Reduced Basis Methods: Certified Machine Learning for Parametric Partial Differential Equations

Yanlai Chen

UMass Dartmouth

Lectures Series on High-Order Numerical Methods

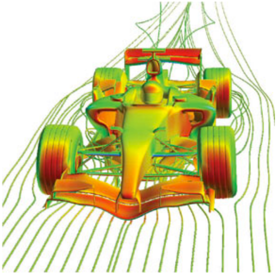
2020

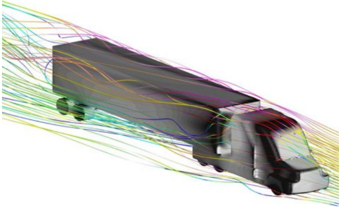
Outline

- First hour: a Reduced Basis Method (RBM) primer
- Second hour: Theory, Empirical Interpolation Method (EIM), Successive Constraint Method (SCM)
- Third hour: Examples of what RBM can do
- Fourth hour: Recent works addressing two challenges

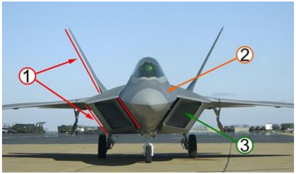
RBM Applications and motivations: Multi-query context

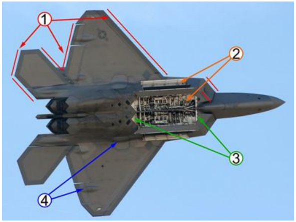
Aerodynamics





Stealth Technology

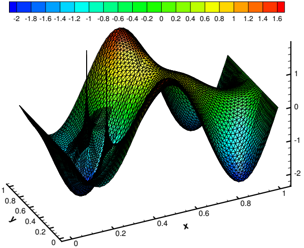




Optimization, Inverse problems, Sensitivity analysis, Uncertainty quantification . . .

RBM: Model Reduction intuition and idea

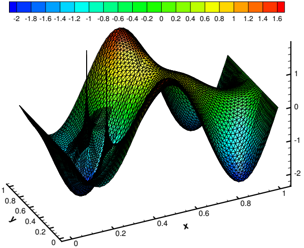
Traditional methods:



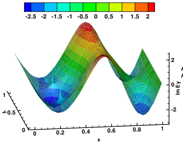
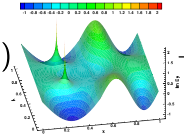
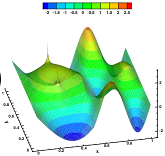
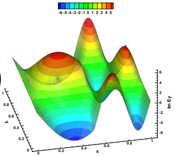
Reduced basis method (RBM) relies on traditional methods but solves (some) parameter-dependent problems (e.g. $-\nabla \cdot (\kappa \nabla u) + c\, u = f$) much faster.

RBM: Model Reduction intuition and idea

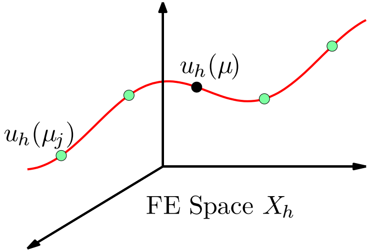
Traditional methods:



Reduced basis method (RBM) relies on traditional methods but solves (some) parameter-dependent problems (e.g. $-\nabla \cdot (\kappa \nabla u) + c\, u = f$) much faster.

 $\approx c_1(\mu)$  $+c_2(\mu)$  $+c_3(\mu)$ 

RBM: Model Reduction intuition and idea



Traditional method looks for a solution in X_h.

RBM: a, low-order yet high-fidelity, approx. around the red curve.

Classical RB: Almroth et al. 1978; Noor & Peters 1980; Porsching 1985

Recent Monographs on RB: Quarteroni et al. 2016; Hesthaven et al. 2016

Recent relevant review articles: Rozza et al. 2007; Haasdonk 2017

Key: A decomposition

	Full simulations	Reduced simulations
Stage	Offline i.e. Training	Online i.e. Testing/Appl.
Purpose	Investment	Return
Duration	Days	Seconds - minutes

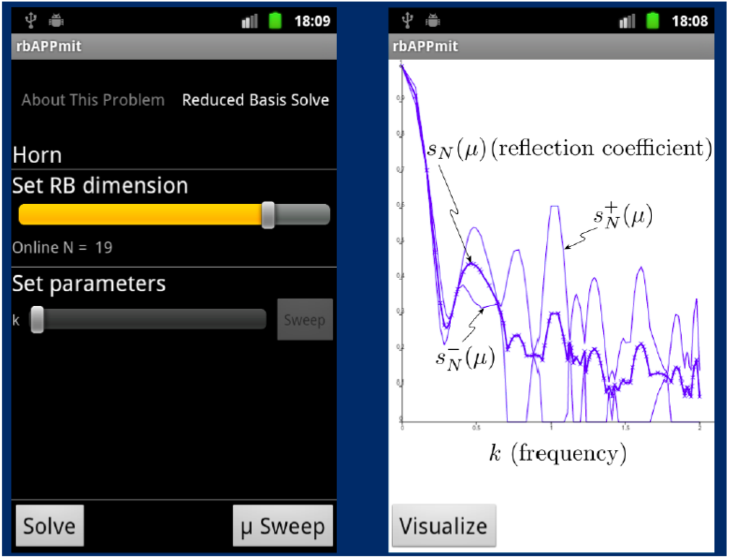
Equipment

What RB can achieve?

Multiple orders of magnitudes of reduction in marginal computation time.

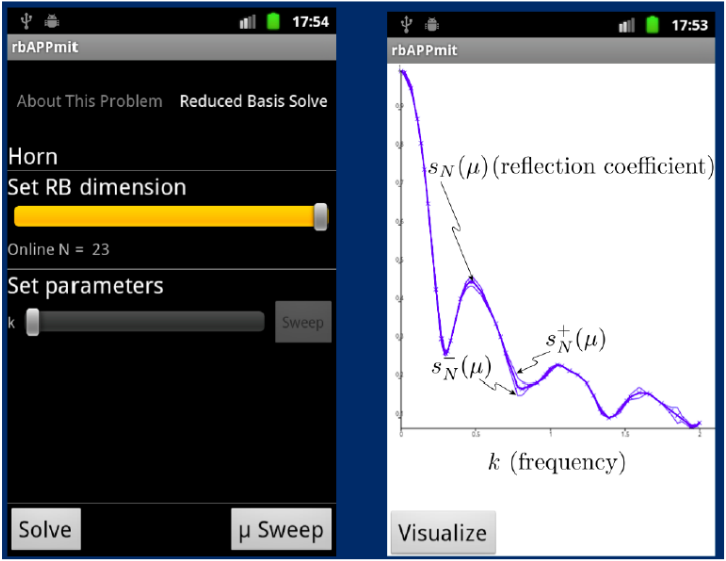
Wide variety/range of parameters describing physical/geometrical properties.

Field-deployment enabled by RB: Load & solve on a phone



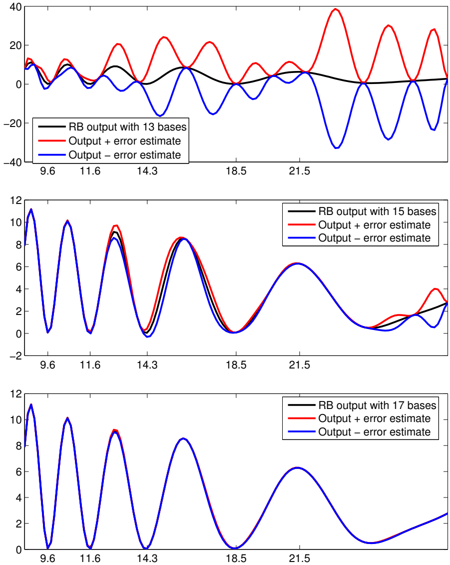
Smartphone RB: Huynh et al. (2011)

Field-deployment enabled by RB: accuracy assured



Smartphone RB: Huynh et al. (2011)

RBM: The surrogate is more than just a surrogate



Low-rank approximation through energy projection

$$-\nabla \cdot (\kappa \nabla u) + c\, u = f \quad \text{in } \Omega; u = 0 \quad \text{on } \partial\Omega. \quad \mu := \{\kappa, c\} \in \mathcal{D}.$$

Low-rank approximation through energy projection

$$-\nabla \cdot (\kappa \nabla u) + c\, u = f \quad \text{in } \Omega; u = 0 \quad \text{on } \partial\Omega. \quad \mu := \{\kappa, c\} \in \mathcal{D}.$$

$$\Downarrow$$

Find $u(\mu) \in H_0^1(\Omega)$ such that $a(u(\mu), v; \mu) = f(v)\ \forall v \in H_0^1(\Omega)$ with

$$a(u, v; \mu) = \int_\Omega (\kappa \nabla u \cdot \nabla v + cuv)\ dx \quad f(v) = \int_\Omega fv\, dx$$

