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,

\[ a^{-1}(x, \omega)u(x, \omega) + \nabla p(x, \omega) = 0 \quad \text{in} \quad D \times \Omega, \]

\[ \nabla \cdot u(x, \omega) = f(x) \quad \text{in} \quad D \times \Omega, \]

\[ p(x, \omega) = 0 \quad \text{on} \quad \partial D \times \Omega, \]

which arises in models of potential flow when there is uncertainty in \( a \). Here \( a : D \times \Omega \to \mathbb{R} \) is a random field, \( D \subset \mathbb{R}^2 \) is the bounded physical domain and \( \Omega \) 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 \( \mu(x) = \mathbb{E}[a(x, \omega)] \) and covariance function

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

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

\[ C(x, \hat{x}) = \sigma^2 \exp(-\|x - \hat{x}\|_1 / \gamma). \]
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 \( \zeta_k(\omega) \in L^2(\Omega) \). A common choice is a (truncated) Karhunen–Loève (KL) expansion (see Loève, 1978),

\[
a_M(\mathbf{x}, \xi) = \mu(\mathbf{x}) + \sigma \sum_{k=1}^{M} \sqrt{\lambda_k} c_k(\mathbf{x}) \zeta_k(\omega), \quad \xi := (\zeta_1(\omega), \ldots, \zeta_M(\omega)).
\]

Here \( \mathbb{E}[\zeta_k] = 0, \mathbb{E}[\zeta_k\zeta_l] = \delta_{k,l} \) and \( \{\lambda_k, c_k(\mathbf{x})\}_{k=1}^{M} \) are the leading eigenpairs of the integral operator associated with \( V(\mathbf{x}, \mathbf{x'}) \) in (1.2). We note that \( \lambda_k \to 0 \) as \( k \to \infty \) at a rate that depends on the regularity of \( C(\mathbf{x}, \mathbf{x'}) \) (see Frauenfelder et al., 2005). Speaking generally, the smaller the correlation length \( \gamma \), 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 \( C(\mathbf{x}, \mathbf{x'}) \) of \( b(\mathbf{x}, \omega) \) are available, \( a(\mathbf{x}, \omega) \) may also be approximated via

\[
a_M(\mathbf{x}, \xi) = \exp\left( \mu(\mathbf{x}) + \sigma \sum_{k=1}^{M} \sqrt{\lambda_k} c_k(\mathbf{x}) \zeta_k(\omega) \right).
\]

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 \( \zeta_k(\Omega) = \Gamma_k \subseteq \mathbb{R} \) and denote the probability density function of \( \zeta_k \) by \( \rho_k \). For example, if \( \zeta_k \) is uniformly distributed then \( \Gamma_k = [-\sqrt{3}, \sqrt{3}] \). We assume the \( \zeta_k \) are independent so that the joint density function of \( \xi \) is \( \rho(\mathbf{y}) = \prod_{k=1}^{M} \rho_k(\mathbf{y}_k) \), where \( \mathbf{y}_k = \zeta_k(\omega) \in \Gamma_k \) and \( \mathbf{y} \in \Gamma = \Gamma_1 \times \cdots \times \Gamma_M \subseteq \mathbb{R}^M \).

Recall now that for any measurable function \( f \) of the multivariate random variable \( \xi \),

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

Replacing \( a(\mathbf{x}, \omega) \) by \( a_M(\mathbf{x}, \xi) \) 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

\[
\int_{\Gamma} \rho(\mathbf{y}) \int_{\mathcal{D}} a_M^{-1}(\mathbf{x}, \mathbf{y}) \mathbf{u} \cdot \mathbf{v} \, d\mathbf{x} \, d\mathbf{y} - \int_{\Gamma} \rho(\mathbf{y}) \int_{\mathcal{D}} p \nabla \cdot \mathbf{v} \, d\mathbf{x} \, d\mathbf{y} = 0,
\]

\[
- \int_{\Gamma} \rho(\mathbf{y}) \int_{\mathcal{D}} q \nabla \cdot \mathbf{u} \, d\mathbf{x} \, d\mathbf{y} = - \int_{\Gamma} \rho(\mathbf{y}) \int_{\mathcal{D}} f q \, d\mathbf{x} \, d\mathbf{y}
\]

