Università di Bologna

An Efficient Reduced Basis Solver for Stochastic Galerkin Matrix Equations

V. Simoncini

Dipartimento di Matematica, Università di Bologna

valeria.simoncini@unibo.it

Joint with Catherine Powell and David Silvester, Univ. Manchester

Elliptic PDE problems with random inputs

Stochastic steady-state diffusion eqn.

Given $D \subset \mathbb{R}^2$ *and* Ω *sample space of prob.space, find* $u : D \times \Omega \to \mathbb{R}$ *s.t.* $\mathbb{P}$*-a.s.,*

$$\begin{cases} -\nabla \cdot (a(\vec{x},\omega)\nabla u(\vec{x},\omega)) = & f(\vec{x}) \quad in \;\; D \\ u(\vec{x},\omega) = & 0 \qquad\quad on \;\; \partial D \end{cases}$$

f: deterministic;

a: random field, linear function of finite no. of real-valued random variables $\xi_r : \Omega \to \Gamma_r \subset \mathbb{R}$

$$a(\vec{x},\omega) = a_0(\vec{x}) + \sum_{r=1}^{m} a_r(\vec{x})\,\xi_r(\omega)$$

Elliptic PDE problems with random inputs

Common choice: truncated Karhunen–Loève (KL) expansion:

$$a(\vec{x},\omega) = \mu(\vec{x}) + \sigma \sum_{r=1}^{m} \sqrt{\lambda_r}\phi_r(\vec{x})\xi_r(\omega),$$

$\mu(\vec{x})$: expected value of diffusion coef. σ: std dev.

$(\lambda_r, \phi_r(\vec{x}))$ eigs of the integral operator $\mathcal{B}$ wrto $B(\vec{x},\vec{x}') = \frac{1}{\sigma^2}C(\vec{x},\vec{x}')$ with $\lambda_r \searrow$ and $C : D \times D \to \mathbb{R}$ covariance function

Discretization by stochastic Galerkin

Approximate with space in tensor product form[a] $Z_h \times S_p$

$$\mathcal{A}\mathbf{x} = \mathbf{b}, \qquad \mathcal{A} = G_0 \otimes K_0 + \sum_{r=1}^{m} G_r \otimes K_r, \quad \mathbf{b} = \mathbf{g}_0 \otimes \mathbf{f}_0$$

$\mathcal{A} \in \mathbb{R}^{n_x n_\xi \times n_x n_\xi}$

$\mathbf{x}$: expansion coef. of approx to u in the tensor product basis $\{\varphi_i \psi_k\}$

$K_r \in \mathbb{R}^{n_x \times n_x}$, FE matrices (sym)

$G_r \in \mathbb{R}^{n_\xi \times n_\xi}$, Galerkin matrices associated w/ S_p (sym.), spec$(G_r) \in [-1, 1]$

$\mathbf{g}_0$: first column of G_0

$\mathbf{f}_0$: FE rhs of deterministic PDE

[a] $Z_h \subset H_0^1(D)$ FE space, and S_p set of multivariate polyn of total degree $\leq p$

Discretization by stochastic Galerkin

Approximate with space in tensor product form[a] $Z_h \times S_p$

$$\mathcal{A}\mathbf{x} = \mathbf{b}, \qquad \mathcal{A} = G_0 \otimes K_0 + \sum_{r=1}^{m} G_r \otimes K_r, \quad \mathbf{b} = \mathbf{g}_0 \otimes \mathbf{f}_0$$

$\mathcal{A} \in \mathbb{R}^{n_x n_\xi \times n_x n_\xi}$

$\mathbf{x}$: expansion coef. of approx to u in the tensor product basis $\{\varphi_i \psi_k\}$

$K_r \in \mathbb{R}^{n_x \times n_x}$, FE matrices (sym)

$G_r \in \mathbb{R}^{n_\xi \times n_\xi}$, Galerkin matrices associated w/ S_p (sym.), spec$(G_r) \in [-1, 1]$

$\mathbf{g}_0$: first column of G_0

$\mathbf{f}_0$: FE rhs of deterministic PDE

$$n_\xi = \dim(S_p) = \frac{(m+p)!}{m!p!} \qquad \Rightarrow \boxed{n_x \cdot n_\xi} \text{ huge}$$