Low-rank approximation through energy projection

$$-\nabla \cdot (\kappa \nabla u) + c\, u = f \quad \text{in } \Omega; u = 0 \quad \text{on } \partial\Omega. \quad \mu := \{\kappa, c\} \in \mathcal{D}.$$

$$\Downarrow$$

$$\text{Find } u(\mu) \in H_0^1(\Omega) \text{ such that } a(u(\mu), v; \mu) = f(v)\ \forall v \in H_0^1(\Omega) \text{ with}$$

$$a(u, v; \mu) = \int_\Omega (\kappa \nabla u \cdot \nabla v + cuv)\ dx \quad f(v) = \int_\Omega fv\, dx$$

$$\Downarrow$$

$$\text{FEM : Find } u_h(\mu) \in V_h \text{ such that } a_h(u_h(\mu), v; \mu) = f_h(v)\ \forall v \in V_h$$

$$\dim(V_h) = \mathcal{N} \quad (u_h(\mu) : \text{Truth Approximation})$$

Low-rank approximation through energy projection

$$-\nabla \cdot (\kappa \nabla u) + c\, u = f \quad \text{in } \Omega; u = 0 \quad \text{on } \partial\Omega. \quad \mu := \{\kappa, c\} \in \mathcal{D}.$$

$$\Downarrow$$

$$\text{Find } u(\mu) \in H_0^1(\Omega) \text{ such that } a(u(\mu), v; \mu) = f(v)\ \forall v \in H_0^1(\Omega) \text{ with}$$

$$a(u, v; \mu) = \int_\Omega (\kappa \nabla u \cdot \nabla v + cuv)\ dx \quad f(v) = \int_\Omega fv\, dx$$

$$\Downarrow$$

$$\text{FEM : Find } u_h(\mu) \in V_h \text{ such that } a_h(u_h(\mu), v; \mu) = f_h(v)\ \forall v \in V_h$$

$$\dim(V_h) = \mathcal{N} \quad (u_h(\mu) : \text{Truth Approximation})$$

$$\Downarrow$$

$$\text{RBM : Find } u_N(\mu) \in W_{RB} \text{ such that } a_h(u_N(\mu), v; \mu) = f_h(v)\ \forall v \in W_{RB}$$

$$\text{with} \dim(W_{RB}) = N << \mathcal{N}, \ W_{RB} \subset V_h$$

Low-rank approximation through energy projection

$$-\nabla \cdot (\kappa \nabla u) + c\, u = f \quad \text{in } \Omega; u = 0 \quad \text{on } \partial\Omega. \quad \mu := \{\kappa, c\} \in \mathcal{D}.$$

$$\Downarrow$$

Find $u(\mu) \in H_0^1(\Omega)$ such that $a(u(\mu), v; \mu) = f(v)\ \forall v \in H_0^1(\Omega)$ with

$$a(u, v; \mu) = \int_\Omega (\kappa \nabla u \cdot \nabla v + cuv)\, dx \quad f(v) = \int_\Omega fv\, dx$$

$$\Downarrow$$

FEM : Find $u_h(\mu) \in V_h$ such that $a_h(u_h(\mu), v; \mu) = f_h(v)\ \forall v \in V_h$

$$\dim(V_h) = \mathcal{N} \quad (u_h(\mu) : \text{Truth Approximation})$$

$$\Downarrow$$

RBM : Find $u_N(\mu) \in W_{RB}$ such that $a_h(u_N(\mu), v; \mu) = f_h(v)\ \forall v \in W_{RB}$

with $\dim(W_{RB}) = N << \mathcal{N}$, $W_{RB} \subset V_h$

$$\left[\textbf{Fast decay of } d_N[u(\cdot; \mathcal{D})] := \inf_{\substack{X_N \subset V_h \\ \dim X_N = N}} \sup_{\mu \in \mathcal{D}} \inf_{v \in X_N} \|u(\cdot, \mu) - v\|_{V_h} \right]$$

$a_h(u_h(\mu), v; \mu) = f_h(v), \forall v \in V_h \longrightarrow a_h(u_N(\mu), v; \mu) = f_h(v), \forall v \in W_{RB}$

$a_h(u_h(\mu), v; \mu) = f_h(v), \forall v \in V_h \longrightarrow a_h(u_N(\mu), v; \mu) = f_h(v), \forall v \in W_{RB}$

RBM sets $W_{RB} = \mathrm{span}\{u_h(\mu_1), \ldots, u_h(\mu_N)\}$.
The final system is going from $\mathcal{N} \times \mathcal{N}$ to $N \times N$.

RBM sets $W_{RB} = \mathrm{span}\{u_h(\mu_1),\, \ldots,\, u_h(\mu_N)\}$.
The final system is going from $\mathcal{N} \times \mathcal{N}$ to $N \times N$.
Difficulty: calculations of $a_h(u_h(\mu_i),\, u_h(\mu_j);\, \mu)$ and $f_h(u_h(\mu_i);\, \mu)\ \forall\mu$ are still $\mathcal{N}-$dependent!

RBM sets $W_{RB} = \mathrm{span}\{u_h(\mu_1), \ldots, u_h(\mu_N)\}$.
The final system is going from $\mathcal{N} \times \mathcal{N}$ to $N \times N$.
Difficulty: calculations of $a_h(u_h(\mu_i), u_h(\mu_j); \mu)$ and $f_h(u_h(\mu_i); \mu)$ $\forall \mu$ are still $\mathcal{N}$−dependent!
Solution: affine assumption

$$a(u, v; \mu) \equiv \sum_{q=1}^{Q_a} \Theta_a^q(\mu)\, a^q(u, v), \;\; f(v; \mu) \equiv \sum_{q=1}^{Q_f} \Theta_f^q(\mu)\, f^q(v).$$

RBM sets $W_{RB} = \mathrm{span}\{u_h(\mu_1), \ldots, u_h(\mu_N)\}$.
The final system is going from $\mathcal{N} \times \mathcal{N}$ to $N \times N$.
Difficulty: calculations of $a_h(u_h(\mu_i), u_h(\mu_j); \mu)$ and $f_h(u_h(\mu_i); \mu)$ $\forall \mu$ are still $\mathcal{N}-$dependent!
Solution: affine assumption

$$a(u, v; \mu) \equiv \sum_{q=1}^{Q_a} \Theta_a^q(\mu)\, a^q(u, v), \;\; f(v; \mu) \equiv \sum_{q=1}^{Q_f} \Theta_f^q(\mu)\, f^q(v).$$

Decomposing

$$a(u, v; \mu) = \kappa\, (\nabla u, \nabla v)_\Omega + c\, (u, v)_\Omega$$

$$a(u, v; \mu) = \nu^{-1}\, (\nabla \times u, \nabla \times v)_\Omega - \omega^2 \varepsilon\, (u, v)_\Omega\, .$$

RBM sets $W_{RB} = \mathrm{span}\{u_h(\mu_1), \ldots, u_h(\mu_N)\}$.
The final system is going from $\mathcal{N} \times \mathcal{N}$ to $N \times N$.
Difficulty: calculations of $a_h(u_h(\mu_i), u_h(\mu_j); \mu)$ and $f_h(u_h(\mu_i); \mu)$ $\forall \mu$ are still $\mathcal{N}-$dependent!
Solution: affine assumption

$$a(u, v; \mu) \equiv \sum_{q=1}^{Q_a} \Theta_a^q(\mu)\, a^q(u, v), \;\; f(v; \mu) \equiv \sum_{q=1}^{Q_f} \Theta_f^q(\mu)\, f^q(v).$$

Decomposing

$$a(u, v; \mu) = \kappa\, (\nabla u, \nabla v)_\Omega + c\, (u, v)_\Omega$$
$$a(u, v; \mu) = \nu^{-1} (\nabla \times u, \nabla \times v)_\Omega - \omega^2 \varepsilon\, (u, v)_\Omega\,.$$

Remedy for non-affinity/linearity: EIM, Empirical Interpolation.

Empirical Interpolation Method (EIM): Barrault et al. 2004; Maday et al. 2012, 2016

RBM (with EIM, if necessary) achieving $\mathcal{N}$-independence

$$a^{\mathcal{N}}(u^{\mathcal{N}}(\mu_i), u^{\mathcal{N}}(\mu_j); \mu) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu) a^q(u^{\mathcal{N}}(\mu_i), u^{\mathcal{N}}(\mu_j))$$

$$= \sum_{q=1}^{Q_a} \Theta_a^q(\mu) A_{ij}^q$$

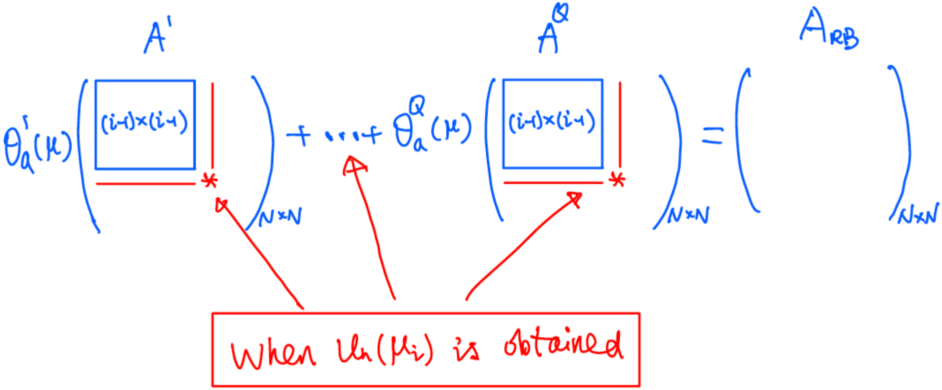
$$f^{\mathcal{N}}(u^{\mathcal{N}}(\mu_i);\, \mu) = \sum_{q=1}^{Q_f} \Theta_f^q(\mu) f^q(u^{\mathcal{N}}(\mu_i)) = \sum_{q=1}^{Q_f} \Theta_f^q(\mu)\, f_i^q$$

$$\text{with } A_{ij}^q := a^q(u^{\mathcal{N}}(\mu_i),\, u^{\mathcal{N}}(\mu_j)) \text{ and } f_i^q := f^q(u^{\mathcal{N}}(\mu_i)).$$