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

\[
Q = L^2_\rho(\Gamma, L^2(\mathcal{D})) := \left\{ q : \mathcal{D} \times \Gamma \to \mathbb{R} \middle| \int_{\Gamma} \rho(\mathbf{y}) \| q \|_{L^2(\mathcal{D})}^2 \, d\mathbf{y} < \infty \right\},
\]

\[
\mathbf{V} = L^2_\rho(\Gamma, H_{\text{div}}(\mathcal{D})) := \left\{ \mathbf{v} : \mathcal{D} \times \Gamma \to \mathbb{R}^2 \middle| \int_{\Gamma} \rho(\mathbf{y}) \left( \| \mathbf{v} \|_{L^2(\mathcal{D})}^2 + \| \nabla \cdot \mathbf{v} \|_{L^2(\mathcal{D})}^2 \right) \, d\mathbf{y} < \infty \right\}.
\]
If we assume $0 < a_1 \leq a_M(x, y) \leq a_2 < \infty$ 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 $p_h(x, y) \in Q_h = L^2_p(\Gamma, \Psi_h) \subset Q$ and $u_h(x, y) \in V_h = L^2_p(\Gamma, \Upsilon_h) \subset V$ satisfying

$$
\int_{\Gamma} \rho(y) \int_D a_M^{-1}(x, y) u_h \cdot v \, dx \, dy - \int_{\Gamma} \rho(y) \int_D p_h \nabla \cdot v \, dx \, dy = 0,
$$

(1.6)

$$
- \int_{\Gamma} \rho(y) \int_D q \nabla \cdot u_h \, dx \, dy = - \int_{\Gamma} \rho(y) \int_D f q \, dx \, dy
$$

(1.7)

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

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 $E[p_h]$ and $E[u_h]$ by the sample average at randomly chosen points $y_r \in \Gamma$. For each sample $a^r_M(x) = a_M(x, y_r)$, $p_h^r(x) = p_h(x, y_r) \in \Psi_h$ and $u_h^r(x) = u_h(x, y_r) \in \Upsilon_h$ satisfy

$$
\int_D a^r_M(x)^{-1} u_h^r \cdot v \, dx - \int_D p_h^r \nabla \cdot v \, dx = 0 \quad \forall v(x) \in \Upsilon_h,
$$

(1.8)

$$
- \int_D q \nabla \cdot u_h^r \, dx = - \int_D f q \, dx \quad \forall q(x) \in \Psi_h,
$$

(1.9)

leading to a sequence of decoupled deterministic saddle point problems

$$
\begin{pmatrix}
K_r & B^T \\
B & 0
\end{pmatrix}
\begin{pmatrix}
u_r \\
p_r
\end{pmatrix} =
\begin{pmatrix}
0 \\
f_r
\end{pmatrix},
$$

(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 $u_{hd} \in \Upsilon_h \otimes S_d$ with $S_d \subset L^2(\Gamma)$. 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 $a_M(x, \xi)$ is linear in the variables $\xi_k$, 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 $(u_h, p_h)$ 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 $\sigma$ in (1.2) is large relative to $\mu(x)$. 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 $y_1, \ldots, y_{nc}$ in the hypercube $\Gamma$. A global approximation is then obtained by performing Lagrange interpolation. We obtain

$$u_{hd}(x, y) = \sum_{r=1}^{nc} u_h^r(x) L_r(y), \quad p_{hd}(x, y) = \sum_{r=1}^{nc} p_h^r(x) L_r(y),$$

(2.1)

where each $(u_h^r(x), p_h^r(x)) \in \mathcal{Y}_h \times \Psi_h$ satisfies (1.8)–(1.9) at $y_r \in \Gamma$, and $L_r(y)$ is a multivariate Lagrange polynomial. By construction, $u_{hd}(x, y) \in \mathcal{Y}_h \otimes S_d$ and $p_{hd}(x, y) \in \Psi_h \otimes S_d$, where $S_d = \text{span}\{L_1(y), \ldots, L_{nc}(y)\} \subset L^2(\Gamma)$ and $\dim(S_d) = nc$. Unlike standard MCMs, which choose points
y_r ∈ Γ 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 I_k. Possibilities include Clenshaw–Curtis (CC) points (Clenshaw & Curtis, 1960) and Gauss points. If d_k + 1 points are selected on I_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 I_k of size m_i + 1 where for i ∈ N,

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

Writing \( \mathbf{i} = (i_1, \ldots, i_M) \in \mathbb{N}_M \), and given an approximation level \( l \), the sparse grid on \( \Gamma \) is defined via

\[
H(l, M) = \bigcup_{l \leq \| \mathbf{i} \|_1 < l + M} Z_{i_1} \times \cdots \times Z_{i_M}. 
\]

(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 \( u - u_{hd} \) 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).](http://imajna.oxfordjournals.org/)
3. AMG-based preconditioning

