectral inclusion bounds similar to those of the finely-tuned preconditioner P_r but which has set-up costs similar to P_μ . We suggest that some AMG set-up information can be computed once (using *one* representative matrix S) and recycled for all systems. The mean matrix $S_\mu = BD_\mu^{-1}B^T$, associated with the collocation point $\mathbf{y}_r = \mathbf{0}$, is a natural candidate. Let $V_{\mu,r}$ be the matrix for which $V_{\mu,r}^{-1}\mathbf{v}$ denotes the application of one AMG V-cycle to

$S_r \mathbf{u} = \mathbf{v}$ with coarse grids C_k and prolongation matrices P_k^{k-1} generated using S_μ , and consider now, the preconditioner

$$P_{\mu,r} = \begin{pmatrix} D_r & 0 \\ 0 & V_{\mu,r} \end{pmatrix}, \quad r = 1, 2, \dots, n_c.$$

When applying multigrid to the r th system, the coarse grid matrices $S_r^k = R_{k-1}^k S_r^{k-1} P_k^{k-1}$ are computed using S_r and not S_μ , and so the preconditioner is actually distinct for each r . This means that full AMG set-up is performed only once with S_μ . We refer to this strategy as AMG preconditioning with *recycled set-up*.

COROLLARY 3.6 Denote by $\alpha_{1,r}$ and $\alpha_{2,r}$ the minimum and maximum eigenvalues of $D_r^{-1} K_r$, respectively, and by $\theta_{\mu,r}$ and $\Theta_{\mu,r}$ the minimum and maximum eigenvalues of $V_{\mu,r}^{-1} S_r$. The eigenvalues of $P_{\mu,r}^{-1} A_r$ lie in

$$\begin{aligned} & \left[\frac{1}{2} \left(\alpha_{1,r} - \sqrt{\alpha_{1,r}^2 + 4\Theta_{\mu,r}} \right), \frac{1}{2} \left(\alpha_{2,r} - \sqrt{\alpha_{2,r}^2 + 4\Theta_{\mu,r}} \right) \right] \\ & \cup \left[\alpha_{1,r}, \frac{1}{2} \left(\alpha_{2,r} + \sqrt{\alpha_{2,r}^2 + 4\Theta_{\mu,r}} \right) \right]. \end{aligned} \quad (3.10)$$

The spectral inclusion bounds (3.10) for this new preconditioner are the same as the ones obtained for the finely-tuned preconditioner in (3.4) if $\theta_{\mu,r} = \theta_r$ and $\Theta_{\mu,r} = \Theta_r$. To achieve this we require that strongly-influencing points for S_μ are strongly-influencing points for S_r for each r .

EXAMPLE 3.7 Let us reconsider Example 3.4. The multigrid constants $\theta_{\mu,r}$ from (3.10) are plotted in Fig. 3. The average value is 0.663 and the range is $[0.624, 0.683]$. We also have $\Theta_r = 1$ for each $\mathbf{y}_r$. Comparing these multigrid constants with those in Example 3.4 suggests that for this example, strongly-influencing points for S_μ are strongly-influencing points for S_r . Note that although some eigenfunctions associated with (1.3) oscillate more heavily in one spatial direction than the other, these are damped by small eigenvalues, so that the reuse of coarse grids is successful across the whole range of linear systems.

FIG. 3. Multigrid constants $\theta_{\mu,r}$ for Example 3.7 appearing in the spectral bounds (3.10) for AMG preconditioning with recycled set-up.

Recall that the only difference between S_μ and S_r is due to the coefficient samples $a_M(\mathbf{x}, \mathbf{0})$ and $a_M(\mathbf{x}, \mathbf{y}_r)$. The off-diagonal entries of S_μ are given by $[S_\mu]_{ik} = -[K_\mu]_{l,l}^{-1}$, where K_μ is the mean mass matrix and l is the edge connecting elements i and k . Comparing with $[S_r]_{ik}$ in (3.5), and noting that for a finite element partition $\mathfrak{T}_h$ of D with elements $T_i, T_k \in \mathfrak{T}_h$,

$$[K_\mu]_{ll} = \int_{T_i} \mu(\mathbf{x})^{-1} \boldsymbol{\varphi}_l(\mathbf{x}) \cdot \boldsymbol{\varphi}_l(\mathbf{x}) \, d\mathbf{x} + \int_{T_k} \mu(\mathbf{x})^{-1} \boldsymbol{\varphi}_l(\mathbf{x}) \cdot \boldsymbol{\varphi}_l(\mathbf{x}) \, d\mathbf{x},$$