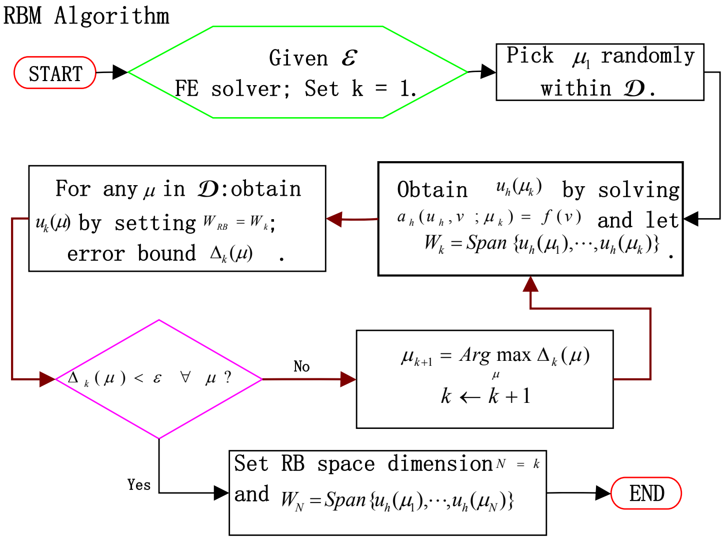
Key words: Preparation/Training - execution/testing, Offline - online

Offline-Online Procedure: Rozza et al. 2008

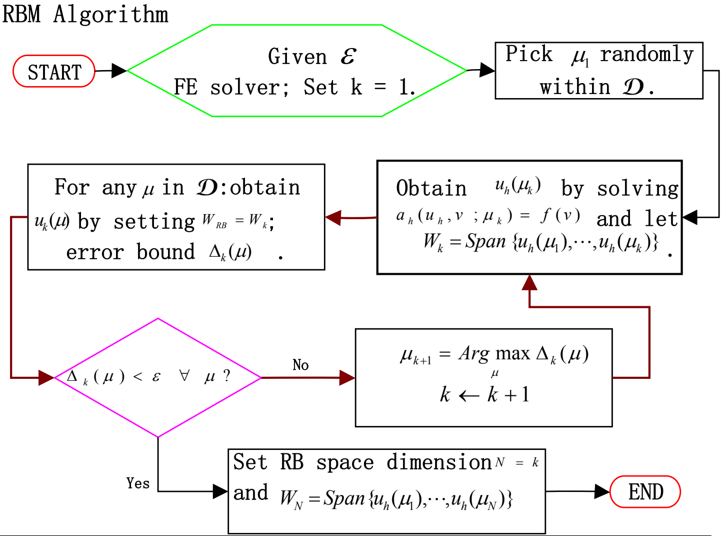
RBM: example for proceeding with care



Identifying $\{\mu_1, \cdots, \mu_N\}$: the greedy algorithm



Identifying $\{\mu_1, \cdots, \mu_N\}$: the greedy algorithm



Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaszczyk 2011; Buffa, Maday, Patera, Prudhomme, Turinici 2012

Maday, Patera, Turinici 2002; Rozza, Patera 2008; Chen, Gottlieb 2012; Hesthaven, Rozza, Stamm 2016

RBM: Algorithm breakup

Preparation Stage:

Solve $a_h(u_h(\mu), v; \mu) = f_h(v; \mu) \ \forall v \in X_h$ for $\mu \in \{\mu_1, \dots, \mu_N\}$ to obtain $u_h(\mu_1)$, ..., $u_h(\mu_N)$.

Evaluate $A^q_{ij} = a^q(u_h(\mu_i), \, u_h(\mu_j))$ and $f^q_i := f^q(u_h(\mu_i))$.

For any given μ:

Set $A = \sum_{q=1}^{Q_a} \Theta^q_a(\mu) A^q$ and $f = \sum_{q=1}^{Q_f} \Theta^q_f(\mu) f^q$.

Obtain the reduced basis solution by solving $A u_N(\mu) = f$.

What is left:

How to obtain the error bound $\Delta_N(\mu)$ for $\|u_N(\mu) - u_h(\mu)\|$?

A Key and messy component: *A Posteriori* Error Estimate

Defining $e(\mu) = u_h(\mu) - u^N(\mu)$, we have

$$a_h(e(\mu), v; \mu) = r(v; \mu) := f(v) - a_h(u^N(\mu), v; \mu).$$

Define an operator $T^\mu : X_h \to X_h$ as

$$(T^\mu w, v)_{X_h} = a_h(w, v; \mu), \ \forall v \in X_h.$$

[1] Huynh, Rozza, Sen, Patera 2007; C., Hesthaven, Maday, Rodriguez 2009; Huynh, Knezevic, C., Hesthaven, Patera 2010

A Key and messy component: *A Posteriori* Error Estimate

Defining $e(\mu) = u_h(\mu) - u^N(\mu)$, we have

$$a_h(e(\mu), v; \mu) = r(v; \mu) := f(v) - a_h(u^N(\mu), v; \mu).$$

Define an operator $T^\mu : X_h \to X_h$ as

$$(T^\mu w, v)_{X_h} = a_h(w, v; \mu), \ \forall v \in X_h.$$

It is easy to show that $\|T^\mu e(\mu)\|_{X_h} = \|r(\cdot; \mu)\|_{X_h'}$ and

$$\beta_h(\mu) \equiv \inf_{\omega \in X_h} \sup_{v \in X_h} \frac{a_h(\omega, v; \mu)}{\|\omega\|_{X_h} \|v\|_{X_h}} = \inf_{w \in X_h} \frac{\|T^\mu w\|_{X_h}}{\|w\|_{X_h}}.$$

Hence, $\|e(\mu)\|_{X_h} \leq \frac{\|T^\mu e(\mu)\|_{X_h}}{\beta_h(\mu)} = \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_h(\mu)} \leq \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_{\rm LB}(\mu)} := \Delta_N(\mu).$

[1] Huynh, Rozza, Sen, Patera 2007; C., Hesthaven, Maday, Rodriguez 2009; Huynh, Knezevic, C., Hesthaven, Patera 2010

A Key and messy component: *A Posteriori* Error Estimate

Defining $e(\mu) = u_h(\mu) - u^N(\mu)$, we have

$$a_h(e(\mu), v; \mu) = r(v; \mu) := f(v) - a_h(u^N(\mu), v; \mu).$$

Define an operator $T^\mu : X_h \to X_h$ as

$$(T^\mu w, v)_{X_h} = a_h(w, v; \mu), \ \forall v \in X_h.$$

It is easy to show that $\|T^\mu e(\mu)\|_{X_h} = \|r(\cdot;\mu)\|_{X_h'}$ and

$$\beta_h(\mu) \equiv \inf_{\omega \in X_h} \sup_{v \in X_h} \frac{a_h(\omega, v; \mu)}{\|\omega\|_{X_h} \|v\|_{X_h}} = \inf_{w \in X_h} \frac{\|T^\mu w\|_{X_h}}{\|w\|_{X_h}}.$$

Hence, $\|e(\mu)\|_{X_h} \le \frac{\|T^\mu e(\mu)\|_{X_h}}{\beta_h(\mu)} = \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_h(\mu)} \le \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_{\rm LB}(\mu)} := \Delta_N(\mu)$.

- $\bigstar_1$ Only $N-$dependent online evaluation of $\|r(\cdot;\mu)\|_{X_h'}$ for any μ!
- $\bigstar_2$ Successive Constraint Method (SCM) for efficient evaluation of $\beta_{\rm LB}(\mu)$.[1]

[1] Huynh, Rozza, Sen, Patera 2007; C., Hesthaven, Maday, Rodriguez 2009; Huynh, Knezevic, C., Hesthaven, Patera 2010

Summary

RBM	provides	
Accurate	$U^N \approx u_h(\boldsymbol{\mu})$	Approximation
Reliable	$\Delta_N \geq \|u_h - u^N\|_X$	Error Est.
Efficient	with $O(N \ll \mathcal{N})$ cost	Offline-Online
surrogates		Decomposition
with a small	N	Greedy

to solutions of **parameterized systems** for the **many-query, real-time** contexts.

Summary

RBM	provides	
Accurate	$U^N \approx u_h(\boldsymbol{\mu})$	Approximation
Reliable	$\Delta_N \geq \|u_h - u^N\|_X$	Error Est.
Efficient	with $O(N \ll \mathcal{N})$ cost	Offline-Online
surrogates		Decomposition
with a small	N	Greedy

to solutions of **parameterized systems** for the **many-query, real-time** contexts.

Other prominent features

A wide variety of parameters

Summary

RBM	provides	
Accurate	$U^N \approx u_h(\boldsymbol{\mu})$	Approximation
Reliable	$\Delta_N \geq \|u_h - u^N\|_X$	Error Est.
Efficient	with $O(N \ll \mathcal{N})$ cost	Offline-Online
surrogates		Decomposition
with a small	N	Greedy

to solutions of **parameterized systems** for the **many-query, real-time** contexts.

Other prominent features

A wide variety of parameters

Not restricted to truth solver being FEM or Galerkin type

Summary

RBM	provides	
Accurate	$U^N \approx u_h(\boldsymbol{\mu})$	Approximation
Reliable	$\Delta_N \geq \|u_h - u^N\|_X$	Error Est.
Efficient	with $O(N \ll \mathcal{N})$ cost	Offline-Online
surrogates		Decomposition
with a small	N	Greedy

to solutions of **parameterized systems** for the **many-query, real-time** contexts.

Other prominent features

A wide variety of parameters
Not restricted to truth solver being FEM or Galerkin type
Can be integrated with Domain Decomposition for powerful component synthesis - Reduced Basis Element Method (RBEM).