Each \((u_h^r(x), p_h^r(x))\) in (2.1) solves (1.8)–(1.9), where \(a_M^r(x)\) is (1.4) or (1.5) sampled at \(y_r\). For the spatial discretization we choose lowest-order Raviart–Thomas elements. Then \(\Psi_h = \text{span}\{\psi_1(x), \ldots, \psi_{m_h}(x)\}\) and \(\Upsilon_h = \text{span}\{\varphi_1(x), \ldots, \varphi_{h}(x)\}\) 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 \(i, j = 1, \ldots, l_h\) and \(k = 1, \ldots, m_h\),

\[
[K_r]_{ij} = \int_D a_M^r(x)^{-1} \varphi_i(x) \cdot \varphi_j(x) \, dx, \quad [B]_{ki} = -\int_D \psi_k(x) \nabla \cdot \varphi_i(x) \, dx. \tag{3.1}
\]

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

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

If the covariance \(C(x, \hat{x}) \in C(\bar{D} \times \bar{D})\) then the eigenfunctions \(c_k(x)\) of the associated integral operator belong to \(C(\bar{D})\) (Riesz & Sz.-Nagy, 1990, pp. 245). Hence, if \(\mu(x) \in C(\bar{D})\) then for \(M\) fixed we can assume that our samples \(a_M^r(x)\) are continuous. To simplify presentation, unless otherwise stated, we assume \(\mu(x) = \mu \in \mathbb{R}\). 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 \(a_M^r(x)\) (e.g. see Powell & Silvester, 2004) and preconditioning is essential. The dependence on \(a_M^r(x)\) comes from \(K_r\) and \(\kappa(K_r) \lesssim a_{2,r}^{-1}\). This varies with the choice of collocation point \(y_r := (y_1^r, \ldots, y_M^r)\) and some matrices are ‘nicer’ than others. Consider \(y_r = 0\), which sits at the centre of the hypercube \(I\) (when each density \(\rho_k\) is symmetric about zero). The associated matrix \(K_r\), which we denote \(K_{\mu}\), is the so-called mean mass matrix and \(\kappa(K_{\mu}) = \mathcal{O}(1)\). For an arbitrary \(y_r\) 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

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

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

\[
P_r = \begin{pmatrix} D_r & 0 \\ 0 & V_r \end{pmatrix}, \quad r = 1, 2, \ldots, n_c. \tag{3.3}
\]
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 $\theta_r$ and $\Theta_r$ the minimum and maximum eigenvalues of $V_r^{-1}S_r$. The eigenvalues of $P_r^{-1}A_r$, for each $r = 1, 2, \ldots, n_c$, lie in the intervals

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

(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 $S^1 u = b$ with $u \in \mathbb{R}^{n_1}$. Before iteration begins the following set-up information is generated.

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

‘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 $\alpha > 0$, $||S_{ij}|| \geq \alpha \max_{k\neq j} ||S_{ik}||$. 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

$$
[S_r]_{ii} = \sum_j [K_r]_{ij,l_j}, \quad [S_r]_{ik} = -[K_r]_{i,l_k}, \quad i \neq k.
$$

(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 $O(n)$ work and the eigenvalues of $P^{-1}S$ are contained in the interval $[\theta, \Theta]$ with constants $\theta, \Theta > 0$ independent of $n$. 


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 $P^{-1}v$ denotes the application of one AMG V-cycle to $Su = v$, 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 $\theta_r$: $\Theta_r = \text{diag}(Kr)$ and set $V_r$ to be the matrix for which $V_r^{-1}v$ denotes the application of one AMG V-cycle to $Sr u = v$ with set-up information generated using $Sr$. We call this finely-tuned AMG preconditioning. Since the $M$-matrix property is satisfied we expect $\theta_r$ and $\Theta_r$ in (3.4) to be independent of $h$ and also quite insensitive to $\alpha_{M}(\mathbf{x})$.

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 $n \cdot v = 0$ on $(-1, 1) \times [-1, 1]$. Suppose $a_M$ is given by (1.5), with Gaussian random variables $\xi_k$, $\mu = 1$, and covariance function (1.3) with $\gamma = 1$ and $\sigma^2 = 5$. For the spatial discretization consider a uniform mesh of 162 triangles. Now for $M = 40$ and sparse grid level $l = 3$ we have $n_c = 3361$ linear systems. The multigrid constants $\theta_r$ 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 $a^r_M(\mathbf{x})$ are highly oscillatory then we expect some degradation in the performance of the preconditioner as $\theta_r \to 0$ (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 $C(x, \hat{x})$ is not differentiable at $x = \hat{x}$. However, samples of the truncated KL expansion $a^r_M(\mathbf{x})$ are smoother. For fixed $M$, $a^r_M(\mathbf{x})$ becomes more oscillatory for decreasing $\gamma$ and for fixed $\gamma$ becomes more oscillatory for increasing $M$. If $\gamma \to 0$ then we require $M \to \infty$ 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 $K_\mu$ be the mean mass matrix associated with $y_r = 0$, set

