Reduced-Order Models for Stochastic Partial Differential Equations

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

In honor of Martin Schultz
Yale University, October 2012

Yale Numerical Analysis Group, c. 1981

Partial Differential Equations with Uncertain Coefficients

Examples:

Diffusion equation: $-\nabla \cdot (a(\mathbf{x}, \boldsymbol{\xi}) \nabla u) = f$

Navier-Stokes equations: $-\nabla \cdot (a(\mathbf{x}, \boldsymbol{\xi}) \nabla \vec{u}) + (\vec{u} \cdot \nabla)\vec{u} + \nabla p = \vec{f}$

$\nabla \cdot \vec{u} = 0$

Posed on $\mathcal{D} \subset \mathbb{R}^d$ with suitable boundary conditions

Sources: models of diffusion in media with uncertain permeabilities
multiphase flows

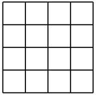
Uncertainty / randomness:

$a = a(\mathbf{x}, \boldsymbol{\xi})$ is a *random field*: for each fixed $x \in \mathcal{D}$, $a(x, \boldsymbol{\xi})$ is a random variable depending on m random parameters $\xi_1, \ldots, \xi_m$

In this study: $a(\mathbf{x}, \boldsymbol{\xi}) = a_0(\mathbf{x}) + \sum_{r=1}^{m} a_r(\mathbf{x})\, \xi_r$

Possible sources:

Karhunen-Loève expansion or Piecewise constant coefficients on $\mathcal{D}$



Monte-Carlo Simulation

Traditional approach:

Sample $a(\mathbf{x}, \xi)$ at all required $\mathbf{x} \in \mathcal{D}$, solve in usual way

Multiple realizations (samples) of $a(\mathbf{x}, \cdot)$ $\longrightarrow$
Multiple realizations of u $\longrightarrow$
Statistical properties of u obtained by averaging

Done using spatial discretization, finite elements or finite differences
$\Longrightarrow$ multiple linear(ized) discrete problems

$$A_{\boldsymbol{\xi}} u_{\boldsymbol{\xi}}^{(h)} = f_h$$

for realization of discrete solution $u_{\boldsymbol{\xi}}^{(h)}$

Problem: convergence is slow, requires many discrete PDE solves

The Stochastic Galerkin Method

Philosophy: Extend finite-element methodology to develop alternative to Monte-Carlo (Ghanem, Spanos, Babuška, Deb, Oden, Matthies, Keese, Karniadakis, Xue, Schwab, Todor)

Standard weak formulation of diffusion problem: find $u \in H^1_E(\mathcal{D})$ s.t.

$$a(u,v) = \ell(v) \quad \text{for all } v \in H^1_{E_0}(\mathcal{D}),$$

where

$$a(u,v) = \int_{\mathcal{D}} a\nabla u \cdot \nabla v dx, \quad \ell(v) = \int_{\mathcal{D}} fv dx$$

Introduce extended (*stochastic*) weak formulation

$$\langle a(u,v)\rangle = \int_{\Omega}\int_{\mathcal{D}} a\nabla u \cdot \nabla v \, dx \, dP(\Omega) = \int_{\boldsymbol{\xi}(\Omega)}\int_{\mathcal{D}} a(\mathbf{x},\boldsymbol{\xi})\,\nabla u \cdot \nabla v \, d\mathbf{x}\, \rho(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$

Bilinear form entails integral over image of random variables $\boldsymbol{\xi}$

Require joint density function associated with $\boldsymbol{\xi}$

$\boldsymbol{\xi}$ plays the role of a Cartesian coordinate

Result:

- From problem in d-dimensional physical space depending on m random parameters, get $(d+m)$-dimensional "continuous" problem
- $d = 2$ or 3, $m = 5, 50, 100, \ldots$

Discretization / Finite dimensional spaces:

