SOLVING STOCHASTIC COLLOCATION SYSTEMS WITH ALGEBRAIC MULTIGRID

ANDREW D. GORDON † AND CATHERINE E. POWELL †

Abstract. Stochastic collocation methods facilitate the numerical solution of PDEs with random data and give rise to large sequences of linear systems. For elliptic PDEs, algebraic multigrid (AMG) is a robust solver and considered individually, the systems are trivial to solve. The challenge lies in exploiting the systems' similarities to recycle information and minimize the cost of solving the entire sequence. We propose an efficient solver that is more robust than other solution strategies in the literature. In particular, we show that it is feasible to use a finely-tuned AMG preconditioner for each system if key set-up information is reused. The method is robust with respect to variations in discretization and statistical parameters for stochastically linear and nonlinear data.

1. Introduction. Our starting point is the stochastic steady-state diffusion problem

$$\begin{aligned} -\nabla \cdot (a(\boldsymbol{x},\omega)\nabla p(\boldsymbol{x},\omega)) &= f(\boldsymbol{x}) \qquad && \text{in } D\times\Omega, \quad D\subset\mathbb{R}^2 \\ p(\boldsymbol{x},\omega) &= 0 && \text{on } \partial D\times\Omega, \end{aligned} \tag{1.1}$$

which arises when only limited information about the diffusion coefficient a is available. Here, $a : D\times\Omega\to\mathbb{R}$ is a random field, D is the physical or spatial domain and Ω is a sample space from a probability space $(\Omega,\mathcal{F},P)$ with σ-algebra $\mathcal{F}$ and associated probability measure P. In short, for each $\boldsymbol{x}\in D$, a is a real-valued random variable and fixing $\omega\in\Omega$ corresponds to choosing a realization of a, which is a standard deterministic function in $\boldsymbol{x}\in D$. In (1.1) the input a and the solution p, which is also a random field, are measurable functions on $D\times\Omega$ with respect to $\ell\times P$ where ℓ denotes Lebesgue measure on D. For simplicity, the source term $f\in L^2(D)$ is assumed to be a deterministic function and the boundary condition is deterministic and homogeneous.

We assume that for each $\boldsymbol{x}\in D$, $a\in L^2(\Omega)$. That is, a has finite variance for each $\boldsymbol{x}\in D$ and $a(\boldsymbol{x},\omega)$ is a second-order random field, representing colored noise. Thus, there is an underlying spatial correlation structure and we assume the mean $\mu(\boldsymbol{x})=\mathbb{E}[a(\boldsymbol{x},\omega)]$ and covariance function

$$C(\boldsymbol{x},\boldsymbol{y}) \;=\; \mathbb{E}\big[\big(a(\boldsymbol{x},\omega)-\mu(\boldsymbol{x})\big)\big(a(\boldsymbol{y},\omega)-\mu(\boldsymbol{y})\big)\big] = \sigma^2 V(\boldsymbol{x},\boldsymbol{y}) \tag{1.2}$$

are known. Here, $\sigma^2=C(\boldsymbol{x},\boldsymbol{x})$ is the variance of $a(\boldsymbol{x},\omega)$, which we will assume, for simplicity, is constant in D. The correlation function $V : D\times D\to\mathbb{R}$ in (1.2) typically depends on a parameter $\gamma>0$ called the correlation length. Decreasing the correlation length decreases $V(\boldsymbol{x},\boldsymbol{y})$, the correlation between the random variables $a(\boldsymbol{x},\omega)$ and $a(\boldsymbol{y},\omega)$ and hence the covariance $C(\boldsymbol{x},\boldsymbol{y})$.

We follow the now well-established procedure (e.g. see [6, 2]) of assuming that $a(\boldsymbol{x},\omega)$ is, or 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 [13],

$$a_M(\boldsymbol{x},\boldsymbol{\xi}) = \mu(\boldsymbol{x}) + \sigma\sum_{k=1}^{M}\sqrt{\lambda_k}c_k(\boldsymbol{x})\xi_k(\omega) \tag{1.3}$$

in terms of M uncorrelated random variables ξ_k, with zero mean and unit variance, where $\boldsymbol{\xi}=(\xi_1(\omega),\ldots,\xi_M(\omega))$. In (1.3), $\{\lambda_k,\, c_k(\boldsymbol{x})\}_{k=1}^{M}$ are the leading eigenpairs of the integral operator associated with $V(\boldsymbol{x},\boldsymbol{y})$ in (1.2). We note that $\lambda_k\to 0$ as $k\to\infty$ at a rate that depends on the regularity of $C(\boldsymbol{x},\boldsymbol{y})$ (see [9] for details). Speaking generally, the smaller the correlation length γ, the larger the number of terms M in (1.3) needed to obtain a good approximation $a_M(\boldsymbol{x},\boldsymbol{\xi})$

†School of Mathematics, University of Manchester, Oxford Road, Manchester, M13 9PL, UK. (gordona@cs.man.ac.uk, c.powell@manchester.ac.uk)

to $a(\boldsymbol{x},\omega)$. Alternatively, it is common to assume that $\log(a(\boldsymbol{x},\omega))$ has a given mean $\mu(\boldsymbol{x})$ and covariance function $C(\boldsymbol{x},\boldsymbol{y})$. In this case, a suitable approximation of $a(\boldsymbol{x},\omega)$ is given by

$$a_M(\boldsymbol{x},\boldsymbol{\xi}) = \exp\Big(\mu(\boldsymbol{x}) + \sigma \sum_{k=1}^{M} \sqrt{\lambda_k} c_k(\boldsymbol{x}) \xi_k(\omega)\Big). \tag{1.4}$$

