Efficient Solution Algorithms for Partial Differential Equations with Random Coefficients

Howard C. Elman
Department of Computer Science
University of Maryland at College Park

Collaborators
Darran Furnival, University of Maryland
Catherine Powell, University of Manchester, England
Christopher Miller, University of Maryland
Eric Phipps, Sandia National Laboratories
Ray Tuminaro, Sandia National Laboratories
Oliver Ernst, University of Freiberg, Germany
Elisabeth Ullmann,University of Bath, England

Problem Statement and Assumptions

Diffusion equation $-\nabla \cdot (a(\mathbf{x}, \omega)\nabla u) = f$ on $\mathcal{D} \subset \mathbb{R}^d$,
with suitable boundary conditions

Initial Assumptions

Coercivity: $0 < \alpha_1 \leq a \leq \alpha_2 < \infty \;\Rightarrow$ well posed

Finite expansion: $a(\mathbf{x}, \boldsymbol{\xi}) = a_0 + \sigma \sum_{r=1}^{m} a_r(\mathbf{x})\boldsymbol{\xi}_r$

Independence: $\{\xi_r = \xi_r(\omega)\}$ uncorrelated with density functions $\rho_r(\xi_r)$,
joint density $\rho(\boldsymbol{\xi}) = \rho_1(\xi_1)\rho_2(\xi_2)\cdots\rho_r(\xi_r)$

Stochastic Galerkin: FEM/FD in space,
Polynomial chaos of total degree p in $\boldsymbol{\xi}$

$\longrightarrow$ Requirement: Solve one large algebraic system $A\mathbf{u} = \mathbf{f}$

$$A = G_0 \otimes A_0 + \sum_{r=1}^{m} G_r \otimes A_r$$

Stochastic collocation: "level-p"

$\longrightarrow$ Solve multiple systems of standard structure

Multigrid Methods for the Stochastic Problems

I. Apply multigrid across spatial component (E. & Furnival)

Solving $A\mathbf{u} = \mathbf{f}$, $\quad A = G_0 \otimes A_0^{(h)} + \sum_{r=1}^{m} G_r \otimes A_r^{(h)}$

$$[A_r]_{jk} = \int_{\mathcal{D}} a_r(x) \nabla\phi_k(x) \cdot \nabla\phi_j(x) dx, \quad [G_r]_{lq} = \int_{\Gamma(\Omega)} \xi_r \psi_q(\xi) \psi_l(\xi) \rho(\xi) d\xi$$

Fine grid operators: $A^{(h)}$, $A_r^{(h)}$ spatial discretization parameter h
Course grid operators: $A^{(2h)}$, $A_r^{(2h)}$ spatial discretization parameter $2h$

One multigrid (two-grid) step:

for $j = 1 : k$
$\quad u^{(h)} \leftarrow (I - Q^{-1}A^{(h)})u^{(h)} + Q^{-1}f^{(h)}$ $\quad$ k smoothing steps
end
$r^{(2h)} = \mathcal{R}(f^{(h)} - A^{(h)}u^{(h)})$ $\quad$ Restriction
Solve $A^{(2h)}c^{(2h)} = r^{(2h)}$ $\quad$ Coarse grid correction $\mathcal{R} = I \otimes R$
$u^{(h)} \leftarrow u^{(h)} + \mathcal{P}c^{(2h)}$ $\quad$ Prolongation $\mathcal{P} = I \otimes P$

Sketch of convergence analysis: Use "standard" approach

$$e^{(i+1)} = [(A^{(h)})^{-1} - \mathcal{P}(A^{(2h)})^{-1}\mathcal{R}]\,[A^{(h)}(I - Q^{-1}A^{(h)})^k]\,e^{(i)}$$

Establish for all y

Approximation property $\left\|[(A^{(h)})^{-1} - \mathcal{P}(A^{(2h)})^{-1}\mathcal{R}]y\right\|_{A^{(h)}} \leq \|y\|_2$