$$[K_r]_{ll} = \int_{T_i} a_M^r(\mathbf{x})^{-1} \boldsymbol{\varphi}_l(\mathbf{x}) \cdot \boldsymbol{\varphi}_l(\mathbf{x}) \, d\mathbf{x} + \int_{T_k} a_M^r(\mathbf{x})^{-1} \boldsymbol{\varphi}_l(\mathbf{x}) \cdot \boldsymbol{\varphi}_l(\mathbf{x}) \, d\mathbf{x},$$

strongly-influencing points for S_μ will only be strongly-influencing points for S_r if, for the same constant c ,

$$\mu(\mathbf{x})|_{T_i} \approx ca_M^r(\mathbf{x})|_{T_i}, \quad \mu(\mathbf{x})|_{T_k} \approx ca_M^r(\mathbf{x})|_{T_k}, \quad (3.11)$$

in the case of (1.4), or

$$\exp(\mu(\mathbf{x})|_{T_i}) \approx ca_M^r(\mathbf{x})|_{T_i}, \quad \exp(\mu(\mathbf{x})|_{T_k}) \approx ca_M^r(\mathbf{x})|_{T_k}, \quad (3.12)$$

in the case of (1.5). Since $\mu(\mathbf{x})$ and $\exp(\mu(\mathbf{x}))$ are (1.4) and (1.5) sampled at $\mathbf{0} = (0, \dots, 0)$ and $a_M^r(\mathbf{x})$ is (1.4) or (1.5) sampled at $\mathbf{y}_r$, for h fixed, (3.11)–(3.12) will be harder to satisfy for increasing σ^2 , decreasing γ and for collocation points $\mathbf{y}_r$ further away from $\mathbf{0}$. The following example demonstrates the effects of these parameters on the quality of AMG preconditioning with recycled set-up.

EXAMPLE 3.8 Consider (1.1) with D, μ, f , covariance function and boundary conditions defined as in Example 3.4 and nonlinear data (1.5). In Table 2 we list the extremal eigenvalues of $V_{\mu,r}^{-1}S_r$ (recycled set-up) and $V_r^{-1}S_r$ (finely tuned), which appear in the bounds (3.4) and (3.10), for particular choices of σ^2, γ and $\mathbf{y}_r$. In addition, we list the extremal eigenvalues of $V_\mu^{-1}S_r$, to show why it is important not to recycle the coarse grid matrices and rather, to construct them using S_r . For a fixed γ we ensure 90% of the total variance of $\log(a(\mathbf{x}, \omega))$ is captured by (1.5). Thus, when $\gamma = 1$ we require $M = 40$ terms. We choose $h = 1/8$ (small enough to capture the eigenfunctions $c_k(\mathbf{x})$) and consider the particular collocation points

$$\mathbf{y}_1 = (-\sqrt{3}, \sqrt{3}, -\sqrt{3}, \sqrt{3}, 0, \dots, 0), \quad \mathbf{y}_2 = (-\sqrt{3}, \sqrt{3}, -\sqrt{3}, \sqrt{3}, \dots, -\sqrt{3}, \sqrt{3}),$$

TABLE 2 Extremal eigenvalues of $V_\mu^{-1}S_r$ (denoted $\vartheta_{\mu,1}$ and $\vartheta_{\mu,2}$) and of $V_{\mu,r}^{-1}S_r$ ($\theta_{\mu,r}$ and $\Theta_{\mu,r}$) and $V_r^{-1}S_r$ (θ_r and Θ_r)

σ^2	γ	$\mathbf{y}_r$	$[\vartheta_{\mu,1}, \vartheta_{\mu,2}]$	$[\theta_{\mu,r}, \Theta_{\mu,r}]$	$[\theta_r, \Theta_r]$
1	1	$\mathbf{y}_1$	$[5.07 \times 10^{-2}, 1.59]$	$[0.666, 1.000]$	$[0.677, 1.000]$
1	1	$\mathbf{y}_2$	$[1.20 \times 10^{-2}, 5.54]$	$[0.625, 1.000]$	$[0.665, 1.000]$
1	0.4	$\mathbf{y}_3$	$[2.07 \times 10^{-2}, 11.6]$	$[0.636, 1.000]$	$[0.676, 1.000]$
1	0.4	$\mathbf{y}_4$	$[6.13 \times 10^{-3}, 30.9]$	$[0.409, 1.000]$	$[0.592, 1.000]$
5	1	$\mathbf{y}_1$	$[1.90 \times 10^{-3}, 2.83]$	$[0.652, 1.000]$	$[0.677, 1.000]$
5	1	$\mathbf{y}_2$	$[6.81 \times 10^{-5}, 46.5]$	$[0.358, 1.000]$	$[0.634, 1.000]$
5	0.4	$\mathbf{y}_3$	$[2.58 \times 10^{-4}, 242]$	$[0.416, 1.000]$	$[0.635, 1.000]$
5	0.4	$\mathbf{y}_4$	$[2.79 \times 10^{-5}, 2340]$	$[0.060, 1.000]$	$[0.294, 1.000]$