RBM philosophy

Systemize/Parameterize your (sequence of) (sub-)problems.

RBM philosophy

Systemize/Parameterize your (sequence of) (sub-)problems.

Pursue a surrogate.

RBM philosophy

Systemize/Parameterize your (sequence of) (sub-)problems.

Pursue a surrogate.

Be hand-waving in assessing the accuracy of your surrogate.

RBM philosophy

Systemize/Parameterize your (sequence of) (sub-)problems.

Pursue a surrogate.

Be hand-waving in assessing the accuracy of your surrogate.

Quickly examine all the problems.

RBM philosophy

Systemize/Parameterize your (sequence of) (sub-)problems.

Pursue a surrogate.

Be hand-waving in assessing the accuracy of your surrogate.

Quickly examine all the problems.

Iteration, iteration, iteration: trash in, treasure out.

RBM philosophy

Systemize/Parameterize your (sequence of) (sub-)problems.

Pursue a surrogate.

Be hand-waving in assessing the accuracy of your surrogate.

Quickly examine all the problems.

Iteration, iteration, iteration: trash in, treasure out.

Be greedy in refining your surrogate.

Time for ···


1958
University of Science and Technology of China
Lectures Series on High-Order Numerical Methods
2020

RBM: Approximation Theory and two key tools

Some approximation theory for RBM

Empirical interpolation method (EIM) for adaptive separation of variable

Successive constraint method (SCM) for squeezing parametric (extremal) eigenvalues

RBM theory: takeaways

★$_1$ Approximation theory for the solution manifold $\{u(\mu) : \mu \in \mathcal{D}\}$

★$_2$ The Kolmogorov N-width

$$d_N\left[u\left(\cdot;\mathcal{D}\right)\right] := \inf_{\substack{X_N \subset X^{\mathcal{N}} \\ \dim X_N = N}} \left[\operatorname{dist}\left(u\left(\cdot;\mathcal{D}\right), X_N\right) := \sup_{\mu\in\mathcal{D}} \inf_{v \in X_N} \|u(\cdot,\mu) - v\|_X\right]$$

RBM theory: takeaways

★$_1$ Approximation theory for the solution manifold $\{u(\mu) : \mu \in \mathcal{D}\}$

★$_2$ The Kolmogorov N-width

$$d_N\left[u\left(\cdot;\mathcal{D}\right)\right] := \inf_{\substack{X_N \subset X^{\mathcal{N}} \\ \dim X_N = N}} \left[\text{dist}\left(u\left(\cdot;\mathcal{D}\right), X_N\right) := \sup_{\mu\in\mathcal{D}} \inf_{v\in X_N} \|u(\cdot,\mu) - v\|_X\right]$$

★$_3$ A constructive (practically feasible) approximation algorithm, the greedy algorithm, to "realize" $d_N\left[u\left(\cdot;\mathcal{D}\right)\right]$.

RBM theory: takeaways

$\bigstar_1$ Approximation theory for the solution manifold $\{u(\mu) : \mu \in \mathcal{D}\}$

$\bigstar_2$ The Kolmogorov N-width

$$d_N\left[u\left(\cdot;\mathcal{D}\right)\right] := \inf_{\substack{X_N \subset X^{\mathcal{N}} \\ \dim X_N = N}} \left[\operatorname{dist}\left(u\left(\cdot;\mathcal{D}\right), X_N\right) := \sup_{\mu \in \mathcal{D}} \inf_{v \in X_N} \|u(\cdot,\mu) - v\|_X \right]$$

$\bigstar_3$ A constructive (practically feasible) approximation algorithm, the greedy algorithm, to "realize" $d_N\left[u\left(\cdot;\mathcal{D}\right)\right]$.

$\bigstar_4$ Is the greedy algorithm too greedy?

$\bigstar_5$ Holomorphism ($\mathcal{D} \longrightarrow u(\cdot,\mu)$) transfers fast decay of $d_N(\mathcal{D})$ to $d_N\left[u\left(\cdot;\mathcal{D}\right)\right]$.

Pinkus 1985 | Maday, Patera, Turinici 2004

Binev, Cohen, Dahmen, DeVore, Petrova, Wojtaszczyk 2011 | Haasdonk 2011

Cohen, DeVore, since 2014 | Bachmayr, Cohen 2016

Cea's Lemma

$$\|u_h(\boldsymbol{\mu}) - u^N(\boldsymbol{\mu})\|_X \leq \sqrt{\frac{M}{\alpha_{\mathrm{coer}}}} \inf_{v_N \in X_N} \|u_h(\boldsymbol{\mu}) - v_N\|_X$$

Y. Maday, A. T. Patera, and G. Turinici, 2002

$$a(w, v; \mu) \equiv a_0(w, v) + \mu a_1(w, v), \qquad \mu \in [0, \mu_{\max}]$$

a_0, a_1: continuous, symmetric and positive semi-definite.

a_0: coercive.

This means $0 \leq \frac{a_1(v,v)}{a_0(v,v)} \leq \gamma_1, \quad \forall v \in X, \ v \neq 0.$

Y. Maday, A. T. Patera, and G. Turinici, 2002

$$a(w, v; \mu) \equiv a_0(w, v) + \mu a_1(w, v), \qquad \mu \in [0, \mu_{\max}]$$

a_0, a_1: continuous, symmetric and positive semi-definite.

a_0: coercive.

This means $0 \leq \frac{a_1(v,v)}{a_0(v,v)} \leq \gamma_1, \quad \forall v \in X, \; v \neq 0.$

Theorem

For $N \geq N_{\rm crit} \equiv C \log(\gamma \mu_{\max} + 1)$ and $\forall \mu \in \mathcal{D}$,

$$|\, u_h(\mu) - u^N(\mu) \,|_{\mathcal{A}} \leq (1 + \mu_{\max}\gamma_1)^{1/2} |\, u_h(0) \,|_{\mathcal{A}} \exp\left\{ \frac{-\mathrm{N}}{2\mathrm{N}_{\rm crit}} \right\}.$$

Here, $|\cdot|_{\mathcal{A}} = a_0(\cdot, \cdot)^{1/2}$.

Buffa, Maday, Patera, Prud'homme, and Turinici, 2009

$$|a(u, v; \boldsymbol{\mu})| \leq M \|u\|_X \, \|v\|_X$$

$$a(u, u; \boldsymbol{\mu}) \geq \alpha_{\rm coer} \|u\|_X^2$$

$\alpha_{\rm coer}^{\rm LB} \leq \alpha_{\rm coer},$ the lower bound used in the error estimate.

Buffa, Maday, Patera, Prud'homme, and Turinici, 2009

$|a(u, v; \boldsymbol{\mu})| \leq M \|u\|_X \|v\|_X$

$a(u, u; \boldsymbol{\mu}) \geq \alpha_{\mathrm{coer}} \|u\|_X^2$

$\alpha_{\mathrm{coer}}^{\mathrm{LB}} \leq \alpha_{\mathrm{coer}}$, the lower bound used in the error estimate.

Theorem

If we have that

$$d_N \left[u(\cdot; \mathcal{D}), X^{\mathcal{N}} \right] \leq c e^{-\alpha N} \text{ with } \alpha > \log \left(1 + \gamma_{\mathrm{alg}} \sqrt{\frac{M}{\alpha_{\mathrm{coer}}}} \right)$$

then there exists $\beta > 0$ such that

$$\forall \boldsymbol{\mu} \in \mathcal{D}, \quad \|u_h(\boldsymbol{\mu}) - u^N(\boldsymbol{\mu})\|_X \leq C e^{-\beta N}$$

Buffa, Maday, Patera, Prud'homme, and Turinici, 2009

$|a(u, v; \boldsymbol{\mu})| \leq M\|u\|_X \|v\|_X$

$a(u, u; \boldsymbol{\mu}) \geq \alpha_{\text{coer}}\|u\|_X^2$

$\alpha_{\text{coer}}^{\text{LB}} \leq \alpha_{\text{coer}}$, the lower bound used in the error estimate.

Theorem

If we have that

$$d_N\left[u\left(\cdot;\mathcal{D}\right), X^{\mathcal{N}}\right] \leq ce^{-\alpha N} \text{ with } \alpha > \log\left(1 + \gamma_{\text{alg}}\sqrt{\frac{M}{\alpha_{\text{coer}}}}\right)$$

then there exists $\beta > 0$ such that

$$\forall \boldsymbol{\mu} \in \mathcal{D}, \quad \|u_h(\boldsymbol{\mu}) - u^N(\boldsymbol{\mu})\|_X \leq Ce^{-\beta N}$$

Here, $\gamma_{\text{alg}} = 1$ for strong greedy algorithm, and

$\gamma_{\text{alg}} = \dfrac{M}{\alpha_{\text{coer}}^{\text{LB}}}$ for week greedy algorithm.

Binev, Cohen, Dahmen, Devore, Petrova, and Wojtaszczyk, 2011: how about the actual distance?

V_N: the realized RB (N-dimensional) space.

$\sigma_N(u(\cdot;\mathcal{D})) = \mathrm{dist}(u(\cdot;\mathcal{D}), V_N)$.

The previous result has: $\sigma_N \leq CN2^N d_N$

Binev, Cohen, Dahmen, Devore, Petrova, and Wojtaszczyk, 2011: how about the actual distance?

V_N: the realized RB (N-dimensional) space.
$\sigma_N(u(\cdot;\mathcal{D})) = \mathrm{dist}(u(\cdot;\mathcal{D}), V_N)$.
The previous result has: $\sigma_N \leq CN2^N d_N$

Theorem

If $d_N \leq M \cdot N^{-\alpha}$, then $\sigma_N \leq C_\alpha M \cdot N^{-\alpha}$. Here $C = q^{\frac{1}{2}}(4q)^\alpha$ and $q = \lceil 2^{\alpha+1}\gamma^{-1} \rceil^2$.