![Fig. 2. Multigrid constants for Example 3.4 appearing in the spectral bounds (3.4) for the finely-tuned AMG preconditioner.](image)
$D_\mu := \text{diag}(K_\mu)$ and let $V_\mu$ be the matrix for which $V_\mu^{-1}v$ denotes the application of one AMG V-cycle to $S_\mu u = v$, with set-up information generated using $S_\mu = BD_\mu^{-1}B^T$. The preconditioner

$$P_\mu = \begin{pmatrix} D_\mu & 0 \\ 0 & V_\mu \end{pmatrix}$$

is a finely-tuned preconditioner only for the saddle point matrix $A_\mu$. 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 $\theta_\mu$ and $\Theta_\mu$ the minimum and maximum eigenvalues of $V_\mu^{-1}S_\mu$. The eigenvalues of $P_\mu^{-1}A_r$ lie in

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

(3.7)

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

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

if $a_M$ is given by (1.4), and in

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

(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 $S_\mu$ is an $M$-matrix we may assume that $\theta_\mu$ and $\Theta_\mu$ 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 $\sigma$ and $\|y_r\|_\infty$ (via $m_r$ and $M_r$), which depend on $d$ if unbounded random variables are used. Note that if $y_r = 0$ 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 $\sigma$ is small and $\|y_r\|_\infty$ is bounded for each $r$ (i.e. the random variables $\xi_k$ are bounded). Mean-based preconditioning is generally successful in that case. However, for large $\sigma$ 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 $P_\mu$. 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 $y_r = 0$, is a natural candidate. Let $V_{\mu,r}$ be the matrix for which $V_{\mu,r}^{-1}v$ 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 $S_\mu$, and consider now, the preconditioner

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

When applying multigrid to the $r$th system, the coarse grid matrices $S_k^r = R_{k-1}^r S_r^{k-1} P_k^{k-1}$ are computed using $S_r$ and not $S_\mu$, and so the preconditioner is actually distinct for each $r$. This means that full AMG set-up is performed only once with $S_\mu$. 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

$$\left[ \frac{1}{2} \left( \alpha_{1,r} - \sqrt{\alpha_{1,r}^2 + 4\Theta_{\mu,r}} \right), \frac{1}{2} \left( \alpha_{2,r} - \sqrt{\alpha_{2,r}^2 + 4\theta_{\mu,r}} \right) \right] \cup \left[ \alpha_{1,r}, \frac{1}{2} \left( \alpha_{2,r} + \sqrt{\alpha_{2,r}^2 + 4\theta_{\mu,r}} \right) \right].$$  

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 $S_\mu$ 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 $y_r$. Comparing these multigrid constants with those in Example 3.4 suggests that for this example, strongly-influencing points for $S_\mu$ 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.](image-url)
Recall that the only difference between $S_{\mu}$ and $S_r$ is due to the coefficient samples $a_M(x, \theta)$ and $a_M(x, \theta_r)$. The off-diagonal entries of $S_{\mu}$ are given by $[S_{\mu}]_{ik} = -[K_{\mu}]_{ik}$, where $K_{\mu}$ 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 $\mathbb{T}_h$ of $D$ with elements $T_i, T_k \in \mathbb{T}_h$,

$$[K_{\mu}]_{l} = \int_{T_i} \mu(x)^{-1} \phi_l(x) \cdot \phi_l(x) \, dx + \int_{T_k} \mu(x)^{-1} \phi_l(x) \cdot \phi_l(x) \, dx,$$

$$[K_r]_{l} = \int_{T_i} a_M^r(x)^{-1} \phi_l(x) \cdot \phi_l(x) \, dx + \int_{T_k} a_M^r(x)^{-1} \phi_l(x) \cdot \phi_l(x) \, dx,$$

strongly-influencing points for $S_{\mu}$ will only be strongly-influencing points for $S_r$ if, for the same constant $c$,

$$\mu(x)|_{T_i} \approx c a_M^r(x)|_{T_i}, \quad \mu(x)|_{T_k} \approx c a_M^r(x)|_{T_k},$$

(3.11)

in the case of (1.4), or

$$\exp(\mu(x)|_{T_i}) \approx c a_M^r(x)|_{T_i}, \quad \exp(\mu(x)|_{T_k}) \approx c a_M^r(x)|_{T_k},$$