FIG. 4. First three grids generated when AMG is applied to S_μ (left) and first three grids generated when AMG is applied to S_4 (right) when $\sigma^2 = 5$, $\gamma = 0.4$, $M = 217$, $\mathbf{y}_4 = (-\sqrt{3}, \sqrt{3}, -\sqrt{3}, \sqrt{3}, \dots, -\sqrt{3})$ and $h = 1/16$ for Example 3.8.

which have 40 components. For $\gamma = 0.4$ we need $M = 217$. We choose $h = 1/16$ and consider

$$\mathbf{y}_3 = \left(-\sqrt{3}, \sqrt{3}, \dots, -\sqrt{3}, 0, \dots, 0\right), \quad \mathbf{y}_4 = \left(-\sqrt{3}, \sqrt{3}, -\sqrt{3}, \sqrt{3}, \dots, -\sqrt{3}\right),$$

where $\mathbf{y}_3$ has 21 nonzero entries out of a total of 217. The values $\pm\sqrt{3}$ can arise as both a CC point (for uniform random variables with mean zero and unit variance) and as a Gauss point in the case of the Gaussian density. All four points, $\mathbf{y}_1, \dots, \mathbf{y}_4$, are sampled for a full tensor grid with three points in each dimension ($d = 2$). For a sparse grid, $\mathbf{y}_1$ is a collocation point when $l = 5$, $\mathbf{y}_2$ when $l = 41$, $\mathbf{y}_3$ when $l = 22$ and $\mathbf{y}_4$ only when $l = 218$. Note also that $\mathbf{y}_2$ is further away from $\mathbf{0}$ than $\mathbf{y}_1$, and $\mathbf{y}_4$ is further away than $\mathbf{y}_3$.

In Table 2 observe that there is little difference between the multigrid constants $\theta_{\mu,r}$ and θ_r and no difference between $\Theta_{\mu,r}$ and Θ_r when σ^2 , γ and $\mathbf{y}_r$ are varied. As expected, the biggest difference occurs when σ^2 is large, γ is small and $\mathbf{y}_r$ is furthest away from $\mathbf{0}$. Consider the case $\sigma^2 = 5$, $\gamma = 0.4$ and $\mathbf{y}_4$. Figure 4 shows the grid sequence generated when we apply AMG to S_μ (which is recycled) and the finely-tuned grid sequence generated when full AMG set-up is performed on S_4 . The grids on the left show that a fairly uniform coarsening takes place for S_μ , which is expected for constant μ . The grids on the right display less uniform coarsening as AMG takes into account the variation in $a_{217}^4(\mathbf{x})$ (see Fig. 5, right plot). $a_{40}^1(\mathbf{x})$ (the log of which is shown in the left plot in Fig. 5) satisfies (3.12) well. Note that we have included $\mathbf{y}_4$ here as a worst-case scenario for the solver. SCMs cannot be recommended for problems with $M = 217$ random variables, and a sparse grid method would never be implemented with $l = 218$.

We now investigate the performance of the suggested preconditioning schemes in detail.

3.4 Numerical results

Consider (1.1) with D , μ , γ , f , covariance function and boundary conditions defined as in Example 3.4. Here we consider a_M given by both (1.4) and (1.5) with $M = 6$. Approximations to the mean and variance of the solution are shown in Fig. 6. In the experiments below we solve the collocation systems (1.10) using preconditioned MINRES with initial guess $\mathbf{q}_r^{(0)} = \mathbf{0}$ and the stopping criterion

FIG. 5. Plot of the sample $\log(a_{40}^1(\mathbf{x}))$, $\sigma^2 = 1$, $\gamma = 1$ (left) and $\log(a_{217}^4(\mathbf{x}))$, $\sigma^2 = 5$, $\gamma = 0.4$ (right) for Example 3.8 with $h = 1/32$.

FIG. 6. Top: $\mathbb{E}[p_{hd}]$ (left), $\mathbb{E}([u_{hd}]_1)$ (centre) and $\mathbb{E}([u_{hd}]_2)$ (right) and bottom: $\text{Var}(p_{hd})$ (left), $\text{Var}([u_{hd}]_1)$ (centre) and $\text{Var}([u_{hd}]_2)$ (right) for the model problem Section 3.4 with $a_M(\mathbf{x}, \xi)$ given by (1.4); $\sigma^2 = 0.1$ and $h = 1/32$.

is $\|\mathbf{g} - A_r \mathbf{q}_r^{(k)}\|_{p_r^{-1}} \leq 10^{-6} \|\mathbf{g}\|_{p_r^{-1}}$. Computations are performed in MATLAB on a PC with 4GB of RAM using a MATLAB version of the AMG code (Boyle *et al.*, 2007). AMG is applied as a black box with one pre and post Gauss–Seidel smoothing step per V-cycle.