Let $\xi_k(\Omega) = \Gamma_k \subseteq \mathbb{R}$ and denote the probability density function of ξ_k by ρ_k. For example if $\xi_k(\omega)$ is uniformly distributed then $\Gamma_k = [-\sqrt{3}, \sqrt{3}]$ and if $\xi_k(\omega)$ is Gaussian then $\Gamma_k = \mathbb{R}$. If the random variables are independent (as is the case for uncorrelated Gaussian random variables) the joint density function of the multivariate random variable $\boldsymbol{\xi}$ is $\rho(\boldsymbol{\xi}) = \prod_{k=1}^{M} \rho_k(\xi_k)$ where $\boldsymbol{\xi} \in \Gamma = \Gamma_1 \times \cdots \times \Gamma_M \subseteq \mathbb{R}^M$. Replacing $a(\boldsymbol{x},\omega)$ by $a_M(\boldsymbol{x},\boldsymbol{\xi})$ in (1.1) results in an $M+2$ dimensional deterministic boundary-value problem, find $p : D \times \Gamma \to \mathbb{R}$ such that

$$\begin{aligned} -\nabla \cdot (a_M(\boldsymbol{x},\boldsymbol{\xi}) \nabla p(\boldsymbol{x},\boldsymbol{\xi})) &= f(\boldsymbol{x}) \qquad && \text{in } D \times \Gamma, \\ p(\boldsymbol{x},\boldsymbol{\xi}) &= 0 && \text{on } \partial D \times \Gamma, \end{aligned} \tag{1.5}$$

which can now be discretized using standard approximation schemes.

The weak formulation of (1.5)—which has been well studied, (e.g. see [6, 2, 9])—consists in finding $p(\boldsymbol{x},\boldsymbol{\xi}) \in P = L^2_\rho(\Gamma, H^1_0(D))$ and $\mathbb{E}[q(\boldsymbol{\xi})] = \int_\Gamma \rho(\boldsymbol{\xi}) q(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi}$, such that

$$\mathbb{E}\left[\int_D a_M(\boldsymbol{x},\boldsymbol{\xi}) \nabla p(\boldsymbol{x},\boldsymbol{\xi}) \cdot \nabla q(\boldsymbol{x},\boldsymbol{\xi}) \mathrm{d}\boldsymbol{x}\right] = \mathbb{E}\left[\int_D f(\boldsymbol{x}) q(\boldsymbol{x},\boldsymbol{\xi}) \mathrm{d}\boldsymbol{x}\right] \quad \forall\, q \in P. \tag{1.6}$$

If we assume $0 < a_1 \leq a_M(\boldsymbol{x},\boldsymbol{\xi}) \leq a_2 < \infty$, a.e. in $D \times \Gamma$, then a unique solution can be shown to exist, in the standard way, using the Lax-Milgram lemma. So-called stochastic finite element methods proceed by partitioning D in the usual way, leading to the semi-discrete problem: find $p_h \in P_h = L^2_\rho(\Gamma, \Phi_h) \subset P$ with $\Phi_h \subset H^1_0(D)$ and $\dim(\Phi_h) = n_h$ such that

$$\mathbb{E}\left[\int_D a_M(\boldsymbol{x},\boldsymbol{\xi}) \nabla p_h(\boldsymbol{x},\boldsymbol{\xi}) \cdot \nabla q(\boldsymbol{x},\boldsymbol{\xi}) \mathrm{d}\boldsymbol{x}\right] = \mathbb{E}\left[\int_D f(\boldsymbol{x}) q(\boldsymbol{x},\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{x}\right] \quad \forall\, q \in P_h. \tag{1.7}$$

After fixing a spatial discretization, we can tackle (1.6)–(1.7) with Monte Carlo methods (MCMs), stochastic Galerkin methods (SGMs) [11, 6, 2] or stochastic collocation methods (SCMs) [22, 1, 14]. Standard MCMs approximate $\mathbb{E}[p_h]$ by the sample average at randomly chosen points $\boldsymbol{\xi}_r \in \Gamma$. If $a^r_M(\boldsymbol{x}) = a_M(\boldsymbol{x},\boldsymbol{\xi}_r)$ is strictly positive then each $p^r_h(\boldsymbol{x}) = p_h(\boldsymbol{x},\boldsymbol{\xi}_r) \in \Phi_h$ satisfies

$$\int_D a^r_M(\boldsymbol{x}) \nabla p^r_h(\boldsymbol{x}) \cdot \nabla v(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \int_D f(\boldsymbol{x}) v(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \qquad \forall\, v \in \Phi_h, \tag{1.8}$$

leading to a sequence of decoupled, symmetric positive definite linear systems

$$A_r \boldsymbol{p}_r = \boldsymbol{b}, \qquad r = 1, 2, \ldots, \qquad A_r \in \mathbb{R}^{n_h \times n_h}. \tag{1.9}$$

The individual systems in (1.9) can be solved using state of the art deterministic solvers, of which there are many. However, the sample average is slow to converge to the true expected value and alternative discretization schemes on Γ have gained much attention recently.