$\gamma = \frac{c_r}{C_r}$ with $c_r r_N \leq \|u_h - u^N\| \leq C_r r_N$ where r_N is the error estimate.

Binev, Cohen, Dahmen, Devore, Petrova, and Wojtaszczyk, 2011: how about the actual distance?

V_N: the realized RB (N-dimensional) space.
$\sigma_N(u(\cdot;\mathcal{D})) = \mathrm{dist}(u(\cdot;\mathcal{D}), V_N)$.
The previous result has: $\sigma_N \leq CN2^N d_N$

Theorem

If $d_N \leq M \cdot N^{-\alpha}$, then $\sigma_N \leq C_\alpha M \cdot N^{-\alpha}$. Here $C = q^{\frac{1}{2}}(4q)^\alpha$ and $q = \lceil 2^{\alpha+1}\gamma^{-1}\rceil^2$.

$\gamma = \frac{c_r}{C_r}$ with $c_r r_N \leq \|u_h - u^N\| \leq C_r r_N$ where r_N is the error estimate.

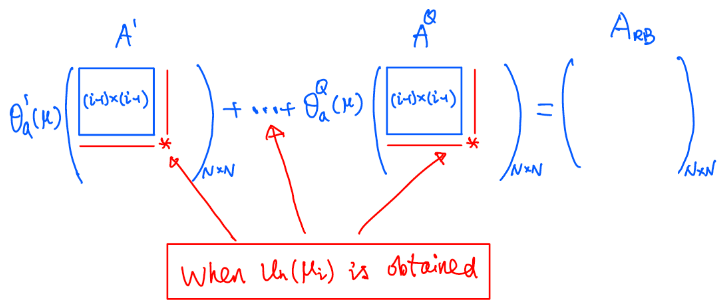
Theorem

If $d_N \leq M \cdot e^{-aN^\alpha}$, then $\sigma_N \leq CM \cdot e^{-cN^\beta}$. Here $\beta = \frac{\alpha}{\alpha+1}$, $C = \max\{e^{cN_0^\beta}, q^{\frac{1}{2}}\}$, $c = \min\{|\ln\theta|, (4q)^{-\alpha}a\}$, $q = \lceil 2\gamma^{-1}\theta^{-1}\rceil^2$, $N_0 = \lceil (8q)^{\alpha+1}\rceil$, any fixed $0 < \theta < 1$.

Why EIM: need of affinity

RBM achieving $\mathcal{N}$-independence

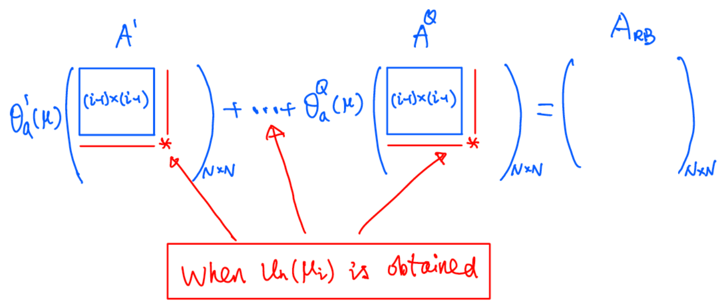
$$a^{\mathcal{N}}(u^{\mathcal{N}}(\mu_i), u^{\mathcal{N}}(\mu_j); \mu) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu) a^q(u^{\mathcal{N}}(\mu_i), u^{\mathcal{N}}(\mu_j)) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu) A_{ij}^q$$



Why EIM: need of affinity

RBM achieving $\mathcal{N}$-independence

$$a^{\mathcal{N}}(u^{\mathcal{N}}(\mu_i), u^{\mathcal{N}}(\mu_j); \mu) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu) a^q(u^{\mathcal{N}}(\mu_i), u^{\mathcal{N}}(\mu_j)) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu) A_{ij}^q$$



What to do when we don't have affinity e.g. $\kappa(x, \mu) = \exp^{x\mu}$?

A RBM-ish procedure producing $\{\mu^1, \ldots, \mu^k\}$ so that $\kappa(x, \mu)$ can be well approximated by $\sum_{q=1}^{Q_h} \theta_q(\mu)\kappa(x, \mu^k)$.

EIM for recovering affinity: examples

$g(x; k) \approx \sin(k\pi x), k \in \mathcal{D}, x \in \Omega$ (Incidence wave)

$$g(x; k) = \sum_{q=1}^{Q_h} \theta_q(k) g_q(x)$$

EIM for recovering affinity: examples

$g(x;k) \approx \sin(k\pi x), k \in \mathcal{D}, x \in \Omega$ (Incidence wave)

$$g(x;k) = \sum_{q=1}^{Q_h} \theta_q(k) g_q(x)$$

with $g_q(x) = \sin(k_q \pi x)$ for well-chosen $\{k_q\}_{q=1}^{Q_h}$.

EIM for recovering affinity: examples

$g(x;k) \approx \sin(k\pi x), k \in \mathcal{D}, x \in \Omega$ (Incidence wave)

$$g(x;k) = \sum_{q=1}^{Q_h} \theta_q(k) g_q(x)$$

with $g_q(x) = \sin(k_q \pi x)$ for well-chosen $\{k_q\}_{q=1}^{Q_h}$.

$h(y;s) = y^{1-2s}$, $s \in (0,1)$ $\quad y \in (0,\infty)$ (fractional Laplacian)

Severe anisotropy, singular dependence / degeneracy at $y = 0, \infty$.

EIM for recovering affinity: examples

$g(x;k) \approx \sin(k\pi x), k \in \mathcal{D}, x \in \Omega$ (Incidence wave)

$$g(x;k) = \sum_{q=1}^{Q_h} \theta_q(k) g_q(x)$$

with $g_q(x) = \sin(k_q \pi x)$ for well-chosen $\{k_q\}_{q=1}^{Q_h}$.

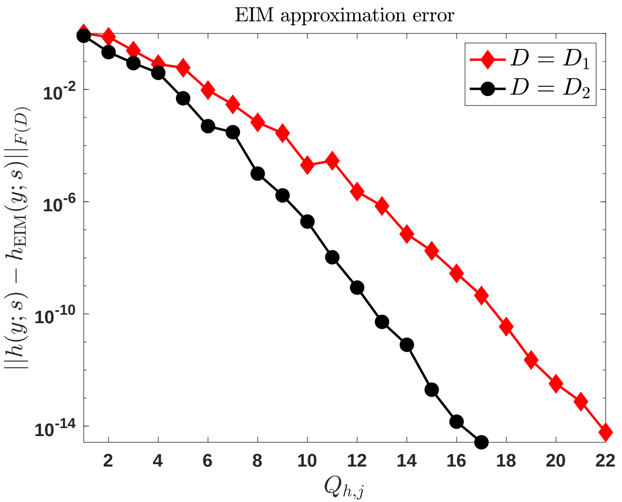
$h(y;s) = y^{1-2s}$, $s \in (0,1) \quad y \in (0,\infty)$ (fractional Laplacian)

Severe anisotropy, singular dependence / degeneracy at $y = 0, \infty$.

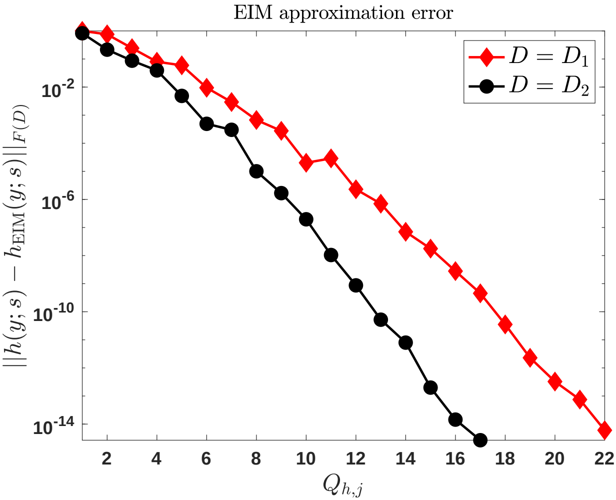
$$h(y;s) = y^{1-2s} \approx \sum_{j=1}^{Q_{h,1}} \theta_{j,1}(s) h_{q,1}(y), \quad s \in (0, \frac{1}{2}], \quad h_{q,1} = h(y; s_{q,1})$$

$$yh(y;s) = y^{2-2s} \approx \sum_{j=1}^{Q_{h,2}} \theta_{j,2}(s) h_{q,2}(y), \quad s \in (\frac{1}{2}, 1), \quad h_{q,2} = yh(y; s_{q,2})$$

EIM result for $h(y; s) = y^{1-2s}$



EIM result for $h(y; s) = y^{1-2s}$



17 or 22 is awesome for this singular / degenerate (1D) case.
Warning: it can be hundreds for higher-D or geometrical parameters.