EXAMPLE 3.9 First, we investigate robustness with respect to h and σ . We use bounded uniform random variables and apply a sparse grid SCM with CC points. Results are presented in Tables 3 and 4. Table 3 shows that both the fine-tuned and recycled set-up strategies are optimal in terms of the number of preconditioned MINRES iterations with respect to variations in h and σ , for both (1.4) and (1.5). By recycling set-up information, however, we obtain a finely-tuned preconditioner for each system at a reduced cost (see Table 4). AMG set-up can of course be executed quicker in programming languages like FORTRAN and so the exact benefits in terms of time depend on the coding environment. However, no systems arise that cannot be solved in an acceptably low number of iterations with the recycled method (see Figs 7 and 8). In addition, there are considerable computational savings over mean-based preconditioning, whose performance, as Lemma 3.5 predicts, deteriorates as σ^2 increases. Note that when $a_M(\mathbf{x}, \boldsymbol{\xi})$ has the form (1.4), $\sigma^2 = 0.27$ is the largest value for which all subproblems are well posed.

When uniform random variables are used, as in Example 3.9, the collocation points lie in the bounded hypercube $\Gamma = [-\sqrt{3}, \sqrt{3}]^M$ (for both full tensor and sparse grid SCMs). Using Gaussian variables, which is permitted when a_M has the form (1.5), is more difficult. For example, if we use d Gauss points in each dimension, the collocation points are contained in $\Gamma = [-C_d, C_d]^M$ with $C_d = \mathcal{O}(\sqrt{d})$. The impact of this in terms of the bounds in (3.7) is an increase in M_r and decrease in m_r for some systems.

TABLE 3 Average MINRES iterations for Example 3.9, sparse grid (CC points) $l = 3$ and $n_c = 85$

Preconditioning strategy	h	Linear problem (1.4)			Nonlinear problem (1.5)		
		$\sigma^2 = 0.1$	0.2	0.27	$\sigma^2 = 1$	5	10
Finely tuned	1/32	27	27	27	27	27	27
	1/128	28	28	28	28	28	28
Mean based	1/32	36	48	84	79	456	1873
	1/128	37	49	87	80	455	1848
Recycled set-up	1/32	26	26	26	26	27	27
	1/128	28	28	28	28	28	28

TABLE 4 Total AMG set-up times in seconds, maximum MINRES iterations and total iteration times in seconds (in parentheses) for Example 3.9 with nonlinear data, $h = 1/128$, $l = 3$ and $n_c = 85$ fixed

Preconditioning strategy	h	Set-up	$\sigma^2 = 1$	$\sigma^2 = 10$
Finely tuned	1/32	122	28 (39)	28 (38)
	1/128	11305	30 (875)	30 (887)
Mean based	1/32	1	150 (108)	8620 (2402)
	1/128	134	150 (2547)	8324 (56397)
Recycled set-up	1/32	2	28 (39)	30 (39)
	1/128	155	28 (904)	28 (909)

FIG. 7. Maximum and minimum value of $a_M(\mathbf{x}, \mathbf{y}_r)^{-1}$ (top) and preconditioned MINRES iteration counts (bottom) for Example 3.9 with $a_M(\mathbf{x}, \boldsymbol{\xi})$ as in (1.4), $h = 1/128$, $\sigma^2 = 0.2$ and $l = 3$.

FIG. 8. Maximum and minimum value of $a_M(\mathbf{x}, \mathbf{y}_r)^{-1}$ (top) and preconditioned MINRES iteration counts (bottom) for Example 3.9 with $a_M(\mathbf{x}, \boldsymbol{\xi})$ as in (1.5), $h = 1/128$, $\sigma^2 = 5$ and $l = 3$.

EXAMPLE 3.10 We now investigate the robustness of the preconditioning schemes with respect to d when unbounded random variables are used with (1.5). This time we apply both sparse grid and full tensor SCMs with appropriate Gauss points. For the same level of approximation full tensor SCMs yield more very badly ill-conditioned systems than the sparse grid SCMs (see Table 5) and so are more challenging from a linear algebra point of view. The results in Table 5 show that (even for a relatively small variance) mean-based AMG preconditioning becomes increasingly inefficient with respect to l and d when using Gaussian random variables. However, AMG with recycled set-up performs, as predicted, like finely-tuned AMG and is almost insensitive to l and d .

We note that mean-based preconditioning was applied in Ernst *et al.* (2009) to solve decoupled stochastic Galerkin saddle point systems arising for stochastically linear coefficients. Our recycled set-up strategy also provides significant computational savings for those problems.