(3.12)

in the case of (1.5). Since $\mu(x)$ and $\exp(\mu(x))$ are (1.4) and (1.5) sampled at $\theta = (0, \ldots, 0)$ and $a_M^r(x)$ is (1.4) or (1.5) sampled at $y_r$, for $h$ fixed, (3.11)–(3.12) will be harder to satisfy for increasing $\sigma^2$, decreasing $\gamma$ and for collocation points $y_r$ further away from $\theta$. 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 $D$, $\mu$, $f$, 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 $V_{r, r}^{-1} S_r$ (finely tuned), which appear in the bounds (3.4) and (3.10), for particular choices of $\sigma^2$, $\gamma$ and $y_r$. In addition, we list the extremal eigenvalues of $V_{\mu, r}^{-1} S_r$, to show why it is important not to recycle the coarse grid matrices and rather, to construct them using $S_r$. For a fixed $\gamma$ we ensure 90% of the total variance of $\log(a(x, \omega))$ 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 $c_k(x)$) and consider the particular collocation points

$$y_1 = \left( -\sqrt{3}, \sqrt{3}, -\sqrt{3}, \sqrt{3}, 0, \ldots, 0 \right), \quad y_2 = \left( -\sqrt{3}, \sqrt{3}, -\sqrt{3}, \sqrt{3}, \ldots, -\sqrt{3}, \sqrt{3} \right).$$

**Table 2** Extremal eigenvalues of $V_{\mu, r}^{-1} S_r$ (denoted $\vartheta_{\mu, 1}$ and $\vartheta_{\mu, 2}$) and of $V_{r, r}^{-1} S_r$ ($\vartheta_{r, 1}$ and $\vartheta_{r, 2}$) and of $V_{\mu, r}^{-1} S_r$ ($\vartheta_{\mu, r}$ and $\vartheta_{\mu, r}$) and $V_{r, r}^{-1} S_r$ ($\vartheta_{r}$ and $\vartheta_{r}$)

<table>
<thead>
<tr>
<th>$\sigma^2$</th>
<th>$\gamma$</th>
<th>$y_r$</th>
<th>$[\vartheta_{\mu, 1}, \vartheta_{\mu, 2}]$</th>
<th>$[\vartheta_{r, 1}, \vartheta_{r, 2}]$</th>
<th>$[\vartheta_{\mu, r}, \vartheta_{\mu, r}]$</th>
<th>$[\vartheta_{r}, \vartheta_{r}]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>$y_1$</td>
<td>$[5.07 \times 10^{-2}, 1.59]$</td>
<td>$[0.666, 1.000]$</td>
<td>$[0.677, 1.000]$</td>
<td>$[0.666, 1.000]$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>$y_2$</td>
<td>$[1.20 \times 10^{-2}, 5.54]$</td>
<td>$[0.625, 1.000]$</td>
<td>$[0.665, 1.000]$</td>
<td>$[0.665, 1.000]$</td>
</tr>
<tr>
<td>1</td>
<td>0.4</td>
<td>$y_3$</td>
<td>$[2.07 \times 10^{-2}, 11.6]$</td>
<td>$[0.636, 1.000]$</td>
<td>$[0.676, 1.000]$</td>
<td>$[0.676, 1.000]$</td>
</tr>
<tr>
<td>1</td>
<td>0.4</td>
<td>$y_4$</td>
<td>$[6.13 \times 10^{-3}, 30.9]$</td>
<td>$[0.409, 1.000]$</td>
<td>$[0.592, 1.000]$</td>
<td>$[0.592, 1.000]$</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>$y_1$</td>
<td>$[1.90 \times 10^{-3}, 2.83]$</td>
<td>$[0.652, 1.000]$</td>
<td>$[0.677, 1.000]$</td>
<td>$[0.677, 1.000]$</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>$y_2$</td>
<td>$[6.81 \times 10^{-5}, 46.5]$</td>
<td>$[0.358, 1.000]$</td>
<td>$[0.634, 1.000]$</td>
<td>$[0.634, 1.000]$</td>
</tr>
<tr>
<td>5</td>
<td>0.4</td>
<td>$y_3$</td>
<td>$[2.58 \times 10^{-4}, 242]$</td>
<td>$[0.416, 1.000]$</td>
<td>$[0.635, 1.000]$</td>
<td>$[0.635, 1.000]$</td>
</tr>
<tr>
<td>5</td>
<td>0.4</td>
<td>$y_4$</td>
<td>$[2.79 \times 10^{-5}, 2340]$</td>
<td>$[0.060, 1.000]$</td>
<td>$[0.294, 1.000]$</td>
<td>$[0.294, 1.000]$</td>
</tr>
</tbody>
</table>
which have 40 components. For $\gamma = 0.4$ we need $M = 217$. We choose $h = 1/16$ and consider

$$y_3 = \left( -\sqrt{3}, \sqrt{3}, \ldots, -\sqrt{3}, 0, \ldots, 0 \right), \quad y_4 = \left( -\sqrt{3}, \sqrt{3}, -\sqrt{3}, \sqrt{3}, \ldots, -\sqrt{3} \right),$$

where $y_3$ 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, $y_1, \ldots, y_4$, are sampled for a full tensor grid with three points in each dimension ($d = 2$). For a sparse grid, $y_1$ is a collocation point when $l = 5$, $y_2$ when $l = 41$, $y_3$ when $l = 22$ and $y_4$ only when $l = 218$. Note also that $y_2$ is further away from 0 than $y_1$, and $y_4$ is further away than $y_3$.