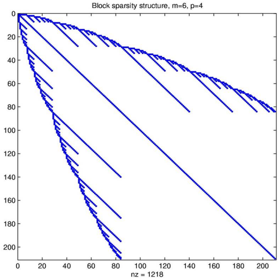
- In physical space: $\mathcal{S}_h \subset H^1_{E_0}(\mathcal{D})$, basis $\{\phi_j\}_{j=1}^N$
  Example: piecewise linear "hat functions"
- In space of random variables: $\mathcal{T}_p \subset L^2(\Gamma)$, basis $\{\psi_\ell\}_{\ell=1}^M$
  Example: m-variate polynomials in $\boldsymbol{\xi}$ of total degree p

Discrete solution:

$$u^{(hp)}(\mathbf{x}, \boldsymbol{\xi}) = \sum_{j=1}^N \sum_{\ell=1}^M u_{j\ell}\phi_j(\mathbf{x})\psi_\ell(\boldsymbol{\xi})$$

Requires solution of large coupled system (right)

Stochastic dimension: $M = \binom{m+p}{p}$



The Stochastic Collocation Method

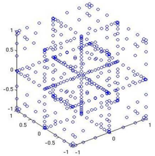
Monte-Carlo (sampling) method: find $u \in H^1_E(\mathcal{D})$ s.t.

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

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 $u^{(hp)}(\mathbf{x}, \boldsymbol{\xi}) \in \mathcal{S}_h^E \otimes \mathcal{T}^{(p)}$ to interpolate $\{u_h(\mathbf{x}, \boldsymbol{\xi}^{(k)})\}$



Structure of collocation solution:

$$u_p^{(hp)}(\mathbf{x}, \boldsymbol{\xi}^{(k)}) := \sum_{\boldsymbol{\xi}^{(k)} \in \Theta_p} u_c(x, \boldsymbol{\xi}^{(k)}) L_{\boldsymbol{\xi}^{(k)}}(\boldsymbol{\xi})$$

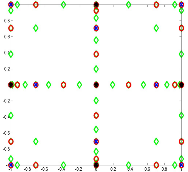
Advantages (vs. stochastic Galerkin):

- decouples algebraic system (like MC)
- applies in a straightforward way to nonlinear random terms

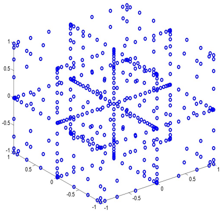
Examples of sparse grids

At right: 2D

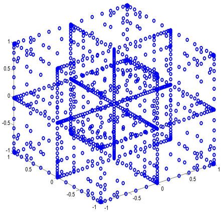
$*$ Level $p = 1$
$\times$ Level $p = 2$
$\circ$ Level $p = 3$
$\diamond$ Level $p = 4$



Below: 3D



Level $p = 5$



Level $p = 6$

Properties of These Methods

For both Galerkin and collocation

- Each computes a discrete function $u^{(hp)}$
- Moments of u estimated using moments of $u^{(hp)}$ (cheap)
- Convergence: $\|E(u) - E(u^{(hp)})\|_{H_1(\mathcal{D})} \leq c_1 h + c_2 r^p$, $r < 1$
  Exponential in polynomial degree
- Contrast with Monte Carlo:
  Perform N_{MC} (discrete) PDE solves to obtain samples $\{u_h^{(s)}\}_{s=1}^{N_{MC}}$
  Moments from averaging, e.g., $\hat{E}(u_h) = \frac{1}{N_{MC}} \sum_{s=1}^{N_{MC}} u_h^{(s)}$
  Error $\sim 1/\sqrt{N_{MC}}$

One other thing:

"p" has different meaning for Galerkin and collocation
For comparable accuracy:

$\#$ stochastic dof (Collocation) $\approx 2^p$ ($\#$ stochastic dof (Galerkin))