TABLE 5 Average MINRES iterations for (1.5) for Example 3.10 with $h = 1/32$ and $\sigma^2 = 1$

	Sparse grid (Gauss points)					
	Uniform variables			Gaussian variables		
	$l = 3$ $n_c = 97$	4 545	5 2561	$l = 3$ $n_c = 97$	4 545	5 2561
Preconditioning strategy						
Finely tuned	27	27	27	27	27	27
Mean based	66	74	80	81	101	123
Recycled set-up	26	26	27	27	27	27
	Full tensor (Gauss points)					
	Uniform variables			Gaussian variables		
	$d = 2$ $n_c = 729$	3 4096	4 15625	$d = 2$ $n_c = 729$	3 4096	4 15625
Preconditioning strategy						
Finely tuned	27	27	27	27	27	27
Mean based	121	130	135	204	355	570
Recycled set-up	27	27	27	27	27	27

4. Ordering strategies

It is natural to ask whether additional savings can be made by reusing other information. An obvious strategy is to reuse solutions to previous systems as initial guesses in order to reduce initial residual errors. For this to be effective, the systems (1.10) need to be ordered so that successive ones are similar. Recall that K_r differs from K_s only due to the difference between $a_M^r(\mathbf{x})$ and $a_M^s(\mathbf{x})$. Define $E_{r,s} = \sigma \sum_{k=1}^M \sqrt{\lambda_k} c_k(\mathbf{x})(y_k^r - y_k^s)$. Then $a_M^r(\mathbf{x}) - a_M^s(\mathbf{x}) = E_{r,s}$ for (1.4), while for (1.5), $\log(a_M^r(\mathbf{x})) - \log(a_M^s(\mathbf{x})) = E_{r,s}$. Now, when the correlation length γ is comparable to the length of the physical domain (when SCMs are favoured over MCMs), the values λ_k tend to zero rapidly and only the first few terms in the sum are significant. If $y_k^r - y_k^s$ is small for small k (corresponding to the largest eigenvalues λ_k) then $E_{r,s}$ is small. It is then reasonable to expect that the solution to $A_r \mathbf{q}_r = \mathbf{g}$ provides a good initial guess for solving the system $A_s \mathbf{q}_s = \mathbf{g}$.

For full tensor SCMs, it is easier (and cheaper) to sort the points $\mathbf{y}_r$ as the grid is defined by *one* Cartesian product. We have $y_k^r \in Y_k, k = 1, \dots, M$, and can order such points so that successive ones have $M - 1$ components in common and differ only in, say, the l th component. In that case $E_{r,r+1} = \sigma \sqrt{\lambda_l} c_l(\mathbf{x})(y_l^r - y_l^{r+1})$. However, we also need to ensure that the component l , where the change occurs, corresponds to a large KL eigenvalue as infrequently as possible. We refer to our preferred strategy as FTS (full tensor sorting).

Let $Y_i = \{y_1, \dots, y_{d+1}\}$ for $i = 1, \dots, M$. The standard ordering is (4.1). Successive points do not necessarily have $M - 1$ components in common; for example, $\mathbf{y}_{(d+1)^{M-1}}$ and $\mathbf{y}_{(d+1)^{M-1}+1}$ have none. However, successive points only differ in more than one component every $d + 1$ points. We first divide the points into smaller groups by fixing the first i components. For each $i = M - 1, M - 2, \dots, 1$, there are $(d + 1)^i$ groups of size $(d + 1)^{M-i}$. For example, $\{\mathbf{y}_1, \dots, \mathbf{y}_{(d+1)^{M-1}}\}$ is one of $d + 1$ groups of size $(d + 1)^{M-1}$ obtained by fixing the first component and $\{\mathbf{y}_1, \dots, \mathbf{y}_{d+1}\}$ is one of $(d + 1)^{M-1}$ groups of size $d + 1$ obtained by fixing all except the M th component. If we loop through $i = M - 1, M - 2, \dots, 1$, and reverse the order of the points in every *even* numbered group, we obtain the desired ordering.

$$\begin{aligned}
\mathbf{y}_1 &= (y_1 & y_1 & \dots & \dots & \dots & y_1) \\
\mathbf{y}_2 &= (y_1 & \dots & \dots & \dots & y_1 & y_2) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{y}_{d+1} &= (y_1 & \dots & \dots & \dots & y_1 & y_{d+1}) \\
\mathbf{y}_{(d+1)+1} &= (y_1 & \dots & \dots & y_1 & y_2 & y_1) \\
\mathbf{y}_{(d+1)+2} &= (y_1 & \dots & \dots & y_1 & y_2 & y_2) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{y}_{2(d+1)} &= (y_1 & \dots & \dots & y_1 & y_2 & y_{d+1}) \\
\mathbf{y}_{2(d+1)+1} &= (y_1 & \dots & \dots & y_1 & y_3 & y_1) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{y}_{(d+1)^{M-1}} &= (y_1 & y_{d+1} & \dots & \dots & \dots & y_{d+1}) \\
\mathbf{y}_{(d+1)^{M-1}+1} &= (y_2 & y_1 & \dots & \dots & \dots & y_1) \\
\mathbf{y}_{(d+1)^{M-1}+2} &= (y_2 & y_1 & \dots & \dots & y_1 & y_2) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{y}_{2(d+1)^{M-1}} &= (y_2 & y_{d+1} & \dots & \dots & \dots & y_{d+1}) \\
\mathbf{y}_{2(d+1)^{M-1}+1} &= (y_3 & y_1 & \dots & \dots & \dots & y_1) \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{y}_{(d+1)^M} &= (y_{d+1} & y_{d+1} & \dots & \dots & \dots & y_{d+1})
\end{aligned} \tag{4.1}$$