Why SCM: Recall the *A Posteriori* Error Estimate

Defining $e(\mu) = u_h(\mu) - u^N(\mu)$, we have

$$a_h(e(\mu), v; \mu) = r(v; \mu) := f(v) - a_h(u^N(\mu), v; \mu).$$

Define an operator $T^\mu : X_h \to X_h$ as

$$(T^\mu w, v)_{X_h} = a_h(w, v; \mu), \ \forall v \in X_h.$$

Why SCM: Recall the *A Posteriori* Error Estimate

Defining $e(\mu) = u_h(\mu) - u^N(\mu)$, we have

$$a_h(e(\mu), v; \mu) = r(v; \mu) := f(v) - a_h(u^N(\mu), v; \mu).$$

Define an operator $T^\mu : X_h \to X_h$ as

$$(T^\mu w, v)_{X_h} = a_h(w, v; \mu), \ \forall v \in X_h.$$

It is easy to show that $\|T^\mu e(\mu)\|_{X_h} = \|r(\cdot; \mu)\|_{X_h'}$ and

$$\beta_h(\mu) \equiv \inf_{\omega \in X_h} \sup_{v \in X_h} \frac{a_h(\omega, v; \mu)}{\|\omega\|_{X_h} \|v\|_{X_h}} = \inf_{w \in X_h} \frac{\|T^\mu w\|_{X_h}}{\|w\|_{X_h}}.$$

Hence, $\|e(\mu)\|_{X_h} \le \frac{\|T^\mu e(\mu)\|_{X_h}}{\beta_h(\mu)} = \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_h(\mu)} \le \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_{\mathrm{LB}}(\mu)} := \Delta_N(\mu)$.

Why SCM: Recall the *A Posteriori* Error Estimate

Defining $e(\mu) = u_h(\mu) - u^N(\mu)$, we have

$$a_h(e(\mu), v; \mu) = r(v; \mu) := f(v) - a_h(u^N(\mu), v; \mu).$$

Define an operator $T^\mu : X_h \to X_h$ as

$$(T^\mu w, v)_{X_h} = a_h(w, v; \mu), \ \forall v \in X_h.$$

It is easy to show that $\|T^\mu e(\mu)\|_{X_h} = \|r(\cdot;\mu)\|_{X_h'}$ and

$$\beta_h(\mu) \equiv \inf_{\omega \in X_h} \sup_{v \in X_h} \frac{a_h(\omega, v; \mu)}{\|\omega\|_{X_h} \|v\|_{X_h}} = \inf_{w \in X_h} \frac{\|T^\mu w\|_{X_h}}{\|w\|_{X_h}}.$$

Hence, $\|e(\mu)\|_{X_h} \le \frac{\|T^\mu e(\mu)\|_{X_h}}{\beta_h(\mu)} = \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_h(\mu)} \le \frac{\|r(\cdot;\mu)\|_{X_h'}}{\beta_{\mathrm{LB}}(\mu)} := \Delta_N(\mu)$.

- $\bigstar_1$ Only $N-$dependent online evaluation of $\|r(\cdot;\mu)\|_{X_h'}$ for any μ!
- $\bigstar_2$ Successive Constraint Method (SCM) for efficient evaluation of $\beta_{\mathrm{LB}}(\mu)$.

Goal of SCM

To obtain a (not-so-pessimistic) $\beta_{LB}(\mu)$ extremely fast, by drastically decreasing the number of necessary eigen-solves, in a RB-fashion!

Original: Huynh, Rozza, Sen, Patera 2007

Improved: C., Hesthaven, Maday, Rodriguez 2009

Natural norm SCM: Huynh, Knezevic, C., Hesthaven, Patera 2010

Certified natural norm SCM: C., 2016

Successive Constraint Method

Objective: Given μ, we need to find a lower bound for $\inf_{u \in X_h} \frac{a_h(u, u; \mu)}{\|u\|_{X_h}^2}$.

Setting $y_q(u) = \frac{a_h^q(u,u)}{\|u\|_{X_h}^2}$, we have

$$\alpha_h(\mu) = \inf_{u \in X_h} \sum_{q=1}^{Q} \Theta^q(\mu) \frac{a_h^q(u, u)}{\|u\|_{X_h}^2} = \inf_{u \in X_h} \sum_{q=1}^{Q} \Theta^q(\mu) y_q(u).$$

Successive Constraint Method

Objective: Given μ, we need to find a lower bound for $\inf_{u \in X_h} \frac{a_h(u,u;\mu)}{\|u\|^2_{X_h}}$.

Setting $y_q(u) = \frac{a^q_h(u,u)}{\|u\|^2_{X_h}}$, we have

$$\alpha_h(\mu) = \inf_{u \in X_h} \sum_{q=1}^{Q} \Theta^q(\mu) \frac{a^q_h(u,u)}{\|u\|^2_{X_h}} = \inf_{u \in X_h} \sum_{q=1}^{Q} \Theta^q(\mu) y_q(u).$$

Realizing that $(y_1(u), \ldots, y_Q(u))$ belongs to

$$\mathcal{Y} \equiv \left\{ y = (y_1, \ldots, y_Q) \in \mathbb{R}^Q \mid \exists\ u \in X_h \text{ s.}t.\ y_q = y_q(u),\ 1 \le q \le Q \right\},$$

we reinterpret $\alpha_h(\mu)$ as

$$\alpha_h(\mu) = \inf_{y\ \in \mathcal{Y}} \mathcal{J}(\mu; y) \ \text{ where } \ \mathcal{J}(\mu; y) = \sum_{q=1}^{Q} \Theta^q(\mu) y_q.$$

Successive Constraint Method — How to quantify $\mathcal{Y}$?

It is a subset of $\mathcal{B}_Q \equiv \Pi_{q=1}^{Q}[\sigma_q^-, \sigma_q^+] \subset \mathbb{R}^Q$ if we define

$$\sigma_q^- \equiv \inf_{u \in X_h} y_q(u), \quad \sigma_q^+ \equiv \sup_{u \in X_h} y_q(u), \quad 1 \leq q \leq Q.$$

Successive Constraint Method — How to quantify $\mathcal{Y}$?

It is a subset of $\mathcal{B}_Q \equiv \Pi_{q=1}^{Q}[\sigma_q^-, \sigma_q^+] \subset \mathbb{R}^Q$ if we define

$$\sigma_q^- \equiv \inf_{u \in X_h} y_q(u), \quad \sigma_q^+ \equiv \sup_{u \in X_h} y_q(u), \quad 1 \le q \le Q.$$

For a given $C_K = \{w_1, \ldots, w_K\}$, there exist

$$\mathcal{Y}_{UB}(C_K) \equiv \{y^*(w_k), 1 \le k \le K\} \text{ for } y^*(\mu) \equiv \text{argmin}_{y \in \mathcal{Y}} \mathcal{J}(\mu; y),$$

$$\mathcal{Y}_{LB}(\mu; C_K) \equiv \left\{ y \in \mathcal{B}_Q \mid \sum_{q=1}^{Q} \Theta^q(\mu') y_q \ge \alpha_h(\mu'), \text{ for some } \mu'; \right.$$
$$\left. \sum_{q=1}^{Q} \Theta^q(\mu') y_q \ge 0, \text{ for some other } \mu' \right\},$$

such that $\mathcal{Y}_{UB}(C_K) \subset \mathcal{Y} \subset \mathcal{Y}_{LB}(C_K)$.

Successive Constraint Method – a greedy algorithm!

We have, for $\alpha_h(\mu)$, a *lower bound* and a *upper bound*:

$$\alpha_{LB}(\mu; C_K) = \inf_{y \in \mathcal{Y}_{LB}(\mu; C_K)} \mathcal{J}(\mu; y)$$

$$\alpha_{UB}(\mu; C_K) = \inf_{y \in \mathcal{Y}_{UB}(C_K)} \mathcal{J}(\mu; y).$$

(1.) Set $K = 1$ and choose $C_1 = \{w_1\}$ arbitrarily.

(2.) Find $w_{K+1} = \mathrm{argmax}_{\mu \in \Xi} \frac{\alpha_{UB}(\mu; C_K) - \alpha_{LB}(\mu; C_K)}{\alpha_{UB}(\mu; C_K)}$.

(3.) Update $C_{K+1} = C_K \cup \{w_{K+1}\}$.

(4.) Repeat (2) and (3) until

$$\max_{\mu \in \Xi} \frac{\alpha_{UB}(\mu; C_{K_{max}}) - \alpha_{LB}(\mu; C_{K_{max}})}{\alpha_{UB}(\mu; C_{K_{max}})} \leq \epsilon_\alpha.$$

Successive Constraint Method – one key improvement

Replace $\mathcal{Y}_{LB}(\mu; C_K)$ by

$$\mathcal{Y}_{LB}(\mu; C_K) \equiv \left\{ y \in \mathcal{B}_Q \mid \sum_{q=1}^{Q} \Theta^q(\mu') y_q \geq \alpha_h(\mu'), \text{ for some } \mu'; \right.$$

$$\left. \sum_{q=1}^{Q} \Theta^q(\mu') y_q \geq \alpha_{LB}(\mu', C_{K-1}), \forall \text{ for some other } \mu' \right\}.$$

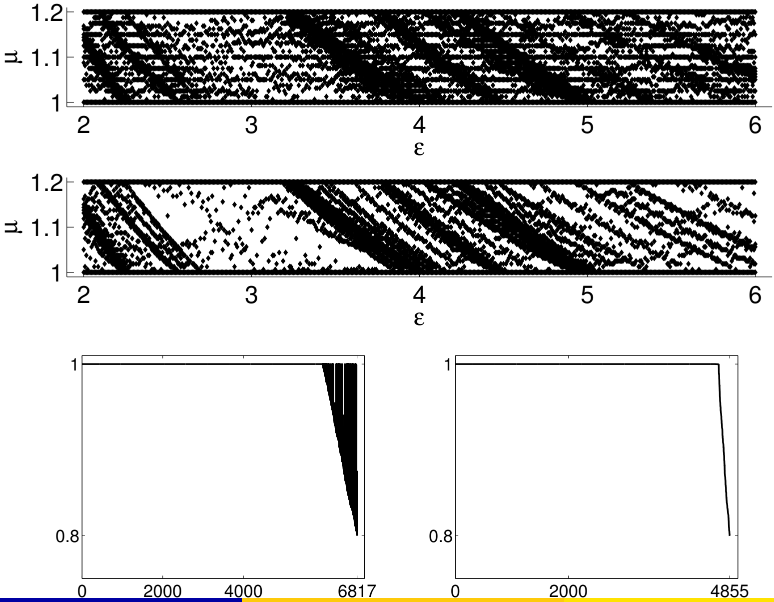
(1.) $\alpha_{LB}(\mu, C_K)$ is nondecreasing.