SGMs seek $p_{hd} \in \Phi_h \otimes S_d$ with $S_d \subset L^2_\rho(\Gamma)$ and have a superior convergence rate to standard MCMs for low values of M (see [2]). However, they yield a single linear system of equations of dimension $n_h \times \dim(S_d)$ that often cannot be decoupled and thus require more sophisticated solvers (e.g. see [7, 17]). If S_d consists of complete polynomials (of total degree d in $\boldsymbol{\xi}$) the Galerkin equations must be solved simultaneously. However, at the expense of the convergence rate, tensor

product polynomials (polynomials of degree d_k in each ξ_k) can also be used. In that case, if $a_M(\boldsymbol{x}, \boldsymbol{\xi})$ is linear in the variables ξ_k, as in (1.3), 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 of dimension $n_h \times n_h$ as in (1.9). SCMs provide a happy medium. Like MCMs, they sample the finite element solution p_h, and so naturally lead to a sequence of decoupled systems (1.9) for both the stochastically linear problem with coefficient (1.3) and the stochastically nonlinear problem with coefficient (1.4). This allows for the re-use of existing deterministic solvers. Unlike MCMs, however, their convergence rate is comparable to that of SGMs.

The conjugate gradient method (CG) is the most efficient iterative solver for the individual systems in (1.9) and algebraic multigrid (AMG) [18, 20] is a widely-used preconditioner for discretized elliptic PDEs that is highly robust with respect to variations in the diffusion coefficients. When the number of systems to be solved in (1.9) is large, however, it may be infeasible to fine-tune an AMG-based preconditioner (or any preconditioner) to the individual matrices. In that case, the one-preconditioner-fits-all approach has merit. That strategy reduces set-up costs but the chosen preconditioner can be so weak for some systems that no savings are made overall.

Jin et al. [12] and Ullmann [21] study the sequences of positive definite systems that arise when (1.6) is discretized with a certain SGM based on doubly-orthogonal polynomials and apply Krylov subspace recycling techniques [16]. However, only stochastically linear diffusion coefficients (1.3) are considered, which always lead to fairly well-conditioned, and highly similar matrices A_r. As we will see, the stochastically nonlinear case (1.4) is far more challenging. Most recycled Krylov subspace solvers are suboptimal (compared to CG) for individual symmetric positive definite systems, but have benefits when applied to a long similar sequence if a weak (but cheap) preconditioner is selected. The domain decomposition preconditioner in [12] is optimal for most systems but weak for a certain subset of them. In [21], one V-cycle of AMG applied to the 'mean' stiffness matrix (with diffusion coefficient $a(\boldsymbol{x}) = \mu(\boldsymbol{x})$) is used to precondition all systems. The efficiency of that preconditioner deteriorates significantly when the standard deviation σ in (1.2) is large relative to the mean $\mu(\boldsymbol{x})$.

Here, we focus on SCMs and investigate the extent to which computational savings can be made by recycling preconditioner information. Our emphasis is on strong, cheap, preconditioners and the re-use of preconditioner information. Specifically, we propose an efficient way to solve the entire sequence (1.9) with CG using AMG-based preconditioners. Unlike previous works that implement SGMs, our method handles (1.3) and (1.4) equally well and is robust with respect to variations in all the discretization and statistical parameters.

2. Stochastic Collocation Methods. SCMs collocate the semi-discrete problem on a set of points $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{n_c}$ in the hypercube Γ. A global approximation is then obtained by performing Lagrange interpolation. For (1.7) we obtain

$$p_{hd}(\boldsymbol{x}, \boldsymbol{\xi}) = \sum_{r=1}^{n_c} p_h^r(\boldsymbol{x}) L_r(\boldsymbol{\xi}), \tag{2.1}$$

where each $p_h^r(\boldsymbol{x}) = p_h(\boldsymbol{x}, \boldsymbol{\xi}_r) \in \Phi_h$ satisfies (1.8) at $\boldsymbol{\xi}_r \in \Gamma$ and $L_r(\boldsymbol{\xi})$ is a multivariate Lagrange polynomial satisfying $L_r(\boldsymbol{\xi}_s) = \delta_{rs}$. By construction, $p_{hd}(\boldsymbol{x}, \boldsymbol{\xi}) \in \Phi_h \otimes S_d$ where $S_d = \text{span}\{L_1(\boldsymbol{\xi}), \ldots, L_{n_c}(\boldsymbol{\xi})\} \subset L^2_\rho(\Gamma)$ and $\dim(S_d) = n_c$.

In the stochastic collocation approach, the error incurred by approximating the finite element solution $p_h(\boldsymbol{x}, \boldsymbol{\xi})$ by $p_{hd}(\boldsymbol{x}, \boldsymbol{\xi})$ is due to interpolation. Unlike standard MCMs, which choose points $\boldsymbol{\xi}_r$ randomly in Γ, good SCMs choose as few points as possible, in a structured way, to minimize the interpolation error. If M is large then this is challenging. Full tensor SCMs (e.g. see [22, 1]) use, as collocation points, the Cartesian product of M sets of interpolation points on the one-dimensional intervals Γ_k. Let Y_k denote the set of interpolation points chosen on Γ_k for $k = 1, \ldots, M$, then the

full tensor collocation grid is simply the set $Y_1 \times \cdots \times Y_M$. Possibilities for Y_k include Clenshaw-Curtis (CC) points [5] (see Figure 2.1) and Gauss points. For the latter, this means the points are roots of the univariate polynomials that are orthogonal with respect to the density function $\rho_k(\xi_k)$. That is, roots of Legendre polynomials for Uniform random variables and roots of Hermite polynomials for Gaussian variables. If $d_k + 1$ points are selected on each interval Γ_k, however, the total number of collocation points is $n_c = \prod_{k=1}^{M}(d_k + 1)$ and this quickly becomes intractable as M increases.