For example, let $M = 3$ and choose $Y_k = \{0, 1\}$, $k = 1, 2, 3$. There are eight collocation points in total. The standard ordering (4.1) is as follows.

$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_4$	$\mathbf{y}_5$	$\mathbf{y}_6$	$\mathbf{y}_7$	$\mathbf{y}_8$
0	0	0	0	1	1	1	1
0	0	1	1	0	0	1	1
0	1	0	1	0	1	0	1

In FTS we first divide the points into four groups of size two ($i = 2$) and flip the second and fourth groups (i.e. $\{\mathbf{y}_3, \mathbf{y}_4\}$ and $\{\mathbf{y}_7, \mathbf{y}_8\}$). Next we divide the points into two groups of size four ($i = 1$) and flip the second group ($\{\mathbf{y}_5, \mathbf{y}_6, \mathbf{y}_7, \mathbf{y}_8\}$) to produce the following ordering.

$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_4$	$\mathbf{y}_5$	$\mathbf{y}_6$	$\mathbf{y}_7$	$\mathbf{y}_8$
0	0	0	0	1	1	1	1
0	0	1	1	1	1	0	0
0	1	1	0	0	1	1	0

For sparse grid SCMs we cannot apply the FTS strategy. Instead, after solving the r th system, one possibility is to identify the closest remaining point $\mathbf{y}_s$ (point with a corresponding linear system still to solve) to $\mathbf{y}_r$ with respect to the measure $\|\mathbf{y}_r - \mathbf{y}_s\|_1^w = \sum_{k=1}^M \sqrt{\lambda_k} |[\mathbf{y}_r]_k - [\mathbf{y}_s]_k|$ and then use the r th solution as an initial guess for solving the r th preconditioned system. We refer to this strategy as SGS (sparse grid sorting). For example, let $M = 2$ and set $\lambda_1 = 1$, $\lambda_2 = 0.1$ and $l = 3$. If we choose the

one-dimensional interpolation points $\{-1, -1/\sqrt{2}, 0, 1/\sqrt{2}, 1\}$ then there are 13 collocation points and the ordering obtained with our SGS strategy is as follows.

y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}
0	0	0	0	0	-1	-1	$-\frac{1}{\sqrt{2}}$	-1	1	1	$\frac{1}{\sqrt{2}}$	1
0	$\frac{1}{\sqrt{2}}$	1	$-\frac{1}{\sqrt{2}}$	-1	-1	0	0	1	1	0	0	-1

The closest pairs of points are in fact y_2 and y_3 , and y_4 and y_5 ; y_9 and y_{10} are furthest apart.

4.1 Numerical results

EXAMPLE 4.1 Consider (1.1) with D, μ, γ, f , covariance function and boundary conditions defined as in Example 3.4. Now, in addition to applying AMG preconditioning with recycled set-up, we implement the ordering strategies described above and reuse previous solutions as initial guesses. The stopping criterion remains the same as for the zero initial guess experiments. Results for varying σ^2 and d are presented in Tables 6 and 7 for linear and nonlinear data, respectively. The greatest savings occur for (1.4) when σ^2 is small and d (or l) is large. Indeed, for small σ , $E_{r,r+1}$ is reduced by a multiplicative factor and increasing d reduces the distance between distinct points. For the nonlinear problem the benefits are not as promising but improve for increasing d . For example, 54% of MINRES iterations are saved with $l = 7$, $\sigma^2 = 0.01$ and some systems can be solved in as few as two iterations. For (1.5), 29% of MINRES iterations are saved when $l = 7$ and $\sigma^2 = 10$. The cost of FTS grows linearly with n_c and sorting requires only 0.01 s (0.005% of the total solver cost) when $d = 2$ and $\sigma^2 = 0.01$ and only 0.11 s (0.003% of the total cost) when $d = 4$. However, the cost of applying SGS grows like $\mathcal{O}(n_c^2)$. For $l = 3$ and $\sigma^2 = 0.01$, sorting requires 0.04 s (0.13% of the cost) but for $l = 7$, 61.5 s are required (1.6% of the cost). Despite the additional costs that SGS incurs, in this experiment they are far outweighed by the reduction in iteration counts.