Smoothing property $\left\|A^{(h)}(I - Q^{-1}A^{(h)})^k y\right\|_2 \leq \|y\|_{A^{(h)}}$

For approximation property: Introduce *semi-discrete space* $H_0^1(\mathcal{D}) \otimes \mathcal{T}^{(p)}$

$\mathcal{T}^{(p)}$ = discrete stochastic space

Weak formulation: $a(u^{(p)}, v^{(p)}) = (f, v^{(p)})$ for all $v^{(p)} \in H_0^1(\mathcal{D}) \otimes \mathcal{T}^{(p)}$

Then:
$$\begin{aligned}\left\|[(A^{(h)})^{-1} - \mathcal{P}(A^{(2h)})^{-1}\mathcal{R}]y\right\|_{A^{(h)}} &= \|u^{(hp)} - u^{(2h,p)}\|_a \\ &\leq \|u^{(hp)} - u^{(p)}\|_a + \|u^{(p)} - u^{(2h,p)}\|_a \\ &\leq c\|y\|_{A^{(h)}}\end{aligned}$$

Last step: from standard arguments based on approximability, regularity for *every realization* in the semi-discrete space

Mean-Based Multigrid

II. Apply multigrid to mean as preconditioner

Solving $A\mathbf{u} = \mathbf{f}$

Preconditioner for use with CG (Kruger, Pellisetti, Ghanem):

Mean $Q = G_0 \otimes A_0$

$A_0 \sim \int_{\mathcal{D}} a_0(x) \nabla \phi_k(x) \cdot \nabla \phi_j(x) dx, \quad G_0 = I$

Further refinement (Le Maitre et al.)

Use multigrid to approximate action of Q^{-1}:

$$Q_{MG}^{-1} \equiv I \otimes A_{0,MG}^{-1}$$

Convergence analysis (E. & Powell):

Coefficient:	$a(\mathbf{x},\boldsymbol{\xi}) = a_0 + \sigma \sum_{r=1}^m a_r(\mathbf{x})\xi_r$
Coefficient matrix:	$A = G_0 \otimes A_0 + \sum_{r=1}^m G_r \otimes A_r$
Mean-based preconditioner:	$Q = G_0 \otimes A_0$
Multigrid preconditioner:	$Q_{MG} = G_0 \otimes A_{0,MG}$

Theorem: For $a_0 = \mu$ constant,

$$1 - \tau \le \frac{(w,Aw)}{(w,Qw)} \le 1 + \tau$$

where $$\tau = (\sigma/\mu)\, c(p) \sum_{r=1}^m \sqrt{\lambda_r}\, \|a_r\|_\infty .$$

If in addition the MG approximation satisfies $\beta_1 \le \frac{(w,Qw)}{(w,Q_{MG}w)} \le \beta_2$, then

$$\frac{(w,Aw)}{(w,Q_{MG}w)} = \frac{(w,Aw)}{(w,Qw)}\ \frac{(w,Qw)}{(w,Q_{MG}w)} \le \left(\frac{1+\tau}{1-\tau}\right)\left(\frac{\beta_2}{\beta_1}\right)$$

Comments

- Establishes textbook convergence of multigrid, rate independent of spatial discretization parameter h
- Minimal dependence on stochastic parameter p.
- Applies to any basis for stochastic space
- Second method: simpler but more dependent on # terms m

Alternative: Collocation Methods

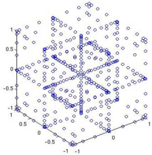
Monte-Carlo (sampling) method: find $u \in H^1_E(\mathcal{D})$ s.t.

$$\int_{\mathcal{D}} a(\mathbf{x}, \boldsymbol{\xi}_k) \nabla u \cdot \nabla v dx \quad \text{for all } v \in H^1_{E_0}(\mathcal{D})$$

for a collection of samples $\{\boldsymbol{\xi}_k\} \in L^2(\Gamma)$

Collocation (Xiu, Hesthaven, Babuška, Nobile, Tempone, Webster)