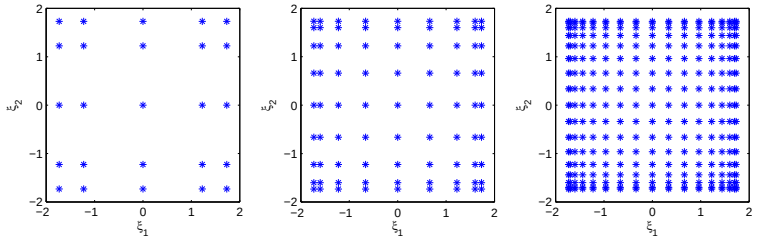

FIG. 2.1. *Full tensor CC collocation points on* $[-\sqrt{3}, \sqrt{3}]^2$ *with* $d_1 = d_2 = 4$, $n_c = 25$ *(left),* $d_1 = d_2 = 8$, $n_c = 81$ *(center),* $d_1 = d_2 = 16$, $n_c = 289$ *(right).*

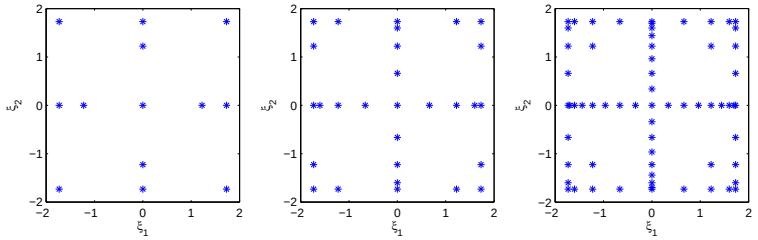

FIG. 2.2. *Sparse grid CC collocation points on* $[-\sqrt{3}, \sqrt{3}]^2$ *with* $l = 3$, $n_c = 13$ *(left),* $l = 4$, $n_c = 29$ *(center),* $l = 5$, $n_c = 65$ *(right).*

Sparse grid SCMs [22, 14] are based on interpolation and cubature rules for high-dimensional problems [15, 10, 3] derived from the work of Smolyak [19]. To define a sparse grid on Γ, 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 $\boldsymbol{i} = (i_1, \ldots, i_M) \in \mathbb{N}^M$, and given an approximation level l, the points are defined via

$$H(l, M) = \bigcup_{l \le \|\mathbf{i}\|_1 < l+M} Z_{i_1} \times \ldots \times Z_{i_M}. \tag{2.2}$$

The convergence rate of multivariate interpolation rules depends on the largest value d for which the rule is exact for all polynomials of *total* degree d. For sparse grid SCMs, we achieve total degree accuracy with $l = d + 1$ [3, Theorem 4] using far fewer points than full tensor SCMs (see Table 2.1). For detailed analysis and discussion of the convergence rates of full tensor SCMs see Babuška et al. [1] and for sparse grid SCMs, see Nobile et al. [14].

We now focus on linear algebra issues associated with computing (2.1).

TABLE 2.1
Dimension of S_d (or n_c) for varying M and d, using Clenshaw-Curtis points

M	5			10			20		
d	1	2	3	1	2	3	1	2	3
Sparse grid SCM	11	61	241	21	221	1581	41	841	11561
Full tensor SCM	32	243	1024	1024	59049	10^6	10^6	10^9	10^{12}

3. Linear systems. Each sample of the finite element solution $p_h^r(\boldsymbol{x})$ in (2.1) solves (1.8) where $a_M^r(\boldsymbol{x})$ is (1.3) or (1.4) sampled at $\boldsymbol{\xi}_r$. If $\Phi_h = \text{span}\{\phi_1(\boldsymbol{x}), \ldots, \phi_{n_h}(\boldsymbol{x})\}$ consists of piecewise polynomials then we have to solve n_c sparse linear systems (1.9) where

$$[A_r]_{ij} = \int_D a_M^r(\boldsymbol{x}) \nabla\phi_i(\boldsymbol{x}) \cdot \nabla\phi_j(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \quad [\boldsymbol{b}]_i = \int_D f(\boldsymbol{x})\phi_i(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \quad i,j = 1,\ldots,n_h. \tag{3.1}$$

We assume for each r that the sample $a_M^r(\boldsymbol{x})$ is strictly positive and bounded, i.e

$$0 < a_{1,r} \le a_M^r(\boldsymbol{x}) \le a_{2,r} < \infty \quad \text{a.e. in } D \tag{3.2}$$

and so we immediately obtain a bound for the condition number of each stiffness matrix A_r. That is, $\kappa(A_r) \lesssim a_{2,r} a_{1,r}^{-1} h^{-2}$, where h denotes the largest edge length in the finite element mesh. Note that $a_M^r(\boldsymbol{x})$ is not strictly positive for (1.3) if unbounded (e.g. Gaussian) random variables are used. If piecewise linear polynomials are used for Φ_h and (3.2) holds, then each A_r is an M-matrix in the sense of Definition 3.1.

DEFINITION 3.1 (M-matrix). *An M-matrix is a symmetric positive definite matrix with positive diagonal entries and non-positive off-diagonal entries.*

Theorem 3.2 gives insight into how ill-conditioned each matrix A_r is with respect to the discretization and statistical parameters. For simplicity, we assume $\mu(\boldsymbol{x}) = \mu > 0$ is spatially constant and that the finite element meshes used to construct Φ_h are shape regular and quasi-uniform.