Finally, we note that Bäck *et al.* (2011) and Elman *et al.* (2011) attempt to compare the computational costs of SGMs and SCMs for model elliptic problems with stochastically linear diffusion coefficients. In those studies, mean-based preconditioning is assumed. The recycling and ordering techniques above can lead to significant savings in the total number of iterations required to solve the entire sequence of decoupled collocation systems and, therefore, paint SCMs in a more favourable light.

TABLE 6 Average and range, i.e. $[min, max]$, of MINRES iterations for Example 4.1 using AMG preconditioning with recycled set-up linear data, $h = 1/32$ and uniform random variables

		$\sigma^2 = 0.01$	$\sigma^2 = 0.01$
Full tensor ($d = 2, n_c = 729$)	No sorting	26.00 [26, 26]	26.13 [26, 28]
	FTS	15.51 [14, 26]	17.99 [16, 26]
Full tensor ($d = 4, n_c = 15625$)	No sorting	26.00 [26, 26]	26.12 [26, 28]
	FTS	13.81 [11, 26]	16.56 [13, 26]
Sparse grid ($l = 3, n_c = 85$)	No sorting	26.00 [26, 26]	26.01 [26, 27]
	SGS	16.12 [11, 26]	18.28 [14, 26]
Sparse grid ($l = 7, n_c = 15121$)	No sorting	26.00 [26, 26]	26.05 [26, 28]
	SGS	12.09 [2, 26]	14.52 [3, 26]

TABLE 7 Average and range, i.e. $[\min, \max]$, of MINRES iterations for Example 4.1 using AMG preconditioning with recycled set-up nonlinear data, $h = 1/32$ and uniform random variables

		$\sigma^2 = 1$	$\sigma^2 = 10$
Full tensor ($d = 2, n_c = 729$)	No sorting	26.70 [26, 28]	27.10 [25, 32]
	FTS	20.48 [18, 26]	22.29 [16, 29]
Full tensor ($d = 4, n_c = 15625$)	No sorting	26.69 [26, 28]	27.07 [25, 32]
	FTS	18.90 [15, 26]	20.56 [11, 29]
Sparse grid ($l = 3, n_c = 85$)	No sorting	26.48 [26, 28]	26.92 [26, 30]
	SGS	20.82 [16, 26]	23.28 [19, 30]
Sparse grid ($l = 7, n_c = 15121$)	No sorting	26.62 [26, 28]	26.99 [25, 32]
	SGS	17.03 [6, 26]	18.90 [7, 34]

5. Conclusions

In conclusion, we have demonstrated that it is feasible to use AMG preconditioning to solve the sequences of linear systems that arise from the numerical solution of mixed formulations of elliptic PDEs with random data via SCMs. Substantial computational savings are achieved over the mean-based or one-preconditioner-fits-all approach for the stochastically nonlinear problem if set-up information is recycled. The scheme is applicable for any sampling method including MCMs and SGMs based on doubly orthogonal polynomials. Costs can be further reduced when previous solutions are used as initial guesses for subsequent systems, provided the systems are suitably ordered. We note that all the techniques described here can be applied readily to primal formulations of elliptic PDEs. Finally, there is potential to further reduce average iteration counts by using Krylov subspace recycling schemes (e.g. see Parks *et al.*, 2006). However, it is not clear that they provide significant savings when strong, cheap preconditioners such as the one we have suggested here are applied. In initial experiments, further reduction of average iteration counts was achieved but the extra computational costs could not always be justified. This will be the subject of future research.

REFERENCES

- BABUŠKA, I., NOBILE, F. & TEMPONE, R. (2007) A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM J. Numer. Anal.*, **45**, 1005–1034.
- BABUŠKA, I., TEMPONE, R. & ZOURARIS, G. E. (2004) Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM J. Numer. Anal.*, **42**, 800–825.
- BARTHELMANN, V., NOVAK, E. & RITTER, K. (2000) High dimensional polynomial interpolation on sparse grids. *Adv. Comput. Math.*, **12**, 273–288.
- BOYLE, J., MIHAJLOVIĆ, M. D. & SCOTT, J. A. (2007) HSL.MI20: an efficient AMG preconditioner. *Technical Report RAL-TR-2007-021*. Didcot, UK: SFTC Rutherford Appleton Laboratory.
- BÄCK, J., NOBILE, F., TAMELLINI, L. & TEMPONE, R. (2011) Stochastic spectral Galerkin and collocation methods for PDEs with random coefficients: a numerical comparison. *Spectral High Order Methods for Partial Differential Equations, of Lecture Notes in Computational Science and Engineering* (J. S. Hesthaven & E. M. Ronquist eds), vol. 76. Berlin: Springer, pp. 43–62.
- CLENSHAW, C. & CURTIS, A. (1960) A method for numerical integration on an automatic computer. *Numer. Math.*, **2**, 197–205.