Choose $\{\boldsymbol{\xi}_k\}$ in a special way (sparse grids), then construct discrete solution $u^{(hp)}(\mathbf{x}, \boldsymbol{\xi}) \in \mathcal{S}^E_h \otimes \mathcal{T}^{(p)}$ to interpolate $\{u_h(\mathbf{x}, \boldsymbol{\xi}_k)\}$



Advantages (vs. stochastic Galerkin):
- decouples algebraic system (like MC)
- applies in a straightforward way to nonlinear random terms

Disadvantage: dimensionality $\sim 2^p \times$ (Galerkin) for comparable accuracy

Experimental Comparison

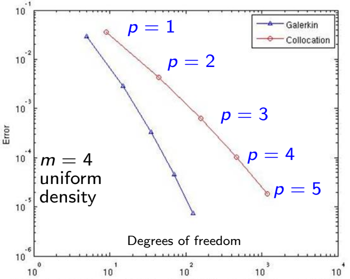
E., Miller, Phipps, Tuminaro

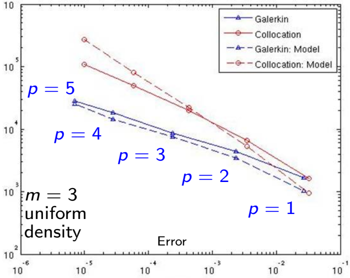
To the right: **Accuracy**

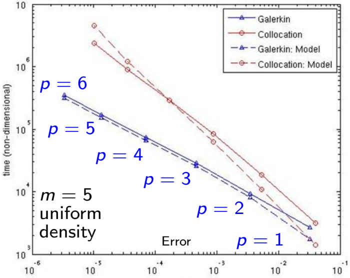
$p =$ polynomial degree for Galerkin
level for collocation
Gaussian abscissas with linear growth

Errors are comparable

Below: **Performance**







Experimental Results: CPU Times

Performed on a serial machine with C code and
CG/AMG code from Trilinos
Truncated Gaussian density

Observation: Galerkin faster, more so as number of stochastic variables (KL terms) grows

	Galerkin			Collocation		
p	$m=5$	$m=10$	$m=12$	$m=5$	$m=10$	$m=12$
1	.058	.147	.263	.069	.163	.218
2	.269	1.20	2.00	.532	2.13	3.17
3	1.20	13.14	24.50	2.41	16.99	29.31
4	3.50	53.79	121.61	8.31	102.60	200.94
5	6.51	117.73		24.56	515.74	

Discussion

Shows Galerkin formulation is tractable

In these circumstances, cheaper than collocation
But: intrusive

Requirements:

- **Linear** dependence on stochastic parameters
- Knowledge of joint density function for parameters

Problem Statement

Diffusion equation $-\nabla \cdot (c(\mathbf{x},\omega)\nabla u) = f$ on $\mathcal{D} \subset \mathbb{R}^d$,
with suitable boundary conditions

Diffusion coefficient $c(\mathbf{x},\omega)$ is a log-normal random field

$$\begin{aligned} k(\mathbf{x},\omega) &= \exp(a(\mathbf{x},\omega)) \\ a(\mathbf{x},\omega) = \log k(\mathbf{x},\omega) &= a_0(\mathbf{x}) + \sigma \textstyle\sum_{m=1}^{\infty} \sqrt{\lambda_m}\, a_m(\mathbf{x})\xi_m(\omega) \\ &\approx \underbrace{a_0(\mathbf{x}) + \sigma \textstyle\sum_{m=1}^{M} \sqrt{\lambda_m}\, a_m(\mathbf{x})\xi_m(\omega)}_{\text{Will use finite-term expression in sequel}^{\dagger}} \end{aligned}$$

Complication: Galerkin much less straightforward when coefficient is nonlinear in $\boldsymbol{\xi}$.