In Table 2 observe that there is little difference between the multigrid constants $\theta_{\mu, r}$ and $\theta_r$ and no difference between $\Theta_{\mu, r}$ and $\Theta_r$ when $\sigma^2, \gamma$ and $y_r$ are varied. As expected, the biggest difference occurs when $\sigma^2$ is large, $\gamma$ is small and $y_r$ is furthest away from 0. Consider the case $\sigma^2 = 5, \gamma = 0.4$ and $y_4$. Figure 4 shows the grid sequence generated when we apply AMG to $S_\mu$ (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 $S_\mu$, which is expected for constant $\mu$. The grids on the right display less uniform coarsening as AMG takes into account the variation in $a_{40}^4(x)$ (see Fig. 5, right plot). $a_{40}^4(x)$ (the log of which is shown in the left plot in Fig. 5) satisfies (3.12) well. Note that we have included $y_4$ 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 $D, \mu, \gamma, f$, 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 $q_r^{(0)} = 0$ and the stopping criterion
Fig. 5. Plot of the sample \( \log(a_4^{1}(x)) \), \( \sigma^2 = 1, \gamma = 1 \) (left) and \( \log(a_4^{2}(x)) \), \( \sigma^2 = 5, \gamma = 0.4 \) (right) for Example 3.8 with \( h = 1/32 \).

Fig. 6. Top: \( E[p_{hd}] \) (left), \( E[[u_{hd}]]_1 \) (centre) and \( E[[u_{hd}]]_2 \) (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 \( a_M(x, \xi) \) given by (1.4); \( \sigma^2 = 0.1 \) and \( h = 1/32 \).
is \( \| g - A_r q^{(k)} \|_{P_{r-1}} \leq 10^{-6} \| g \|_{P_{r-1}} \). 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 \( \sigma \). 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 \( \sigma \), 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 \( \sigma^2 \) increases. Note that when \( a_M(x, \xi) \) 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 \( C_d = O(\sqrt{d}) \). 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 \( n_c = 85 \)

<table>
<thead>
<tr>
<th>Preconditioning strategy</th>
<th>( h )</th>
<th>( \sigma^2 = 0.1 )</th>
<th>( \sigma^2 = 0.2 )</th>
<th>( \sigma^2 = 0.27 )</th>
<th>( \sigma^2 = 1 )</th>
<th>( \sigma^2 = 5 )</th>
<th>( \sigma^2 = 10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finely tuned</td>
<td>1/32</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
</tr>
<tr>
<td>Mean based</td>
<td>1/32</td>
<td>36</td>
<td>48</td>
<td>84</td>
<td>79</td>
<td>456</td>
<td>1873</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>37</td>
<td>49</td>
<td>87</td>
<td>80</td>
<td>455</td>
<td>1848</td>
</tr>
<tr>
<td>Recycled set-up</td>
<td>1/32</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
<td>28</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>Preconditioning strategy</th>
<th>( h )</th>
<th>Set-up</th>
<th>( \sigma^2 = 1 )</th>
<th>( \sigma^2 = 10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finely tuned</td>
<td>1/32</td>
<td>122</td>
<td>28 (39)</td>
<td>28 (38)</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>11305</td>
<td>30 (875)</td>
<td>30 (887)</td>
</tr>
<tr>
<td>Mean based</td>
<td>1/32</td>
<td>1</td>
<td>150 (108)</td>
<td>8620 (2402)</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>134</td>
<td>150 (2547)</td>
<td>8324 (56397)</td>
</tr>
<tr>
<td>Recycled set-up</td>
<td>1/32</td>
<td>2</td>
<td>28 (39)</td>
<td>30 (39)</td>
</tr>
<tr>
<td></td>
<td>1/128</td>
<td>155</td>
<td>28 (904)</td>
<td>28 (909)</td>
</tr>
</tbody>
</table>
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.
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 $a'_M(x)$ and $a'_M(x)$. Define $E_{r,s} = \sigma \sum_{k=1}^M \sqrt{\lambda_k c_k(x)}(y'_k - y''_k)$. Then $a'_M(x) - a'_M(x) = E_{r,s}$ for (1.4), while for (1.5), $\log(a'_M(x)) - \log(a'_M(x)) = E_{r,s}$. Now, when the correlation length $\gamma$ is comparable to the length of the physical domain (when SCMs are favoured over MCMs), the values $\lambda_k$ tend to zero rapidly and only the first few terms in the sum are significant. If $y'_k - y''_k$ is small for small $k$ (corresponding to the largest eigenvalues $\lambda_k$) then $E_{r,s}$ is small. It is then reasonable to expect that the solution to $A_r q_r = g$ provides a good initial guess for solving the system $A_s q_s = g$.