- DEB, M. K., BABUŠKA, I. M. & ODEN, J. T. (2001) Solution of stochastic partial differential equations using Galerkin finite element techniques. *Comput. Methods Appl. Mech. Eng.*, **190**, 6359–6372.
- ELMAN, H. C., FURNIVAL, D. G. & POWELL, C. E. (2010) H(div) preconditioning for a mixed finite element formulation of the stochastic diffusion problem. *Math. Comp.*, **79**, 733–760.
- ELMAN, H. C., MILLER, C. W., PHIPPS, E. T. & TUMINARO, R. S. (2011) Assessment of collocation and Galerkin approaches to linear diffusion equations with random data. *Int. J. Uncertainty Quantification*, **1**, 19–33.
- ERNST, O. G., POWELL, C. E., SILVESTER, D. J. & ULLMANN E. (2009) Efficient solvers for a linear stochastic Galerkin mixed formulation of diffusion problems with random data. *SIAM J. Sci. Comp.*, **31**, 1424–1447.
- ERNST, O. G. & ULLMANN, E. (2010) Stochastic Galerkin matrices. *SIAM J. Matrix Anal. Appl.*, **31**, 1848–1872.
- FRAUENFELDER, P., SCHWAB, C. & TODOR R. A. (2005) Finite elements for elliptic problems with stochastic coefficients. *Comput. Methods Appl. Mech. Eng.*, **194**, 205–228.
- GANIS, B., KLIE, H., WHEELER, M. F., WILDEY, T., YOTOV, I. & ZHANG D. (2008) Stochastic collocation and mixed finite elements for flow in porous media. *Comput. Methods Appl. Mech. Eng.*, **197**, 3547–3559.
- GERSTNER, T. & GRIEBEL, M. (1998) Numerical integration using sparse grids. *Numer. Algorithms*, **18**, 209–232.
- GHANEM, R. G., & SPANOS, P. (1991) *Stochastic Finite Elements: A Spectral Approach*. New York: Springer.
- JIN, C., CAI, X.-C., LI, & C. (2007) Parallel domain decomposition methods for stochastic elliptic equations. *SIAM J. Sci. Comput.*, **29**, 2096–2114.
- LOÈVE, M. (1978) *Probability Theory*, vol. II, 4th edn. New York: Springer.
- NOBILE, F., TEMPONE, R. & WEBSTER C. G. (2008) A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM J. Numer. Anal.*, **46**, 2309–2345.
- NOVAK, E. & RITTER, K. (1996) High dimensional integration of smooth functions over cubes. *Numer. Math.*, **75**, 79–97.
- PAIGE, C. C. & SAUNDERS, M. A. (1975) Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, **12**, 617–629.
- PARKS, M. L., DE STURLER, E., MACKEY, G., JOHNSON, D. D. & MAITI, S. (2006) Recycling Krylov subspaces for sequences of linear systems. *SIAM J. Sci. Comput.*, **28**, 1651–1674.
- POWELL, C. E. (2003) Optimal preconditioning for mixed finite element formulation of second-order elliptic problems. *Ph.D. Thesis*, University of Manchester.
- POWELL, C. E., & ULLMANN, E. (2010) Preconditioning stochastic Galerkin saddle point problems. *SIAM J. Matrix Anal. Appl.*, **31**, 2813–2840.
- POWELL, C. E., & SILVESTER, D.J. (2004) Optimal preconditioning for Raviart–Thomas mixed formulation of second-order elliptic problems. *SIAM J. Matrix Anal. Appl.*, **25**, 718–738.
- RIESZ, F. & SZ.-NAGY, B. (1990) *Functional Analysis*. New York: Dover.
- RUGE, J. W. & STÜBEN, K. (1987) Algebraic multigrid (AMG). *Multigrid Methods, Frontiers in Applied Mathematics* (S. F. McCormick ed.), vol. 3. Philadelphia, PA: SIAM, pp. 73–130.
- SMOLYAK, S. (1963) Quadrature and interpolation formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk. SSSR*, **4**, 240–243.
- STÜBEN, K. (2000) Algebraic multigrid (AMG): an introduction with applications. *Multigrid* (U. Trottenberg et al. eds). New York: Academic Press, pp. 413–532.
- ULLMANN, E. (2008) Krylov subspace recycling methods in stochastic finite element computations. *Technical Report 2008–2002 (ISSN 1433–9307)*. Germany: University of Freiberg.
- XIU, D. & HESTHAVEN, J. S. (2005) High-order collocation methods for differential equations with random inputs. *SIAM J. Sci. Comput.*, **27**, 1118–1139.