---

† For simplicity, we will also take $\{\xi_m\}$ to have *truncated Gaussian distributions*.

New Approach: Convection-Diffusion Formulation

E., Ernst & Ullmann

Diffusion equation $-\nabla \cdot (e^a \nabla u) = f$

Expand using product rule:

$$e^a \left(-\nabla^2 u - \nabla a \cdot \nabla u\right) = f$$

$\longrightarrow$ **Convection-diffusion problem**

$$-\nabla^2 u + \mathbf{w} \cdot \nabla u = e^{-a} f, \quad \mathbf{w} = -\nabla a$$

N.B. This connection is long known, e.g., Varga et al., 1966
Presented in other direction: existence of velocity potential enables recasting of convection-diffusion equation as diffusion equation

Key point:

$$\mathbf{w} = -(\nabla a_0 + \sigma \textstyle\sum_{m=1}^{M} \sqrt{\lambda_m}\, \nabla a_m(\mathbf{x}) \xi_m) \quad \text{is } \textbf{linear} \text{ in } \boldsymbol{\xi}$$

Matrix structure

Extended weak formulation

$$\int_\Gamma \int_{\mathcal{D}} \nabla u \cdot \nabla v - \int_\Gamma \int_{\mathcal{D}} \nabla a_0 \, v - \sigma \sum_{m=1}^{M} \sqrt{\lambda_m} \int_\Gamma \int_{\mathcal{D}} \nabla a_m \, v = \int_\Gamma \int_{\mathcal{D}} e^{-a} f v$$

Generalized polynomial chaos discretization $\longrightarrow$ coefficient matrix:

$$C = I \otimes (L + N_0) + \sum_{m=1}^{M} G_m \otimes N_m$$

L	discrete diffusion operator
N_0	convection term from mean ∇a_0
N_m	convection terms from terms ∇a_m in expansion of ∇a

Advantage: matrix is sparse

Slight disadvantage: matrix is nonsymmetric

For iterative solution: use preconditioned GMRES

Solution Algorithms for Convection-Diffusion Form

Linear system for stochastic Galerkin $Cu = f$

$$C = I \otimes (L + N_0) + \sum_{m=1}^{M} G_m \otimes N_m$$

Right-oriented preconditioning:

Solve $[CP^{-1}]\hat{u} = f$ using GMRES, $u = P^{-1}\hat{u}$

Options for preconditioning

- Diffusion preconditioner: $P = I \otimes L$, n_ξ decoupled diffusion operators
- Mean-based preconditioner: $P = I \otimes (L + N_0)$, n_ξ decoupled *convection-diffusion* operators (provided ∇a_0 is nonzero)
- Refinement: Action of P^{-1}: approximated using multigrid

Representative Analysis

Diffusion preconditioner $P = I \otimes L$:

Consider (generalized) field of values

$$\mathrm{FOV}(C, P) = \left\{ \frac{(\mathbf{v}, C\mathbf{v})}{(\mathbf{v}, P\mathbf{v})} \,:\, \mathbf{v} \in \mathbb{C}^{n_x n_\xi}, \mathbf{v} \neq \mathbf{0} \right\}.$$

Theorem

For the diffusion preconditioner, FOV(C, P) is contained in the circle

$$\{ z \in \mathbb{C} \,:\, |z - 1| \le 2\, c_{\mathcal{D}}\, c_L \}, \quad c_L = \|\nabla a_0\|_\infty + \sigma \nu_{p+1} \sum_{m=1}^{M} \sqrt{\lambda_m} \|\nabla a_m\|_\infty,$$

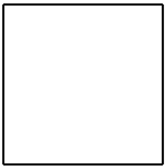
where $\|\nabla a_m\|_\infty = \sup_{\mathbf{x} \in \mathcal{D}} |\nabla a_m(\mathbf{x})|$,

ν_{p+1} = *largest root of orthonormal Rys polynomial of degree* $p + 1$,

$c_{\mathcal{D}} > 0$ *is a constant independent of* h, σ *and* p.