(2.) $\alpha_{UB}(\mu, C_K)$ is nonincreasing.

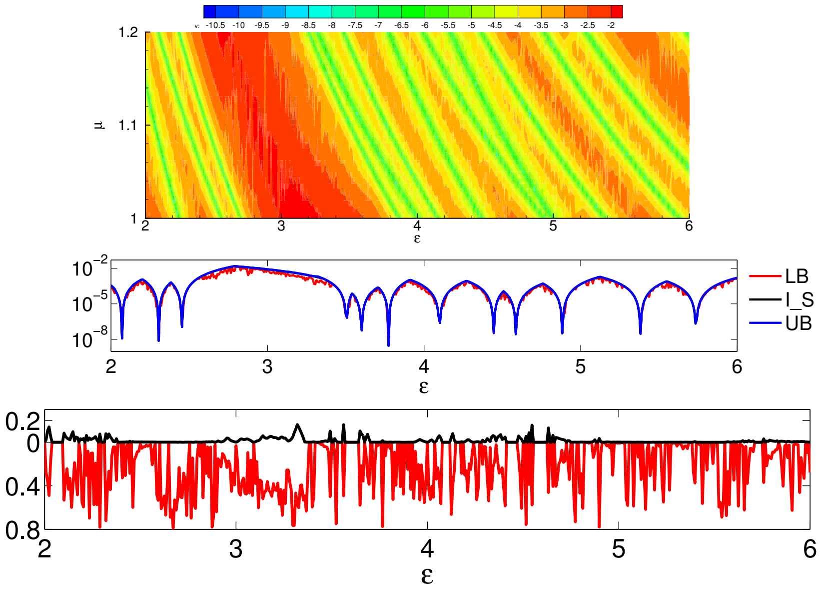
(3.) $\frac{\alpha_{UB}(\mu, C_K) - \alpha_{LB}(\mu, C_K)}{\alpha_{UB}(\mu, C_K)}$ is nonincreasing.

Output: $(\alpha_{LB}(\mu, C_K) \uparrow)$ and $(\alpha_{UB}(\mu, C_K) \downarrow)$

Successive Constraint Method – Comparison



Successive Constraint Method – Accuracy



Time for · · ·


1958
University of Science and Technology of China
Lectures Series on High-Order Numerical Methods
2020

Outline

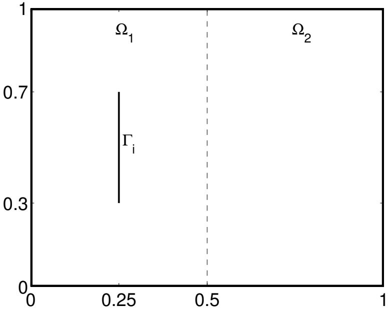
- Three ancient examples in electromagnetics
- A list of recent works
- Two particular challenges: error estimation and efficiency degradation due to EIM

  L1-RBM and solar cell simulation reduction
  A new RBM for (non-parametrized) stochastic PDE
  Reduced Collocation Method (RCM), Reduced Over Collocation (ROC)

Ancient RBM example I: An electromagnetic cavity [2]

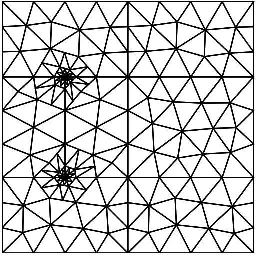
$$\begin{cases} -\epsilon\omega^2 E_x + \frac{1}{\mu}\frac{\partial}{\partial y}\left(\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y}\right) = i\omega J_x \\ -\epsilon\omega^2 E_y - \frac{1}{\mu}\frac{\partial}{\partial x}\left(\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y}\right) = i\omega J_y \end{cases}$$

$$E_x\,\hat{n}_y - E_y\,\hat{n}_x = 0 \text{ on } \partial\Omega.\ \ J_x = 0,\ J_y = \cos(\omega(y - \tfrac{1}{2}))\delta_{\Gamma_i}$$
$$\epsilon|_{\Omega_i} = \epsilon_i,\ \mu|_{\Omega_i} = \mu_i \text{ for } i = 1, 2$$



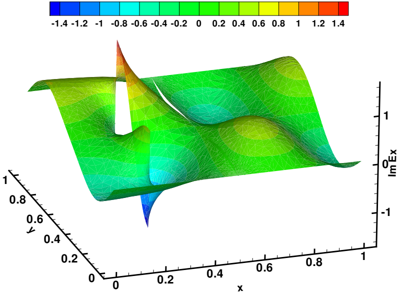
[2] C., Hesthaven, Maday, Rodriguez 2010

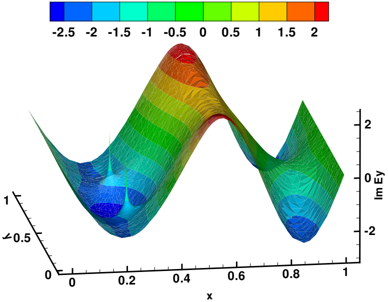
Ancient RBM example I: An electromagnetic cavity

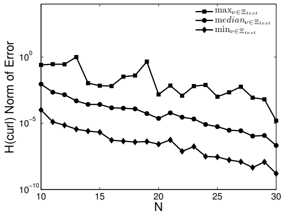


Numbers for a traditional method:

- ★$_1$ Number of elements: 310
- ★$_2$ Number of degrees of freedom 4650
- ★$_3$ Size of the linear system 9300×9300
- ★$_4$ Time for each solve 52 sec



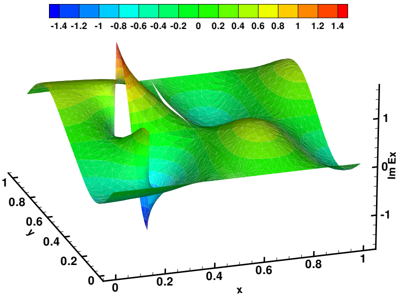


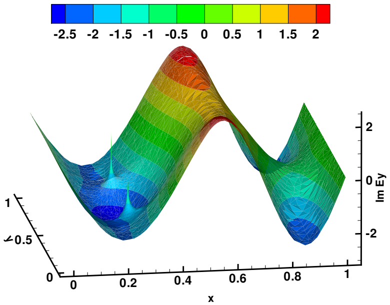


Numbers for RBM:

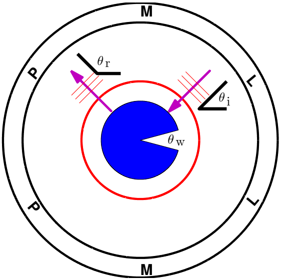
- ★$_1$ Number of degrees of freedom 30
- ★$_2$ Size of the linear system 60×60
- ★$_3$ Time for each solve 0.032 sec
- ★$_4$ Amount of saving 99.94%

Price to pay: a one-time preparation stage taking 87×30 sec





Ancient RBM example II: Where is the Pacman? [3]



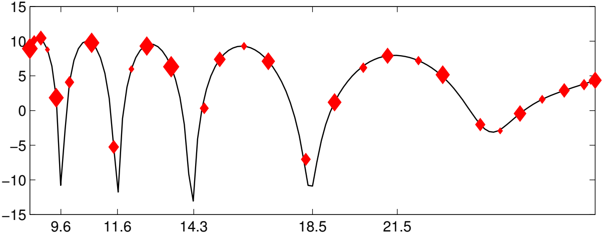
$$
\begin{aligned}
a_h(u_h^{Sca}, v;\ \mu) &= -a_h(u_h^{Inc}, v;\ \mu) \\
J(\omega, \theta_W, \theta_i) &= n \times H^{Sca}, \\
M(\omega, \theta_W, \theta_i) &= -n \times E^{Sca}, \\
RCS(\omega, \theta_W, \theta_i, \theta_r) &= 2\pi \frac{|F(\omega, \theta_W, \theta_i, \theta_r)|^2}{|E^{Inc}|^2}.
\end{aligned}
$$

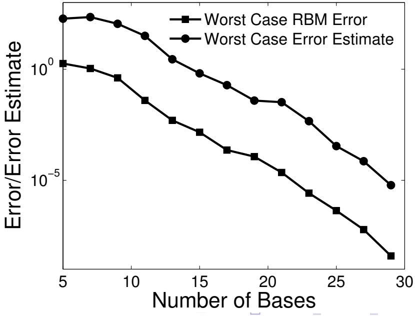
with

$$
F(\omega, \theta_W, \theta_i, \theta_r) = \frac{\sqrt{e^{i\frac{\pi}{4}}}}{\sqrt{8\pi\omega}} \oint \left(-\omega \hat{z} \cdot J - \omega \hat{z} \times M \cdot \hat{r}\right) e^{i\omega \hat{r} \cdot \bar{r}'}\, dC,
$$

[3] C., Hesthaven, Maday, Rodriguez, Zhu 2012

Ancient RBM example II: Where is the Pacman?