[a] $Z_h \subset H_0^1(D)$ FE space, and S_p set of multivariate polyn of total degree $\leq p$

The matrix equation formulation

$$(G_0 \otimes K_0 + G_1 \otimes K_1 + \ldots + G_m \otimes K_m)\,\mathbf{x} = \mathbf{g}_0 \otimes \mathbf{f}_0$$

Equivalent matrix-oriented form:

- Set $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_{n_\xi}]$ (that is, $\mathbf{x} = \mathrm{vec}(\mathbf{X})$)
- Transforms system into

$$K_0\mathbf{X}G_0 + K_1\mathbf{X}G_1 + \ldots + K_m\mathbf{X}G_m = F, \qquad F = \mathbf{f}_0\mathbf{g}_0^\top$$

where we used $(G_r \otimes K_r)\mathbf{x} = \mathrm{vec}(K_r\mathbf{X}G_r) \quad (G_0 = I)$

The matrix equation formulation

$$(G_0 \otimes K_0 + G_1 \otimes K_1 + \ldots + G_m \otimes K_m)\,\mathbf{x} = \mathbf{g}_0 \otimes \mathbf{f}_0$$

Equivalent matrix-oriented form:

- Set $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_{n_\xi}]$ (that is, $\mathbf{x} = \mathrm{vec}(\mathbf{X})$)
- Transforms system into

$$K_0\mathbf{X}G_0 + K_1\mathbf{X}G_1 + \ldots + K_m\mathbf{X}G_m = F, \qquad F = \mathbf{f}_0\mathbf{g}_0^\top$$

where we used $(G_r \otimes K_r)\mathbf{x} = \mathrm{vec}(K_r\mathbf{X}G_r) \quad (G_0 = I)$

Starting Point: it is possible to construct low rank approximation

$$\mathbf{X} \approx \widetilde{\mathbf{X}} = \left[\quad \vphantom{\begin{matrix}a\\a\\a\\a\end{matrix}}\right] [\qquad\qquad\qquad] = (Spatial) \times (Stoch)$$

(Benner, Dolgov, Keese, Grasedyck, Khoromskij, Kressner, Matthies, Onwunta, Schwab, Stoll, Zander, . . .)

Matrix Galerkin approximation of the deterministic part

$$K_0\mathbf{X}G_0 + K_1\mathbf{X}G_1 + \ldots + K_m\mathbf{X}G_m = \mathbf{f}_0\mathbf{g}_0^\top$$

Approximation space $\mathcal{K}_k$ and basis matrix V_k: $\mathbf{X} \approx \widetilde{\mathbf{X}} = V_k\mathbf{Y}$

$$V_k^\top R_k = 0, \qquad R_k := K_0\widetilde{\mathbf{X}} + K_1\widetilde{\mathbf{X}}G_1 + \ldots + K_m\widetilde{\mathbf{X}}G_m - \mathbf{f}_0\mathbf{g}_0^\top$$

Computational challenges:

- Generation of $\mathcal{K}_k$ involves $m+1$ different matrices $\{K_r\}$!
- Matrices K_r have different spectral properties
- n_x, n_ξ so large that $\widetilde{\mathbf{X}}, R_k$ should not be formed explicitly!

The reduced problem. Rough idea.

For $\widetilde{\mathbf{X}} = V_k \mathbf{Y}$ and $R_k = K_0 \widetilde{\mathbf{X}} + K_1 \widetilde{\mathbf{X}} G_1 + \ldots + K_m \widetilde{\mathbf{X}} G_m - \mathbf{f}_0 \mathbf{g}_0^\top$

$$V_k^\top R_k = 0$$

corresponds to

$$V_k^\top (K_0 \widetilde{\mathbf{X}} + K_1 \widetilde{\mathbf{X}} G_1 + \ldots + K_m \widetilde{\mathbf{X}} G_m - \mathbf{f}_0 \mathbf{g}_0^\top) = 0$$

that is

$$(V_k^\top K_0 V_k) \mathbf{Y} + \ldots + (V_k^\top K_m V_k) \mathbf{Y} G_m = (V_k^\top \mathbf{f}_0) \mathbf{g}_0^\top \qquad (*)$$

with $V_k^\top K_r V_k$, $r = 0, \ldots, m$ much smaller dimensions than K_r

$\Rightarrow$ Solve (*) via Kronecker formulation

(either with direct or iterative method - matrix version of CG)