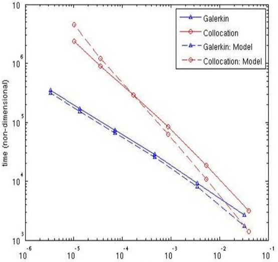
Representative Comparison

Diffusion equation: $-\nabla \cdot (a(\mathbf{x}, \xi) \nabla u) = f$

On unit square with 32×32 finite-difference discretization

Coefficient: Five-term Karhunen-Loève expansion:

$$a(\mathbf{x}, \xi) = a_0(\mathbf{x}) + \sum_{r=1}^{m} \sqrt{\lambda_r}\, a_r(\mathbf{x})\, \xi_r, \quad m = 5$$



Reduced Basis Methods for Parameter-Dependent PDEs

Starting point for these examples:

Parameter-dependent PDE $\mathcal{L}_{\boldsymbol{\xi}} u = f$

In examples given:

$\mathcal{L}_{\boldsymbol{\xi}} = -\nabla \cdot (a_0 + \sum_{r=1}^{m} a_r(\mathbf{x})\xi_r)\nabla$

Complication:

Expensive if many realizations (samples of $\boldsymbol{\xi}$) are required

Idea (Patera, Boyaval, Bris, Lelièvre, Maday, Nguyen, ...):

Solve the problem on a *reduced space*

That is: by some means, choose $\boldsymbol{\xi}^{(1)}, \boldsymbol{\xi}^{(2)}, \ldots, \boldsymbol{\xi}^{(n)}$, $n \ll N$

Solve $\mathcal{L}_{\boldsymbol{\xi}^{(i)}} u^{(i)} = f$, $u^{(i)} = u(\cdot, \boldsymbol{\xi}^{(i)})$, $i = 1, \ldots, n$

For other $\boldsymbol{\xi}$, approximate $u(\cdot, \boldsymbol{\xi})$ by $\tilde{u}(\cdot, \boldsymbol{\xi}) \in span\{u^{(1)}, \ldots, u^{(n)}\}$

Terminology: $\{u^{(1)}, \ldots, u^{(n)}\}$ called **snapshots**

Approximation at discrete level:

$$u_h(\cdot,\boldsymbol{\xi}) \approx \tilde{u}_h(\cdot,\boldsymbol{\xi}) \in span\{u_h^{(1)},\ldots,u_h^{(n)}\}$$

Matrix form:

Coefficient matrix $A_{\boldsymbol{\xi}}$, nodal coefficients $\mathbf{u}_h$, $\tilde{\mathbf{u}}_h$, $\mathbf{u}^{(1)},\ldots\mathbf{u}^{(n)}$

Q = orthogonal matrix whose columns span space spanned by $\{\mathbf{u}^{(i)}\}$

Galerkin condition: make residual orthogonal to spanning space

$r = f - A_{\boldsymbol{\xi}}\tilde{\mathbf{u}}_h(\boldsymbol{\xi}) = f - A_{\boldsymbol{\xi}} Q\mathbf{y}_{\boldsymbol{\xi}}$ orthogonal to Q

Result is reduced problem: Galerkin system of order $n \ll N$:

$$[Q^T A Q]\mathbf{y}_{\boldsymbol{\xi}} = Q^T f, \quad \tilde{\mathbf{u}}_h(\boldsymbol{\xi}) = Q\mathbf{y}_{\boldsymbol{\xi}}$$

Goal: Have reduced model capture features of the model
at significantly lower cost

How are costs reduced?

Matrix A of order N

Reduced matrix $Q^T A Q$ of order $n \ll N$

Solving reduced matrix is cheap for small n

Note: making assumption that $\mathcal{L}_{\boldsymbol{\xi}}$ is affinely dependent on $\boldsymbol{\xi}$

$$\mathcal{L}_{\boldsymbol{\xi}} = \sum_{i=1}^{k} \phi_i(\boldsymbol{\xi})\mathcal{L}_i$$
$$\Rightarrow A_{\boldsymbol{\xi}} = \sum_{i=1}^{k} \phi_i(\boldsymbol{\xi})A_i$$
$$\Rightarrow Q^T A_{\boldsymbol{\xi}} A = \sum_{i=1}^{k} \phi_i(\boldsymbol{\xi})[Q^T A_i Q]$$

Means: constructing reduced matrix for new $\boldsymbol{\xi}$ is cheap

Key question: does reduced basis capture features of model?