Experimental Results

Benchmark problems



$a = \log k(\mathbf{x}, \boldsymbol{\xi})$ constructed from

$a_0 = 1 + 10x^2$ ($\nabla a_0 \neq 0$)

$\mathrm{Cov}(\mathbf{x}, \mathbf{y}) = \sigma^2 \exp\left(-(\|\mathbf{x} - \mathbf{y}\|_2/\ell)^2\right)$

Problem 1: $\ell = 1$, $M = 5$ term truncated KL exp

Problem 2: $\ell = .5$, $M = 10$-term truncated KL exp

Both: capture 95% of total variance

Iteration Counts

Mean-based $P = I \otimes (L + N_0)$ vs. Diffusion $P = I \otimes L$

Problem 1
$\ell = 1$
$M = 5$

n	σ	p=1	2	3	4	p=1	2	3	4
32	0.1	6	6	6	6	23	25	25	25
64	-	6	6	6	6	23	24	24	24
128	-	5	5	5	5	21	22	22	22
32	1.0	8	9	9	10	26	29	31	32
64	-	7	8	9	9	25	28	30	30
128	-	7	8	8	9	23	26	28	29
32	2.0	10	12	14	15	28	33	36	38
64	-	9	11	13	14	27	31	34	36
128	-	9	11	12	13	25	29	32	34

Problem 2
$\ell = .5$
$M = 10$

n	σ	p=1	2	3	4	p=1	2	3	4
32	0.1	6	6	6	6	24	25	26	25
64	-	6	6	6	6	23	24	25	24
128	-	6	6	5	6	21	23	23	22
32	1.0	10	12	13	14	28	32	34	32
64	-	9	12	13	14	26	30	32	30
128	-	9	11	12	13	25	29	30	29
32	2.0	13	19	24	28	30	37	41	46
64	-	12	18	23	27	29	35	39	44
128	-	12	17	21	26	28	33	37	41

Discussion

- Again: establishes textbook MG convergence wrt spatial mesh
- Overcomes difficulties associated with nonlinearity of lognormal coefficients
- One caveat: for series approximation to ∇a
  Require a to be mean-square continuously differentiable
  (λ_m, a_m): eigenpairs derived from covariance of a, $\mathrm{Cov}(\mathbf{x}, \mathbf{y})$
  OK: $c(\mathbf{x}, \mathbf{y}) = exp\left(-(\|\mathbf{x} - \mathbf{y}\|_2/\ell)^2\right)$
  Not: $c(\mathbf{x}, \mathbf{y}) = exp\left(-\|\mathbf{x} - \mathbf{y}\|_1/\ell\right)$

Restrictions for Ideas Above

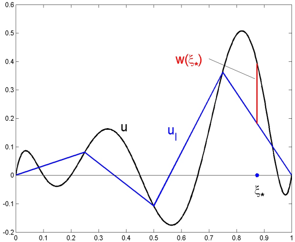
Some restrictions of the approaches just discussed:

- Generally do not have joint pdf for diffusion coefficient
  - Ignored in part I above
  - Worked around in part II (with some limitations)
- Some issues for collocation
  - Simpler than Galerkin (**non-intrusive**), but not cheaper
  - Requires regularity in $\boldsymbol{\xi}$

To address these (E & Miller):

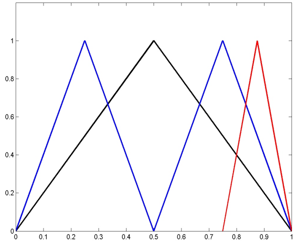
- For pdf: use kernel density estimation methods
- For costs of collocation: use adaptive sparse grid collocation