For full tensor SCMs, it is easier (and cheaper) to sort the points $y_r$ as the grid is defined by one Cartesian product. We have $y'_k \in Y_k, k = 1, \ldots, M$, 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 $E_{r,r+1} = \sigma \sqrt{k_{lcl}}(x)(y'_l - y''_{l+1})$. 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 $Y_i = \{y_1, \ldots, y_{d+1}\}$ for $i = 1, \ldots, M$. The standard ordering is (4.1). Successive points do not necessarily have $M - 1$ components in common; for example, $y_{(d+1)^{M-1}}$ and $y_{(d+1)^{M-1}+1}$ 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 $i = M - 1, M - 2, \ldots, 1$, there are $(d+1)^i$ groups of size $(d+1)^{M-i}$. For example, $\{y_1, \ldots, y_{(d+1)^{M-1}}\}$ is one of $d+1$ groups of size $(d+1)^{M-1}$ obtained by fixing the first component and $\{y_1, \ldots, y_{(d+1)^{M-1}}\}$ 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 $i = M - 1, M - 2, \ldots, 1$, and reverse the order of the points in every even numbered group, we obtain the desired ordering.

<table>
<thead>
<tr>
<th>Table 5</th>
<th>Average MINRES iterations for (1.5) for Example 3.10 with $h = 1/32$ and $\sigma^2 = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preconditioning strategy</td>
<td>Sparse grid (Gauss points)</td>
</tr>
<tr>
<td></td>
<td>Uniform variables</td>
</tr>
<tr>
<td></td>
<td>$n_c = 97$</td>
</tr>
<tr>
<td>Finely tuned</td>
<td>27</td>
</tr>
<tr>
<td>Mean based</td>
<td>66</td>
</tr>
<tr>
<td>Recycled set-up</td>
<td>26</td>
</tr>
</tbody>
</table>

| Preconditioning strategy | Full tensor (Gauss points) | Gaussian variables |
| | Uniform variables | $d = 2$ | 3 | 4 | $d = 2$ | 3 | 4 |
| | $n_c = 729$ | 4096 | 15625 | $n_c = 729$ | 4096 | 15625 |
| 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 |
y_1 = (y_1, y_1 \ldots y_1 y_1)
y_2 = (y_1 \ldots y_1 y_1 y_2)
\vdots \vdots \vdots \vdots \vdots 
y_{d+1} = (y_1 \ldots y_1 y_1 y_{d+1})
y_{(d+1)+1} = (y_1 \ldots y_1 y_2 y_1)
y_{(d+1)+2} = (y_1 \ldots y_1 y_2 y_2)
\vdots \vdots \vdots \vdots \vdots 
y_{2(d+1)} = (y_1 \ldots y_1 y_2 y_{d+1})
y_{2(d+1)+1} = (y_1 \ldots y_1 y_3 y_1)
\vdots \vdots \vdots \vdots \vdots 
y_{(d+1)M-1} = (y_1 y_{d+1} \ldots y_{d+1} y_{d+1})
y_{(d+1)M-1+1} = (y_2 y_1 \ldots y_1 y_{1})
y_{(d+1)M-1+2} = (y_2 y_1 \ldots y_1 y_2)
\vdots \vdots \vdots \vdots \vdots 
y_{2(d+1)M-1} = (y_2 y_{d+1} \ldots y_{d+1} y_{d+1})
y_{2(d+1)M-1+1} = (y_3 y_1 \ldots y_1 y_1)
\vdots \vdots \vdots \vdots \vdots 
y_{(d+1)M} = (y_{d+1} y_{d+1} \ldots y_{d+1} y_{d+1})
(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.

<table>
<thead>
<tr>
<th>y_1</th>
<th>y_2</th>
<th>y_3</th>
<th>y_4</th>
<th>y_5</th>
<th>y_6</th>
<th>y_7</th>
<th>y_8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

In FTS we first divide the points into four groups of size two (i = 2) and flip the second and fourth groups (i.e. \{y_3, y_4\} and \{y_7, y_8\}). Next we divide the points into two groups of size four (i = 1) and flip the second group (\{y_5, y_6, y_7, y_8\}) to produce the following ordering.