Strategy for generating a basis / choosing snapshots (Patera, et al.):

For $\tilde{u}_h(\cdot,\boldsymbol{\xi}) \approx u_h(\cdot,\boldsymbol{\xi})$ (equivalently, $\tilde{\mathbf{u}}_{\boldsymbol{\xi}} \approx \mathbf{u}_{\boldsymbol{\xi}}$), use an **error indicator** $\eta(\tilde{u}_h) \approx \|e_h\|$, $e_h = u_h - \tilde{u}_h$

Given: a set of candidate parameters $\mathcal{X} = \{\boldsymbol{\xi}\}$,
an initial choice $\boldsymbol{\xi}^{(1)} \in \mathcal{X}$, and $u^{(1)} = u(\cdot,\boldsymbol{\xi}^{(1)})$

Set $Q = \mathbf{u}^{(1)}$

`while` $max_{\boldsymbol{\xi}\in\mathcal{X}}\left(\eta(\tilde{u}_h(\cdot,\boldsymbol{\xi}))\right) > \tau$

compute $\tilde{u}_h(\cdot,\boldsymbol{\xi})$, $\eta(\tilde{u}_h(\cdot,\boldsymbol{\xi}))$, $\forall\,\boldsymbol{\xi}\in\mathcal{X}$ % use current reduced

let $\boldsymbol{\xi}^* = argmax_{\boldsymbol{\xi}\in\mathcal{X}}\left(\eta(\tilde{u}_h(\cdot,\boldsymbol{\xi}))\right)$ % basis

`if` $\eta(\tilde{u}_h(\cdot,\boldsymbol{\xi}^*)) > \tau$ `then`

augment basis with $u_h(\cdot,\boldsymbol{\xi}^*)$, update Q with $\mathbf{u}_{\boldsymbol{\xi}^*}$

`endif`

`end`

Potentially expensive, but just viewed as "offline" *preprocessing*
"*Online*" simulation done using reduced basis

Reduced Basis + Sparse Grid Collocation

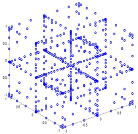
Adapt to sparse grid collocation: Recall collocation solution

$$u_q^{(hp)}(x,\boldsymbol{\xi}^{(k)}) = \sum_{\boldsymbol{\xi}^{(k)}\in\Theta_q} u_c(x,\boldsymbol{\xi}^{(k)}) L_{\boldsymbol{\xi}^{(k)}}(\boldsymbol{\xi}) \qquad (1)$$

Main ideas:

1. Use sparse grid collocation points as candidate set $\mathcal{X}$,
2. Use reduced solution as coefficient $u_c(\cdot,\boldsymbol{\xi}^{(k)})$ whenever possible

Algorithm

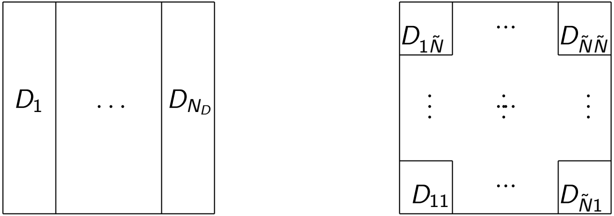


for each sparse grid level p
    for each point $\boldsymbol{\xi}^{(k)}$ at level p
        compute reduced solution $u_R(\cdot,\boldsymbol{\xi}^{(k)})$
        if $\eta(u_R(\cdot,\boldsymbol{\xi}^{(k)})) \le \tau$, then
            use $u_R(\cdot,\boldsymbol{\xi}^{(k)})$ as coefficient $u_c(\cdot,\boldsymbol{\xi}^{(k)})$ in (1)
        else
            compute snapshot $u_h(\cdot,\boldsymbol{\xi}^{(k)})$, use it as $u_c(\cdot,\boldsymbol{\xi}^{(k)})$ in (1)
            augment reduced basis with $u_h(\cdot,\boldsymbol{\xi}^{(k)})$, update Q with $\mathbf{u}_{\boldsymbol{\xi}^{(k)}}$
        endif
    end
end