Adaptive Sparse Grid Collocation



Collocation based on interpolation

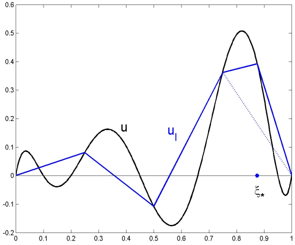
Consider function $u(\xi)$ at left
piecewise linear interpolant $u_I(\xi)$
$\xi_* =$ child, node at next level uniform grid
$w(\xi_*) =$ interpolation error at ξ_*:



Ma & Zabaras: If $u_i(\xi)$ is represented using *hierarchical basis* (left), then
$w(\xi_*) =$ coefficient of basis function in interpolant on refined grid

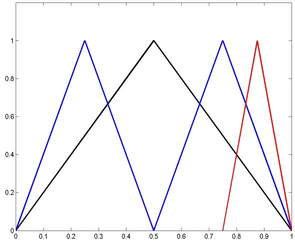
Strategy: Refine grid using child ξ_* iff $|w(\xi_*)| >$ tolerance

Adaptive Sparse Grid Collocation



Collocation based on interpolation

Consider function $u(\xi)$ at left
piecewise linear interpolant $u_I(\xi)$
$\xi_* =$ child, node at next level uniform grid
$w(\xi_*) =$ interpolation error at ξ_*:



Ma & Zabaras: If $u_i(\xi)$ is represented using *hierarchical basis* (left), then
$w(\xi_*) =$ coefficient of basis function
in interpolant on refined grid

Strategy: Refine grid using child ξ_*
iff $|w(\xi_*)| >$ tolerance

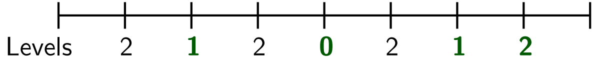
Algorithm: Adaptive interpolation with hierarchical basis functions

Set $\mathcal{A}_0(u)(\xi) = 0$
Set $k = 1$
Set $\Delta\theta^1_{adaptive} = \theta^1$
repeat
  $\Delta\theta^{k+1}_{adaptive} = \emptyset$
  for $\xi_j^{\Delta k} \in \Delta\theta^k_{adaptive}$
    $w_j^k = u(\xi_j^{\Delta k}) - \mathcal{A}_{k-1}(u)(\xi_j^{\Delta k})$
    if $||w_j^k|| > \tau$ **then**
      $\Delta\theta^{k+1}_{adaptive} = \Delta\theta^{k+1}_{adaptive} \cup child(\xi_j^{\Delta k})$
    endif
  endfor
  Set $\mathcal{A}_k(u)(\xi) = \sum_{i=1}^k \sum_j w_j^i \psi_j^i(\xi)$
  $k = k + 1$
until $max(||w_j^{k-1}||) < \tau$

Augment interpolant at refined grid nodes only where interpolation error is too large

Interpolant = sum(levels)
all level-k basis functions

To Use this Idea with Diffusion Equation $-\nabla \cdot (a(\mathbf{x}, \xi) \nabla u) = f$



- Start with $\xi^{(0)}$, compute solution $u(\mathbf{x}, \boldsymbol{\xi}_{(0)})$
  Determines interpolant $[\mathcal{A}^{(0)} u](\mathbf{x}, \xi)$
- Identify children $\{\boldsymbol{\xi}_j^{(1)}\}$ of $\xi^{(0)}$
  Compute solutions $\{u(\mathbf{x}, \boldsymbol{\xi}_j^{(1)})\}$, add $\{\boldsymbol{\xi}_j^{(1)}\}$ to set of collocation points according to adaptive strategy with tolerance test
  $$\|w_j^k\|_\infty \rho(\boldsymbol{\xi}_j^{\Delta k}) > \tau$$
  Determines interpolant $[\mathcal{A}^{(1)} u](\mathbf{x}, \xi)$
- Repeat: identify children of "level-1" points, compute solutions, etc.