THEOREM 3.2. *The eigenvalues of the stiffness matrix A_r in* (3.1) *are contained in the interval*

$$\left[ch^2(\mu - T_r), C(\mu + T_r)\right] \qquad \text{or} \qquad \left[ch^2 e^{\mu - T_r}, Ce^{\mu + T_r}\right]$$

if $a_M(\boldsymbol{x}, \boldsymbol{\xi})$ is given by (1.3) *or* (1.4) *respectively, where $c, C > 0$ are independent of h and $a_M^r(\boldsymbol{x})$, and*

$$T_r = \sigma\, S_M\, ||\boldsymbol{\xi}_r||_\infty, \qquad S_M = \sum_{k=1}^{M} \sqrt{\lambda_k}\, ||c_k||_{L^\infty(D)}. \tag{3.3}$$

Proof. Let $\boldsymbol{u} \in \mathbb{R}^{n_h} \setminus \{\boldsymbol{0}\}$ and set $v(\boldsymbol{x}) = \sum_{j=1}^{n_h} u_j \phi_j(\boldsymbol{x}) \in \Phi_h$. Define the matrix A_0 via

$$[A_0]_{ij} = \int_D \nabla\phi_i(\boldsymbol{x}) \cdot \nabla\phi_j(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \qquad i,j = 1,\ldots n_h,$$

and recall the standard result

$$ch^2 \le \frac{\boldsymbol{u}^T A_0 \boldsymbol{u}}{\boldsymbol{u}^T \boldsymbol{u}} \le C \tag{3.4}$$

(e.g. see [8]) for constants c, C independent of h. If $a_M(\boldsymbol{x}, \boldsymbol{\xi})$ is defined as in (1.3),

$$|\boldsymbol{u}^T A_r \boldsymbol{u} - \mu \boldsymbol{u}^T A_0 \boldsymbol{u}| = \left| \int_D \left(\sigma \sum_{k=1}^{M} \sqrt{\lambda_k}\, c_k(\boldsymbol{x})\, [\boldsymbol{\xi}_r]_k \right) \nabla v(\boldsymbol{x})\, \nabla v(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right| \le T_r \boldsymbol{u}^T A_0 \boldsymbol{u}.$$

Hence $(\mu - T_r)\boldsymbol{u}^T A_0 \boldsymbol{u} \leq \boldsymbol{u}^T A_r \boldsymbol{u} \leq (\mu + T_r)\boldsymbol{u}^T A_0 \boldsymbol{u}$. Combining with (3.4) gives the first result. The bound for (1.4) is similarly obtained. □

As $M \to \infty$, S_M (defined in Theorem 3.2) converges at a rate that depends on $C(\boldsymbol{x}, \boldsymbol{y})$ (see [9]). Note that the spectral inclusion bounds in Theorem 3.2 are different for each stiffness matrix A_r since each one corresponds to a different collocation point and T_r depends on

$$||\boldsymbol{\xi}_r||_\infty = \max_{k \in \{1,\ldots,M\}} |[\boldsymbol{\xi}_r]_k|.$$

In addition, T_r depends on σ and $||\boldsymbol{\xi}_r||_\infty$ depends on d if unbounded random variables are used.

3.1. AMG Preconditioning. AMG [18, 20] is a well-known iterative solver for sparse linear systems that starts from an initial guess and successively eliminates error via a combination of smoothing (e.g. Gauss-Seidel iteration) and coarse grid correction. 'Grids' in this context are simply index sets of unknowns and no geometric information is required.

To focus ideas, consider the single linear system $A^1\boldsymbol{u} = \boldsymbol{b}$ with $\boldsymbol{u} \in \mathbb{R}^{n_1}$. Before AMG iteration can begin, there is a set-up phase, during which the following 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: $A^k = R_{k-1}^k A^{k-1} P_k^{k-1}$, $R_{k-1}^k = (P_k^{k-1})^T$ for $k = 2, \ldots, l$.

Coarse grids and prolongation matrices are constructed by exploiting algebraic information in the given matrix A^1, resulting in a finely-tuned preconditioner for that matrix. An algorithm for one step or 'V-cycle' of AMG applied to the linear system $A^1\boldsymbol{u}^1 = \boldsymbol{b}^1$ is given below. Here, A^k and $\boldsymbol{u}^k$ denote a matrix and vector corresponding to the grid C_k.

Smooth $A^1\boldsymbol{u}^1 = \boldsymbol{b}^1$ with initial guess $\boldsymbol{v}^1$ and update $\boldsymbol{v}^1$ with new estimate.
  Restrict the residual: $\boldsymbol{r}^2 = R_1^2(A^1\boldsymbol{v}^1 - \boldsymbol{b}^1)$.
  Smooth $A^2\boldsymbol{e}^2 = \boldsymbol{r}^2$ with initial guess $\boldsymbol{0}$ to obtain $\boldsymbol{v}^2$.
    ⋮
    Restrict the residual: $\boldsymbol{r}^l = R_{l-1}^l(A^{l-1}\boldsymbol{v}^{l-1} - \boldsymbol{b}^{l-1})$.
    Solve $A^l\boldsymbol{v}^l = \boldsymbol{r}^l$ directly.
    ⋮
  Correct $\boldsymbol{v}^2 = \boldsymbol{v}^2 + P_3^2\boldsymbol{v}^3$.
  Smooth $A^2\boldsymbol{e}^2 = \boldsymbol{r}^2$ with initial guess $\boldsymbol{v}^2$ and update $\boldsymbol{v}^2$ with new estimate
Correct $\boldsymbol{v}^1 = \boldsymbol{v}^1 + P_2^1\boldsymbol{v}^2$.
Smooth $A^1\boldsymbol{u}^1 = \boldsymbol{b}^1$ with initial guess $\boldsymbol{v}^1$ and update $\boldsymbol{v}^1$ with final estimate.