To Assess Effectiveness

Benchmark problems: Diffusion equation $-\nabla \cdot (a(\mathbf{x}, \boldsymbol{\xi})\nabla u) = f$ in $\mathbb{R}^2$

Piecewise constant diffusion coefficient parameterized as a random variable $\boldsymbol{\xi} = [\xi_1, \cdots, \xi_{N_D}]^T$ independently and uniformly distributed in $\Gamma = [0.01, 1]^{N_D}$



(a) Case 1: N_D subdomains (b) Case 2: $N_D = \tilde{N} \times \tilde{N}$ subdomains

To assess effectiveness: consider

Full snapshot set, set of snapshots for all possible parameter values:

$$\mathcal{S}_\Gamma := \{u_h(\cdot, \boldsymbol{\xi}), \boldsymbol{\xi} \in \Gamma\}$$

Finite snapshot set, for finite $\Theta \subset \Gamma$:

$$\mathcal{S}_\Theta := \{u_h(\cdot, \boldsymbol{\xi}), \boldsymbol{\xi} \in \Theta\}$$

Question:

How many samples $\{\boldsymbol{\xi}\}$ / $\{u_h(\cdot, \boldsymbol{\xi})\}$ are needed to accurately represent the features of $\mathcal{S}_\Gamma$?

Experiment: to gain insight into this, estimate "rank" of $\mathcal{S}_\Gamma$

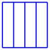
Generate a large set Θ of samples of $\boldsymbol{\xi}$
Generate the finite snapshot set $\mathcal{S}_\Theta$ associated with Θ
Construct the matrix S_Θ of coefficient vectors $\mathbf{u}_{\boldsymbol{\xi}}$ from $\mathcal{S}_\Theta$
Compute the rank of S_Θ

Results follow. Used 3000 samples

Experiment was repeated ten times with similar results

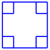
Estimated ranks of $\mathcal{S}_\Gamma$ for two classes of benchmark problems

Case 1



Grid \ N_D	2	3	4	5	6	7	8	9	10
$33^2 = 1089$	3	12	18	30	40	53	55	76	84
$65^2 = 4225$	3	12	18	30	40	48	55	70	87
$129^2 = 16641$	3	12	18	28	39	48	55	72	81

Case 2



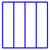
Grid \ N_D	4	9	16	25	36	49	64
$33^2 = 1089$	27	121	193	257	321	385	449
$65^2 = 4225$	28	148	290	465	621	769	897
$129^2 = 16641$	28	153	311	497	746	1016	1298

Trends:

- Rank is dramatically smaller than problem dimension N
- Rank is independent of problem dimension ($\sim$ (mesh size)$^{-2}$)
- In most cases, cost of treating reduced problem of given rank is low

Comparison with algorithm performance

Case 1



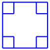
Case 1, 5 × 1 subdomains, 65 × 65 grid, rank=30

q	6	7	8	9	10	11	12	13	16
tol \ $\lvert\Theta_q\rvert$	11	61	241	801	2433	7K	19K	52K	870K
10^{-3}	10	9	0	0	0	0	0	0	0
10^{-4}	10	11	1	0	0	0	0	0	0
10^{-5}	10	13	0	0	0	0	0	0	0

Case 1, 9 × 1 subdomains, 65 × 65 grid, rank=70, $tol = 10^{-4}$

q	10	11	12	13	14	15	16	17
$\lvert\Theta_q\rvert$	19	181	1177	6001	26017	100897	361249	1218049
$N_{full\ solve}$	18	34	2	1	1	0	0	0