Result: Collocation solution $\mathcal{A}^{(k)} u$

Approximate moments, distributions of u using $\mathcal{A}^{(k)} u$

Kernel Density Estimation

Given N samples of ξ, estimate density function by

$$\hat{\rho}(\xi) = \frac{1}{Nh^M} \sum_{k=1}^{N} K\left(\frac{\xi - \xi^{(i)}}{h}\right)$$

For bandwidth h, use *maximum likelihood cross-validation*: maximize

$$CV(h) \equiv \frac{1}{N} \sum_{i=1}^{N} log(\hat{\rho}_{-i}(\xi^{(i)}))$$

where

$$\hat{\rho}_{-i}(\xi) = \frac{1}{Nh^M} \sum_{k=1,k\neq i}^{N} K\left(\frac{\xi - \xi^{(k)}}{h}\right)$$

For $K(\xi)$, use Epanechnikov kernel

$$K(\xi) = \left(\frac{3}{4}\right)^M \prod_{i=1}^{M} (1 - \xi_i^2) \mathbb{1}_{\{-1 \leq \xi_i \leq 1\}}$$

Experimental Results

Test problem:

$$-\frac{d}{dx} a_M(x,\boldsymbol{\xi}) \frac{d}{dx} u(x,\boldsymbol{\xi}) = 1 \quad \forall\ x \in (0,1)$$

$$u(0,\boldsymbol{\xi}) = u(1,\boldsymbol{\xi}) = 0$$

$$a_M = \mu + \sum_{k=0}^{M/2-1} \lambda_k(\xi_{2k} cos(2\pi kx) + \xi_{2k+1} sin(2\pi kx))$$

$$\mu = 3,\ \lambda_k = exp(-k)$$

ξ_k uniformly distributed on $[0,1]$

Experiment: Generate N samples of $\boldsymbol{\xi}$
Use them to generate estimate $\hat{\rho}(\boldsymbol{\xi})$
Use $\hat{\rho}$ to generate adaptive collocation solution $\mathcal{A}u$

Compare with:

Use the same N samples to perform Monte-Carlo simulation

Representative Results: $M = 10$ parameters

N	τ				
	5×10^{-2}	1×10^{-3}	5×10^{-4}	1×10^{-4}	5×10^{-5}
100 9.08×10^{-2}	7.66×10^{-3} **(76)**	8.86×10^{-4} (1026)	4.41×10^{-4} (1655)	4.48×10^{-5} (5026)	8.28×10^{-6} (8111)
500 4.06×10^{-2}	7.13×10^{-3} **(92)**	6.08×10^{-4} (1170)	3.36×10^{-4} (1189)	2.34×10^{-5} (5773)	1.01×10^{-5} (9404)
1000 2.87×10^{-2}	9.19×10^{-3} **(59)**	6.03×10^{-4} (1216)	2.65×10^{-4} (1989)	1.95×10^{-5} (5996)	1.77×10^{-5} (9664)
5000 1.28×10^{-2}	7.16×10^{-3} **(93)**	6.62×10^{-4} **(1120)**	3.03×10^{-4} **(2041)**	2.04×10^{-5} (6095)	1.02×10^{-5} (9787)
20000 6.42×10^{-3}	7.25×10^{-3} **(93)**	6.27×10^{-4} **(1187)**	2.66×10^{-4} **(2127)**	1.96×10^{-5} **(6050)**	5.67×10^{-6} **(9942)**

Monte-Carlo error

Collocation error

$$\left\| \frac{1}{N} \sum_{i=1}^{N} u(x, \boldsymbol{\xi}^{(i)}) - \mathcal{A}(u)(x, \boldsymbol{\xi}^{(i)}) \right\|_{l^2(D)}$$

Parens: Number of DE solves needed for collocation
Green: This number is smaller than N *and* yields smaller error

Concluding Remarks

Various Useful Approaches to Handle Diffusion Problem with Stochastic Coefficient

- Stochastic Galerkin equations with linear dependence on parameters are solvable using multigrid
- Diffusion equation with lognormal coefficent can be handled by transformation to convection-diffusion form
- Adaptive collocation with KDE is general and effective