The actual procedure

Preprocessing:

- “Divide” by K_0 (stiffness matrix)

$$\widehat{\mathbf{X}} + \widehat{K}_1 \widehat{\mathbf{X}} G_1 + \ldots + \widehat{K}_m \widehat{\mathbf{X}} G_m = \widehat{\mathbf{f}}_0 \mathbf{g}_0^\top$$

with $\mathrm{spec}(\widehat{K}_r) \subset [-\tau_r, \tau_r]$, $\tau_r = a_0^{-1} \|a_r\|_{L^\infty(D)}$

(sharper bounds if $a_r(\vec{x}) > \nu_r > 0$)

The actual procedure

Preprocessing:

- "Divide" by K_0 (stiffness matrix)

$$\widehat{\mathbf{X}} + \widehat{K}_1 \widehat{\mathbf{X}} G_1 + \ldots + \widehat{K}_m \widehat{\mathbf{X}} G_m = \widehat{\mathbf{f}}_0 \mathbf{g}_0^\top$$

  with $\mathrm{spec}(\widehat{K}_r) \subset [-\tau_r, \tau_r]$, $\tau_r = a_0^{-1} \|a_r\|_{L^\infty(D)}$

  (sharper bounds if $a_r(\vec{x}) > \nu_r > 0$)

- Shift matrices by α_r, $r = 1, \ldots, m$ to get similar *positive* spectral intervals

$$\widehat{\mathbf{X}} \widehat{G}_0 + (\widehat{K}_1 + \alpha_1 I) \widehat{\mathbf{X}} G_1 + \ldots + (\widehat{K}_r + \alpha_m I) \widehat{\mathbf{X}} G_m = \widehat{\mathbf{f}}_0 \mathbf{g}_0^\top$$

  (with $\widehat{G}_0 = I - \sum_{r=1}^{m} \alpha_r G_r$). If $\sum_{r=1}^{\infty} \|a_r\|_\infty < a_0^{\min}$ then $\alpha_r = 1$ is enough!

The actual procedure

Preprocessing:

- "Divide" by K_0 (stiffness matrix)

$$\widehat{\mathbf{X}} + \widehat{K}_1\widehat{\mathbf{X}}G_1 + \ldots + \widehat{K}_m\widehat{\mathbf{X}}G_m = \widehat{\mathbf{f}}_0\mathbf{g}_0^\top$$

  with $\mathrm{spec}(\widehat{K}_r) \subset [-\tau_r, \tau_r]$, $\tau_r = a_0^{-1}\|a_r\|_{L^\infty(D)}$

  (sharper bounds if $a_r(\vec{x}) > \nu_r > 0$)

- Shift matrices by α_r, $r = 1, \ldots, m$ to get similar *positive* spectral intervals

$$\widehat{\mathbf{X}}\widehat{G}_0 + (\widehat{K}_1 + \alpha_1 I)\widehat{\mathbf{X}}G_1 + \ldots + (\widehat{K}_r + \alpha_m I)\widehat{\mathbf{X}}G_m = \widehat{\mathbf{f}}_0\mathbf{g}_0^\top$$

  (with $\widehat{G}_0 = I - \sum_{r=1}^{m} \alpha_r G_r$). If $\sum_{r=1}^{\infty} \|a_r\|_\infty < a_0^{\min}$ then $\alpha_r = 1$ is enough!

Subspace generation:

- Generate $\mathcal{K}_k$ related to $\bigcup_r \mathbb{K}_k(\widehat{K}_r + \alpha_r I, \widehat{\mathbf{f}}_0)$
  with $\mathbb{K}_k(\widehat{K}_r + \alpha_r I, \widehat{\mathbf{f}}_0)$ (adaptive) rational Krylov subspace