<table>
<thead>
<tr>
<th>y_1</th>
<th>y_2</th>
<th>y_3</th>
<th>y_4</th>
<th>y_5</th>
<th>y_6</th>
<th>y_7</th>
<th>y_8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

For sparse grid SCMs we cannot apply the FTS strategy. Instead, after solving the rth system, one possibility is to identify the closest remaining point y_3 (point with a corresponding linear system still to solve) to y_r with respect to the measure \|y_r - y_3\|_i^s = \sum_{k=1}^{M} \sqrt{\lambda_k} |y_{r,k} - y_{s,k}| and then use the rth solution as an initial guess for solving the rth 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.

\[
\begin{array}{ccccccccccc}
\mathbf{y}_1 & \mathbf{y}_2 & \mathbf{y}_3 & \mathbf{y}_4 & \mathbf{y}_5 & \mathbf{y}_6 & \mathbf{y}_7 & \mathbf{y}_8 & \mathbf{y}_9 & \mathbf{y}_{10} & \mathbf{y}_{11} & \mathbf{y}_{12} & \mathbf{y}_{13} \\
0 & 0 & 0 & 0 & -1 & -1 & -1/\sqrt{2} & -1 & 1 & 1 & 1 & 1/\sqrt{2} & 1 \\
0 & \frac{1}{\sqrt{2}} & 1 & -\frac{1}{\sqrt{2}} & -1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & -1 \\
\end{array}
\]

The closest pairs of points are in fact \( \mathbf{y}_2 \) and \( \mathbf{y}_3 \), and \( \mathbf{y}_4 \) and \( \mathbf{y}_5 \); \( \mathbf{y}_9 \) and \( \mathbf{y}_{10} \) are furthest apart.

4.1 Numerical results

Example 4.1 Consider (1.1) with \( D, \mu, \gamma, f \), 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 \( \sigma^2 \) and \( d \) are presented in Tables 6 and 7 for linear and nonlinear data, respectively. The greatest savings occur for (1.4) when \( \sigma^2 \) is small and \( d \) (or \( l \)) is large. Indeed, for small \( \sigma \), \( E_{r,r+1} \) 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 \( O(n_c^3) \). 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

<table>
<thead>
<tr>
<th></th>
<th>( \sigma^2 = 0.01 )</th>
<th>( \sigma^2 = 0.01 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full tensor</td>
<td>No sorting</td>
<td>26.00 [26, 26]</td>
</tr>
<tr>
<td>( d = 2, n_c = 729 )</td>
<td>FTS</td>
<td>15.51 [14, 26]</td>
</tr>
<tr>
<td>Full tensor</td>
<td>No sorting</td>
<td>26.00 [26, 26]</td>
</tr>
<tr>
<td>( d = 4, n_c = 15625 )</td>
<td>FTS</td>
<td>13.81 [11, 26]</td>
</tr>
<tr>
<td>Sparse grid</td>
<td>No sorting</td>
<td>26.00 [26, 26]</td>
</tr>
<tr>
<td>( l = 3, n_c = 85 )</td>
<td>SGS</td>
<td>16.12 [11, 26]</td>
</tr>
<tr>
<td>Sparse grid</td>
<td>No sorting</td>
<td>26.00 [26, 26]</td>
</tr>
<tr>
<td>( l = 7, n_c = 15121 )</td>
<td>SGS</td>
<td>12.09 [2, 26]</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th></th>
<th>$\sigma^2 = 1$</th>
<th>$\sigma^2 = 10$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Full tensor</td>
<td>No sorting</td>
<td>FTS</td>
</tr>
<tr>
<td>$(d = 2, n_c = 729)$</td>
<td>26.70 [26, 28]</td>
<td>27.10 [25, 32]</td>
</tr>
<tr>
<td>Full tensor</td>
<td>No sorting</td>
<td>FTS</td>
</tr>
<tr>
<td>$(d = 4, n_c = 15625)$</td>
<td>20.48 [18, 26]</td>
<td>22.29 [16, 29]</td>
</tr>
<tr>
<td>Sparse grid</td>
<td>No sorting</td>
<td>SGS</td>
</tr>
<tr>
<td>$(l = 3, n_c = 85)$</td>
<td>26.69 [26, 28]</td>
<td>27.07 [25, 32]</td>
</tr>
<tr>
<td>Sparse grid</td>
<td>No sorting</td>
<td>SGS</td>
</tr>
<tr>
<td>$(l = 7, n_c = 15121)$</td>
<td>18.90 [6, 26]</td>
<td>23.28 [19, 30]</td>
</tr>
</tbody>
</table>

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


A. D. GORDON AND C. E. POWELL


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.


79–97.


POWELL, C. E. (2003) Optimal preconditioning for mixed finite element formulation of second-order elliptic prob-


POWELL, C. E., & SILVESTER, D. J. (2004) Optimal preconditioning for Raviart–Thomas mixed formulation of


SMOLYAK, S. (1963) Quadrature and interpolation formulas for tensor products of certain classes of functions.