Case 2



Comparison with algorithm performance

Case 2, 2×2 subdomains, 65×65 grid, rank=28

q	5	6	7	8	9	10	11	12	15
$\lvert\Theta_q\rvert$ / tol	9	41	137	401	1105	2.9K	7.5K	18.9K	272K
10^{-3}	7	11	3	0	0	0	0	0	0
10^{-4}	7	12	3	0	0	0	0	0	0
10^{-5}	7	13	2	3	0	0	0	0	0

Case 2, 4×4 subdomains, 65×65 grid, rank=290, $tol = 10^{-4}$

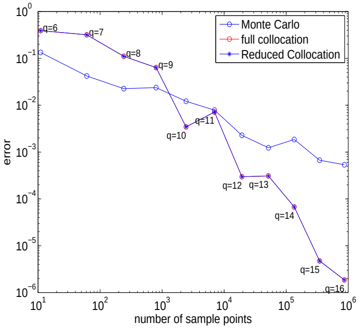
q	17	18	19	20	21
$\lvert\Theta_q\rvert$	33	545	6049	51137	353729
$N_{full\ solve}$	32	168	27	3	4

To assess accuracy: Examine error (vs. reference solution) in expected values of full or reduced collocation solution:

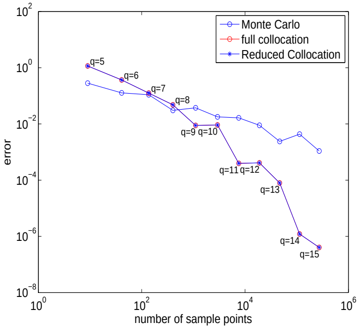
Full collocation $$\epsilon_h := \left\| \tilde{\mathbb{E}}\left(u_q^{hsc}\right) - \tilde{\mathbb{E}}\left(u_r^{hsc}\right) \right\|_0 \Big/ \left\| \tilde{\mathbb{E}}\left(u_r^{hsc}\right) \right\|_0$$

Reduced collocation $$\epsilon_R := \left\| \tilde{\mathbb{E}}\left(u_q^{rsc}\right) - \tilde{\mathbb{E}}\left(u_r^{hsc}\right) \right\|_0 \Big/ \left\| \tilde{\mathbb{E}}\left(u_r^{hsc}\right) \right\|_0$$

Case 1: vertical subdomains



Case 2: square subdomains



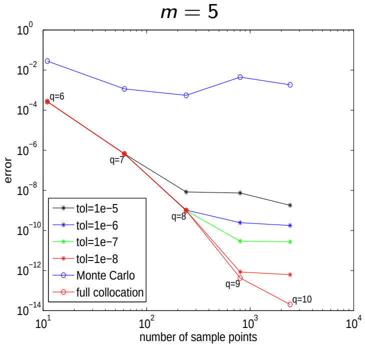
Different example: diffusion coefficient with KL expansion:

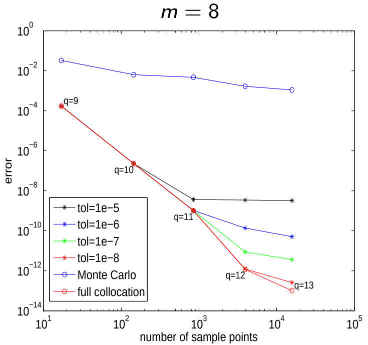
Diffusion coefficient $a_0 + \sigma \sum_{r=1}^{m} \sqrt{\lambda_r} a_r(\mathbf{x}) \xi_r)$

From covariance function $c(\mathbf{x}, \mathbf{y}) = \sigma \exp\left(-\frac{|x_1 - y_1|}{c} - \frac{|x_2 - y_2|}{c} \right)$

Smaller correlation length $c \sim$ more terms m

Examine $c = 4$, $m = 4$ and $c = 2.5$, $m = 8$.





Concluding Remarks

For PDEs with uncertain, parameter-dependent coefficients:

- *Spectral methods*: stochastic Galerkin, stochastic collocation, offer prospects for fast solution
- They suffer from "the curse of dimensionality"
- Costs of collocation can be reduced using reduced basis methodology