Generation of approximation space. Some details

$$\mathcal{K}_k \text{ related to } \bigcup_r \mathbb{K}_k(\widehat{K}_r + \alpha_r I, \widehat{\mathbf{f}}_0, \mathbf{s}_r)$$

with adaptive rational Krylov subspace:

$$\mathbb{K}_k(\widehat{K}_r{+}\alpha_r I, \widehat{\mathbf{f}}_0, \mathbf{s}_r) := \text{span}\left\{\widehat{\mathbf{f}}_0, (\widehat{K}_r + \alpha_r I + s_{r_1} I)^{-1}\widehat{\mathbf{f}}_0, \ldots, \prod_{j=1}^{k}(\widehat{K}_r + \alpha_r I + s_{r_j} I)^{-1}\widehat{\mathbf{f}}_0\right\},$$

Generation of approximation space. Some details

$$\mathcal{K}_k \text{ related to } \bigcup_r \mathbb{K}_k(\widehat{K}_r + \alpha_r I, \widehat{\mathbf{f}}_0, \mathbf{s}_r)$$

with adaptive rational Krylov subspace:

$$\mathbb{K}_k(\widehat{K}_r + \alpha_r I, \widehat{\mathbf{f}}_0, \mathbf{s}_r) := \text{span}\left\{\widehat{\mathbf{f}}_0, (\widehat{K}_r + \alpha_r I + s_{r_1} I)^{-1}\widehat{\mathbf{f}}_0, \ldots, \prod_{j=1}^{k}(\widehat{K}_r + \alpha_r I + s_{r_j} I)^{-1}\widehat{\mathbf{f}}_0\right\},$$

- Space $\mathcal{K}_k$ grows at most r vectors at the time
- Space actually generated (ex. $m = 3$):

$$\begin{aligned} \text{span} \quad & \{\mathbf{v}_0, \underbrace{(\widehat{K}_1 + \alpha_1 I + s_1 I)^{-1}\mathbf{v}_0}_{=:\mathbf{v}_1}, \underbrace{(\widehat{K}_2 + \alpha_2 I + s_1 I)^{-1}\mathbf{v}_0}_{=:\mathbf{v}_2}, \underbrace{(\widehat{K}_3 + \alpha_3 I + s_1 I)^{-1}}_{=:\mathbf{v}_3}\mathbf{v}_0, \\ & (\widehat{K}_1 + \alpha_1 I + s_2 I)^{-1}\mathbf{v}_1, (\widehat{K}_2 + \alpha_2 I + s_2 I)^{-1}\mathbf{v}_1, (\widehat{K}_3 + \alpha_3 I + s_2 I)^{-1}\mathbf{v}_1, \\ & (\widehat{K}_1 + \alpha_1 I + s_3 I)^{-1}\mathbf{v}_2, (\widehat{K}_2 + \alpha_2 I + s_3 I)^{-1}\mathbf{v}_2, (\widehat{K}_3 + \alpha_3 I + s_3 I)^{-1}\mathbf{v}_2\} \end{aligned}$$

- Under certain (natural) conditions on the data, α_r and s_j can be combined:

$$s_* := \alpha_r + s_j = 2$$

Parameter-free procedure! (...and preprocessing avoided)

Practical considerations and comparisons

- An exact reduction procedure:
  - True orthonormal basis
  - True Galerkin condition on the residual

(as opposed to truncation based CG-type strategies)

Practical considerations and comparisons

- An exact reduction procedure:
  - True orthonormal basis
  - True Galerkin condition on the residual

  (as opposed to truncation based CG-type strategies)

- Mesh independent performance:
  - Builds upon robust "preconditioning" procedure with K_0 for mild ratio σ/μ
  - Exploits problem spectral properties

Practical considerations and comparisons

- An exact reduction procedure:
  - True orthonormal basis
  - True Galerkin condition on the residual

  (as opposed to truncation based CG-type strategies)

- Mesh independent performance:
  - Builds upon robust “preconditioning” procedure with K_0 for mild ratio σ/μ
  - Exploits problem spectral properties

- Low memory requirements:
  - Memory and computing costs $\ll \mathcal{O}(n_\xi n_x)$
  - Allow for more accurate simulations in stochastic space

Example 1. SIFISS 1.0, Q1 Finite Elements. Eigel etal.

$$-\nabla \cdot (a \nabla u) = 1, \quad D = (0,1)^2, \quad u|_{\partial D} = 0, \ \text{(fast decay)}$$