For each matrix A_r defined in (3.1) we can employ one V-cycle of AMG as a preconditioner for CG. For M-matrices, this is known to be a good strategy (e.g. see [18]). Specifically, if P_r is the matrix for which $P_r^{-1}\boldsymbol{v}$ denotes the application of one AMG V-cycle to $A_r\boldsymbol{u} = \boldsymbol{v}$, with set-up information (coarse grids, prolongation matrices and coarse grid matrices) generated using A_r, then P_r is expected to be optimal in the sense of Definition 3.3. We refer to this strategy as *finely-tuned* AMG preconditioning.

DEFINITION 3.3 (optimal preconditioner). *An optimal preconditioner for A_r is a matrix P_r, for which the action of P_r^{-1} can be computed in $O(n_h)$ work and the eigenvalues of $P_r^{-1}A_r$ are contained in the interval $[\theta_r, \Theta_r]$ with constants $\theta_r, \Theta_r > 0$ independent of h.*

Rigorous convergence proofs are lacking for AMG but if the M-matrix property is not strongly violated then we expect θ_r and Θ_r to be quite insensitive to h *and* $a_M^r(\boldsymbol{x})$. If the sampled diffusion

coefficient $a_M^r(\boldsymbol{x})$ is oscillatory, which can occur if $C(\boldsymbol{x},\boldsymbol{y})$ has a small correlation length γ, then we expect some degradation [20]. However, MCMs are more appropriate than SCMs in that case since the convergence rate of the latter is less favorable for large M.

The only disadvantage of finely-tuned AMG preconditioning is that set-up information has to be generated for n_c distinct matrices. Depending on the computing environment available, this may be costly. Besides, since the matrices are similar, there should be scope for computational savings. Alternatively, we can employ one generic preconditioner for all systems. To this end, let A_μ be the stiffness matrix with coefficient $a_M(\boldsymbol{x},\mathbf{0})$, where

$$a_M(\boldsymbol{x},\mathbf{0}) = \begin{cases} \mu(\boldsymbol{x}) & \text{for (1.3)}, \\ \exp(\mu(\boldsymbol{x})) & \text{for (1.4)}, \end{cases}$$

and let P_μ be the matrix for which $P_\mu^{-1}\boldsymbol{v}$ denotes the application of one AMG V-cycle to $A_\mu \boldsymbol{u} = \boldsymbol{v}$, with set-up information generated using A_μ. Theorem 3.4 states that the efficiency of P_μ as a preconditioner for A_r, varies from system to system.

THEOREM 3.4. *Let $\mu(\boldsymbol{x}) = \mu > 0$. The eigenvalues of $P_\mu^{-1}A_r$ lie in the interval*

$$[c\,\theta_\mu(1 - T_r\mu^{-1}), C\,\Theta_\mu(1 + T_r\mu^{-1})] \qquad or \qquad [c\,\theta_\mu e^{-T_r}, C\,\Theta_\mu e^{T_r}]$$

if $a_M(\boldsymbol{x},\boldsymbol{\xi})$ is given by (1.3) or (1.4) respectively, where c, $C > 0$ are independent of h and $a_M^r(\boldsymbol{x})$ and the eigenvalues of $P_\mu^{-1}A_\mu$ are contained in $[\theta_\mu, \Theta_\mu]$.

Note that we can expect the multigrid constants θ_μ, Θ_μ in the spectral inclusion bounds in Theorem 3.4 to be independent of h and quite insensitive to $a_M(\boldsymbol{x},\mathbf{0})$. We refer to this second strategy as *mean-based* AMG preconditioning. It was applied in [21] for stochastic Galerkin systems. Using Theorem 3.4 we can see here that it is adequate for (1.3) as $T_r\mu^{-1}$ must be small for the problem to be well-posed. In that case, the bounds in Theorem 3.4 are favorable. For the stochastically nonlinear problem, however, T_r can be arbitrarily large (there are no restrictions on d and σ) and, unfortunately, the bound in Theorem 3.4 deteriorates as T_r increases. A stronger preconditioner is required.

DEFINITION 3.5 (strong influence/dependence). *For an M-matrix A, the j^{th} unknown strongly influences the i^{th} unknown if, for a given threshold $\alpha > 0$, $|A_{ij}| \geq \alpha \max_{k\neq i} |A_{ik}|$. In this case, the i^{th} unknown strongly depends on the j^{th} unknown.*

Relatively speaking, the most expensive part of AMG set-up is the coarse grid selection. Coarse grids are designed to capture error not eliminated by smoothing. For M-matrices, such error is known to vary slowly in the direction of strong dependence, in the sense of Definition 3.5. If $a_M^r(\boldsymbol{x})$ is spatially isotropic, strongly influencing points for the stiffness matrix A_μ are likely to be strongly influencing points for the stiffness matrix A_r, which suggests that coarse grids, and prolongation matrices can be computed once (using *one* representative stiffness matrix) and recycled for all systems. Formally, then, let $P_{\mu,r}$ be the matrix for which $P_{\mu,r}^{-1}\boldsymbol{v}$ denotes the application of one AMG V-cycle to $A_r\boldsymbol{u} = \boldsymbol{v}$, with coarse grids and prolongation matrices generated using A_μ. The coarse grid matrices should be computed using A_r and so the preconditioner is distinct for each system. We refer to this strategy as AMG preconditioning with *recycled setup*. As computing coarse grid matrices is relatively cheap, this strategy has set-up costs similar to mean-based AMG preconditioning. However, if A_r is an M-matrix we expect spectral inclusion bounds for $P_{\mu,r}^{-1}A_r$ to be similar to those of $P_r^{-1}A_r$. That is, the eigenvalues of $P_{\mu,r}^{-1}A_r$ lie in $[\theta_{\mu,r}, \Theta_{\mu,r}]$ with $\theta_{\mu,r} \approx \theta_r$ and $\Theta_{\mu,r} \approx \Theta_r$.