$a_r(\vec{x}) = \gamma_r \cos(\chi_1 x_1) \cos(\chi_2 x_2)$ $\quad n_x = 65,025$, tol$=10^{-5}$ (relative soln change)

Example 1. SIFISS 1.0, Q1 Finite Elements. Eigel etal.

$$-\nabla \cdot (a\nabla u) = 1, \quad D = (0,1)^2, \quad u|_{\partial D} = 0, \text{ (fast decay)}$$

$a_r(\vec{x}) = \gamma_r \cos(\chi_1 x_1) \cos(\chi_2 x_2)$ $n_x = 65,025$, tol$=10^{-5}$ (relative soln change)

m	p	n_ξ	k	inner	n_k	rank	time	CG
				its	$\mathcal{K}_k$	$\widetilde{\mathbf{X}}$	secs	time (its)
5 $\gamma_r \geq 10^{-3}$	2	21	16	12.9	66	19	21.4	8.1 (10)
	3	56	19	15.6	77	28	26.5	21.7 (12)
	4	126	19	16.7	77	36	25.2	53.1 (14)
	5	252	23	18.9	94	42	32.3	103.0 (14)
9 $\gamma_r \geq 10^{-4}$	2	55	14	13.3	79	26	32.7	20.3 (10)
	3	220	16	15.6	94	34	38.6	91.1 (12)
	4	715	17	17.1	100	42	42.2	335.0 (14)
	5	2,002	18	18.3	102	47	49.1	935.0 (14)
16 $\gamma_r \geq 10^{-5}$	2	153	12	12.7	82	32	50.4	67.6 (10)
	3	969	14	15.1	106	41	61.9	490.0 (12)
	4	4,845	15	16.7	117	46	84.7	2810.0 (14)
	5	20,349	15	17.5	117	51	215.0	Out of Mem

CPU time in secs. Dell Precision T7500 w/ 12 cores and 48GB RAM

Example 2. $-\nabla \cdot (a\nabla u) = 1, \quad D = (-1,1)^2$. KL expansion.

$\mu = 1$, $\xi_r \sim U(-\sqrt{3}, \sqrt{3})$ and $C(\vec{x}_1, \vec{x}_2) = \sigma^2 \exp\left(-\frac{\|\vec{x}_1 - \vec{x}_2\|_1}{2}\right)$, $n_x = 65,025$, $\sigma = 0.3$

m	p	n_ξ	k	inner	n_k	rank	time	CG
				its	$\mathcal{K}_k$	$\widetilde{\mathbf{X}}$	secs	time (its)
	2	45	17	9.8	128	45	32.1	13.4 (8)
8	3	165	21	12.2	160	129	41.4	56.6 (10)
87%	4	495	24	14.5	183	178	51.1	197.0 (12)
	5	1,287	27	16.9	207	207	64.0	553.0 (13)
	2	91	15	9.9	165	89	47.8	30.0 (8)
12	3	455	18	12.2	201	196	61.6	175.0 (10)
89%	4	1,820	21	15.0	236	236	86.4	821.0 (12)
	5	6,188	25	18.6	281	281	188.0	3070.0 (13)
	2	231	16	9.4	281	206	111.0	94.7 (8)
20	3	1,771	23	12.3	399	399	197.0	845.0 (10)
93%	4	10,626	26	15.4	454	454	556.0	Out of Mem

% of variance integral of a

Conclusions

- Reduced basis solver allows for large probability space

  $\Rightarrow$ more than 10^9 total degrees of freedom

  in few minutes on desktop computers and low memory requirements
- Parameter-free procedure (mesh size h and α, s_j)
- Mild dependence on m (expansion truncation) and p (polyn degree)

Conclusions

- Reduced basis solver allows for large probability space

  $\Rightarrow$ more than 10^9 total degrees of freedom

  in few minutes on desktop computers and low memory requirements
- Parameter-free procedure (mesh size h and α, s_j)
- Mild dependence on m (expansion truncation) and p (polyn degree)

Outlook:

- Develop 3D (in space)
- Exploit tensorized space
- Derive a reduction strategy in probability space

Reference

An efficient reduced basis solver for stochastic Galerkin matrix equations

Catherine E. Powell, David Silvester, and V. Simoncini

MIMS EPrint 2015.64.

available at `www.dm.unibo.it/~simoncin`