3.2. Numerical Results. We now test the three AMG preconditioning strategies on a model problem. Consider (1.1) on $D = (-1,1)\times(-1,1)$ with $f(\boldsymbol{x}) = 1$ and assume $a(\boldsymbol{x},\omega)$ has mean

$\mu(\boldsymbol{x}) = 1$ and covariance function

$$C(\boldsymbol{x}, \boldsymbol{y}) = \sigma^2 \exp\left(- \frac{||\boldsymbol{x} - \boldsymbol{y}||_1}{\gamma} \right). \tag{3.5}$$

We perform experiments with diffusion coefficients $a_M(\boldsymbol{x}, \boldsymbol{\xi})$ of the form (1.3) and (1.4) with $\gamma = 1$ and $M = 6$ random variables. For the spatial discretization, we use piecewise linear polynomials and the resulting stiffness matrices are M-matrices. An approximation to the mean and variance of the solution is shown in Figure 3.1. Note that first and second order statistics are available via a cheap post-processing of the individual solutions of the linear systems.

In the experiments below, we solve the stochastic collocation systems using preconditioned CG with the zero vector as an initial guess for each individual system. Computations are performed in MATLAB in serial on a laptop PC with 4GB of RAM using a MATLAB version of the AMG code [4]. The multigrid method is applied as a black-box with one pre and post Gauss-Seidel smoothing step per V-cycle. The stopping tolerance for the CG iteration is 10^{-6}.

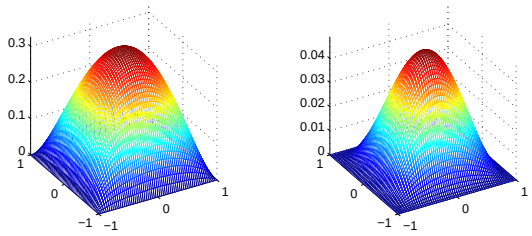

FIG. 3.1. $\mathbb{E}[p_{hd}]$ *(left) and* $Var[p_{hd}]$ *(right) for the model problem with* $a_M(\boldsymbol{x}, \boldsymbol{\xi})$ *given by* (1.4)*; the variance of* a *is* $\sigma^2 = 1$ *and for the spatial discretization,* $h = 1/32$.

Example 3.1. First, we investigate the robustness of the preconditioning schemes with respect to the parameters h and σ. We use bounded Uniform random variables and apply a sparse grid SCM with Clenshaw-Curtis (CC) points. Results are presented in Tables 3.1 and 3.2. Table 3.1 shows that both the fine-tuned and recycled setup strategies are optimal, in terms of the number of preconditioned CG iterations, with respect to variations in h and σ, for both (1.3) and (1.4). By recycling setup information, however, we obtain a finely tuned preconditioner for each system at a fraction of the cost (see Table 3.2). Note, however, that AMG setup can be executed quicker in programming languages like FORTRAN and so the exact benefits in terms of time depend on the coding environment. No systems arise which cannot be solved in an acceptably low number of iterations with the recycled method. In addition, there are considerable computational savings over mean-based preconditioning, whose performance, as Theorem 3.4 predicts, deteriorates as σ^2 increases. Note that, for sparse grid SCMs with $l = 3$, when $a_M(\boldsymbol{x}, \boldsymbol{\xi})$ has the form (1.3) with $\gamma = 1$ and $M = 6$, $\sigma^2 = 0.27$ is the largest value (to two decimal places) for which all subproblems are well-posed.

When Uniform random variables are used, as in Example 3.1, the collocation points always 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 $a_M(\boldsymbol{x}, \boldsymbol{\xi})$ has the form (1.4), is more difficult. For example, for full tensor SCMs using 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 hypercube grows as d increases. The impact of this, in terms of the bounds in Theorems 3.4 and 3.2, is an increase in T_r, for some systems.

Example 3.2. We now investigate the robustness of the preconditioning schemes with respect to d and σ when unbounded random variables are used. This time, we apply full tensor SCMs with

TABLE 3.1
Average CG *iterations for Example 3.1,* $l = 3$ *and* $n_c = 85$.

Preconditioning strategy	h	Linear problem (1.3) $\sigma^2 = 0.1$	0.2	0.27	Nonlinear problem (1.4) $\sigma^2 = 1$	5	10
None	1/32	177	188	195	228	348	500
	1/128	748	799	828	972	1493	2170
Finely-tuned	1/32	6.00	6.00	6.00	6.00	6.13	6.25
	1/128	6.01	6.15	6.32	6.60	6.91	7.01
Mean-based	1/32	8.16	9.38	10.44	13.96	32.64	61.14
	1/128	8.46	9.91	11.01	14.69	34.92	66.61
Recycled setup	1/32	6.00	6.00	6.00	6.02	6.42	6.69
	1/128	6.02	6.14	6.16	6.41	6.87	7.45

TABLE 3.2
Total AMG *setup times in seconds, maximum* CG *iterations and total iteration times in seconds (in parentheses) for Example 3.1 with* $h = 1/128$, $l = 3$ *and* $n_c = 85$ *fixed.*

Preconditioning strategy	Setup	Linear problem (1.3) $\sigma^2 = 0.1$		$\sigma^2 = 0.27$		Nonlinear problem (1.4) $\sigma^2 = 1$		$\sigma^2 = 10$	
Finely-tuned	3212	7	(67)	7	(73)	7	(79)	8	(80)
Mean-based	41	12	(94)	25	(121)	23	(168)	224	(740)
Recycled setup	48	7	(67)	7	(68)	8	(76)	12	(84)

appropriate Gauss points. Sparse grid methods can also be used; they simply result in a subset of the systems solved here (compare Figs 2.1 and 2.2). Our spectral inclusion bounds tell us that the full tensor SCM yields more very badly ill-conditioned systems than the sparse grid SCM, and so, is more challenging from a linear algebra point of view. The results in Table 3.3 reveal how inefficient mean-based AMG preconditioning becomes with increasing d (and σ) using Gaussian random variables. However, AMG with recycled setup performs, as predicted, like finely-tuned AMG and is almost insensitive to d.

TABLE 3.3
Average CG *iterations for* (1.4) *for Example 3.2 with* $h = 1/32$.

Preconditioning strategy	σ^2	Uniform variables $d = 2$ $n_c = 729$	3 4096	4 15625	Gaussian variables $d = 2$ $n_c = 729$	3 4096	4 15625
Finely-tuned	1	6.00	6.01	6.01	6.04	6.12	6.17
	10	6.41	6.40	6.44	6.53	6.61	6.68
Mean-based	1	17.33	17.99	18.27	22.48	29.12	35.97
	10	116.35	130.89	137.89	276.89	738.70	1777.32
Recycled setup	1	6.05	6.07	6.08	6.16	6.30	6.41
	10	7.04	7.10	7.13	7.38	7.65	7.83

4. Conclusions. In conclusion, we have demonstrated that is it feasible to use AMG preconditioning to solve the sequences of linear systems that arise from the numerical solution of elliptic PDEs with random data via SCMs. Substantial computational savings are achieved over the 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. In addition AMG preconditioning with recycled setup can be used to develop preconditioners for the saddle point systems that arise when SCMs are applied to mixed formulations of the model problem. Once again, this strategy leads to significant computational savings compared to mean-based preconditioning.

REFERENCES

[1] Babuška I., Nobile F., Tempone R., *A stochastic collocation method for elliptic partial differential equations with random input data*. SIAM J. Numer. Anal. **45(3)**, 1005–1034 (2007).

[2] Babuška I., Tempone R., Zouraris G.E., *Galerkin finite element approximations of stochastic elliptic partial differential equations*. SIAM J. Numer. Anal. **42(2)**, 800–825 (2004).

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

[4] Boyle J., Mihajlović M.D., Scott J.A., *HSL_MI20: an efficient AMG preconditioner*. Technical Report RAL-TR-2007-021, SFTC Rutherford Appleton Laboratory, Didcot (2007).

[5] Clenshaw C., Curtis A., *A method for the numerical integration on an automatic computer*. Numer. Math. **2**, 197–205 (1960).

[6] Deb M.K., Babuška I.M., Oden, J.T., *Solution of stochastic partial differential equations using Galerkin finite element techniques*. Comput. Methods Appl. Mech. Engrg. **190**, 6359–6372 (2001).

[7] Eiermann M., Ernst O., Ullmann E., *Computational aspects of the stochastic finite element method*. Comput. Vis. Sci. **10**, 3–15 (2007).

[8] Elman H., Silvester D., Wathen A., *Finite Elements and Fast Iterative Solvers*. Oxford University Press, New York (2005).

[9] Frauenfelder P., Schwab C., Todor R.A., *Finite elements for elliptic problems with stochastic coefficients*. Comput. Methods Appl. Mech. Engrg. **194**, 205–228 (2005).

[10] Gerstner T., Griebel M., *Numerical integration using sparse grids*. Numer. Algorithms **18**, 209–232 (1998).

[11] Ghanem R.G., Spanos P., *Stochastic Finite Elements: A Spectral Approach*. Springer-Verlag, New York (1991).

[12] Jin C., Cai X-C., Li C., *Parallel domain decomposition methods for stochastic elliptic equations*. SIAM J. Sci. Comput. **29(5)**, 2096–2114 (2007).

[13] Loève M., *Probability Theory, Vol. II (4th ed.)*. Springer-Verlag, New York (1978).

[14] Nobile F., Tempone R., Webster C. G., *A sparse grid stochastic collocation method for partial differential equations with random input data*. SIAM J. Numer. Anal. **46(5)**, 2309–2345 (2008).

[15] Novak E., Ritter K., *High dimensional integration of smooth functions over cubes*. Numer. Math. **75**, 79–97 (1996).

[16] Parks M.L., de Sturler E., Mackey G., Johnson D.D., Maiti S., *Recycling Krylov subspaces for sequences of linear systems*. SIAM J. Sci. Comput. **28(5)**, 1651–1674 (2006).

[17] Powell C.E., Elman H. *Block-diagonal preconditioning for spectral stochastic finite element systems*. IMA J. Numer. Anal. **29**, 350–375 (2009).

[18] Ruge J.W., Stüben K., *Algebraic multigrid (AMG)*. In: McCormick S. F. (eds.) Multigrid Methods, Frontiers in Applied Mathematics **10**, SIAM, Philadelphia (1986).

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

[20] Stüben K., *Algebraic multigrid (AMG): an introduction with applications*. In: Trottenberg U. et al. (eds.) Multigrid, Academic Press, New York (2000).

[21] Ullmann E., *Krylov subspace recycling methods in stochastic finite element computations*. Technical report 2008–02 (ISSN 1433–9307), University of Freiberg, Germany (2008).

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