Ancient RBM example III: RBEM for Waveguide [4]

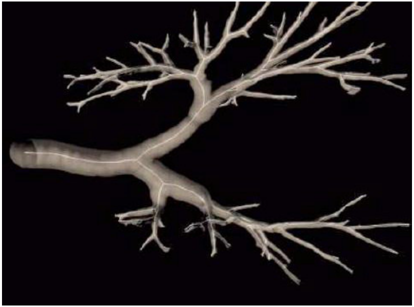
To solve problems on a (complicated but) self-similar domain.

RBEM $\simeq$ RBM $\bigoplus$ Domain Decomposition

Introduced by Yvon Maday and Einar Rønquist in 2002.

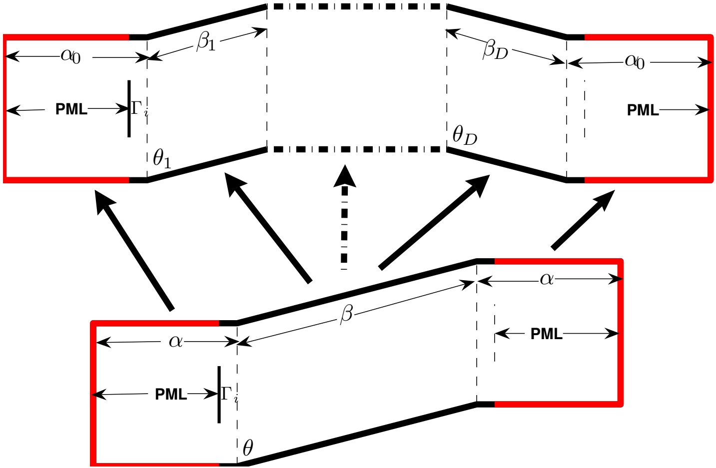
Used for heat transfer and fluid flow.

We focus on waveguide-design-inspired EM problems.



[4] C., Hesthaven, Maday 2011

RBEM Idea



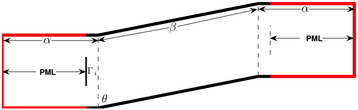
Arrows: mapping of functions on corresponding parallelograms.

Problem setup

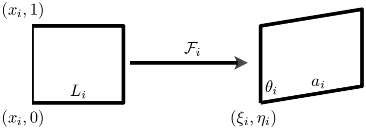
We want to solve

$$\begin{cases} -\epsilon\omega^2 \widehat{E}_\xi + \frac{1}{\nu}\frac{\partial}{\partial\eta}\left(\frac{\partial \widehat{E}_\eta}{\partial\xi} - \frac{\partial \widehat{E}_\xi}{\partial\eta}\right) = i\omega J_\xi, \\ -\epsilon\omega^2 \widehat{E}_\eta - \frac{1}{\nu}\frac{\partial}{\partial\xi}\left(\frac{\partial \widehat{E}_\eta}{\partial\xi} - \frac{\partial \widehat{E}_\xi}{\partial\eta}\right) = i\omega J_\eta, \end{cases}$$

on



. With



,

we apply the following piece-wise Piola transform:

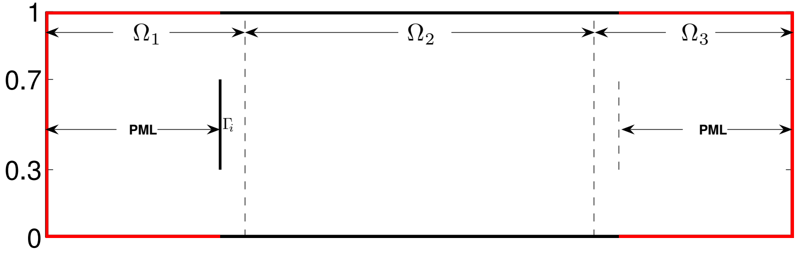
$$\begin{pmatrix} \widehat{E}_\eta \\ -\widehat{E}_\xi \end{pmatrix} = \mathcal{P}_i\left(\begin{pmatrix} E_y \\ -E_x \end{pmatrix}\right) := \frac{1}{|J\mathcal{F}_i|} J\mathcal{F}_i \begin{pmatrix} E_y \\ -E_x \end{pmatrix},$$

Seamless RBEM: C., Hesthaven, Maday, 2011

Problem setup

As a result, we solve the following parameter-dependent problem,

$$\begin{cases} (i\omega\nu + \frac{\mu\sigma}{\epsilon})H_z + \frac{L}{a\sin\theta}\left(\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y}\right) = 0, \\ i\epsilon\omega\left(\frac{L}{a\sin\theta}E_x - \cot\theta(1 - \frac{i\sigma}{\epsilon\omega})E_y\right) - \frac{1}{\nu}(1 - \frac{i\sigma}{\epsilon\omega})\frac{\partial H_z}{\partial y} = (1 - \frac{i\sigma}{\epsilon\omega})J_x, \\ i\epsilon\omega(1 - \frac{i\sigma}{\epsilon\omega})E_y + \frac{1}{\nu}\left(\frac{L}{a\sin\theta}\frac{\partial H_z}{\partial x} - (\cot\theta)\frac{\partial H_z}{\partial y}\right) = J_y. \end{cases}$$

on 

.

RBEM result: A (very) primitive Waveguide

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ   Jiang, C., Narayan, 2016

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ   Jiang, C., Narayan, 2016

Offline-speeding up   Jiang, C., Narayan, 2017

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ   Jiang, C., Narayan, 2016

Offline-speeding up   Jiang, C., Narayan, 2017

Truth solver being radial basis function: C., Gottlieb, Heryudono, Narayan, 2016

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ   Jiang, C., Narayan, 2016

Offline-speeding up   Jiang, C., Narayan, 2017

Truth solver being radial basis function: C., Gottlieb, Heryudono, Narayan, 2016

Stable RBM, L1-RBM   Jiang, C., Narayan, 2019

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ   Jiang, C., Narayan, 2016

Offline-speeding up   Jiang, C., Narayan, 2017

Truth solver being radial basis function: C., Gottlieb, Heryudono, Narayan, 2016

Stable RBM, L1-RBM   Jiang, C., Narayan, 2019

Solar-Cell simulation   C., Monk

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ Jiang, C., Narayan, 2016

Offline-speeding up Jiang, C., Narayan, 2017

Truth solver being radial basis function: C., Gottlieb, Heryudono, Narayan, 2016

Stable RBM, L1-RBM Jiang, C., Narayan, 2019

Solar-Cell simulation C., Monk

Fractional Laplacian Antil, C., Narayan, 2019

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ   Jiang, C., Narayan, 2016

Offline-speeding up   Jiang, C., Narayan, 2017

Truth solver being radial basis function: C., Gottlieb, Heryudono, Narayan, 2016

Stable RBM, L1-RBM   Jiang, C., Narayan, 2019

Solar-Cell simulation   C., Monk

Fractional Laplacian   Antil, C., Narayan, 2019

"Keep-or-toss" greedier algorithm stochastic PDE with "arbitrary" noise   Liu, Chen, C., Shu, 2019

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ Jiang, C., Narayan, 2016

Offline-speeding up Jiang, C., Narayan, 2017

Truth solver being radial basis function: C., Gottlieb, Heryudono, Narayan, 2016

Stable RBM, L1-RBM Jiang, C., Narayan, 2019

Solar-Cell simulation C., Monk

Fractional Laplacian Antil, C., Narayan, 2019

"Keep-or-toss" greedier algorithm stochastic PDE with "arbitrary" noise Liu, Chen, C., Shu, 2019

Collocation, Over Collocation C., Gottlieb 2013; C., Gottlieb, Maday 2014; Ji, C., Xu, 2019; C. Gottlieb, Ji, Maday, Narayan, Xu

Some more recent interesting works of ours

Reduced Basis Decomposition (Interpolatory Decomposition), C., 2015.

RB-enhanced gPC for UQ   Jiang, C., Narayan, 2016

Offline-speeding up   Jiang, C., Narayan, 2017

Truth solver being radial basis function: C., Gottlieb, Heryudono, Narayan, 2016

Stable RBM, L1-RBM   Jiang, C., Narayan, 2019

Solar-Cell simulation   C., Monk

Fractional Laplacian   Antil, C., Narayan, 2019

"Keep-or-toss" greedier algorithm stochastic PDE with "arbitrary" noise   Liu, Chen, C., Shu, 2019

Collocation, Over Collocation   C., Gottlieb 2013; C., Gottlieb, Maday 2014; Ji, C., Xu, 2019; C. Gottlieb, Ji, Maday, Narayan, Xu

RB-inspired fast iterative solver "beating" multigrid   Nguyen, C.

Challenge 1: *A Posteriori* Error Estimate is key but messy

Defining $e(\mu) = u_h(\mu) - u^N(\mu)$, we have

$$a_h(e(\mu), v; \mu) = r(v; \mu) := f(v) - a_h(u^N(\mu), v; \mu).$$

Define an operator $T^\mu : X_h \to X_h$ as

$$(T^\mu w, v)_{X_h} = a_h(w, v; \mu), \ \forall v \in X_h.$$

It is easy to show that $\| T^\mu e(\mu) \|_{X_h} = \| r(\cdot; \mu) \|_{X'_h}$ and

$$\beta_h(\mu) \equiv \inf_{\omega \in X_h} \sup_{v \in X_h} \frac{a_h(\omega, v; \mu)}{\|\omega\|_{X_h} \|v\|_{X_h}} = \inf_{w \in X_h} \frac{\| T^\mu w \|_{X_h}}{\|w\|_{X_h}}.$$

Hence, $\|e(\mu)\|_{X_h} \le \frac{\| T^\mu e(\mu) \|_{X_h}}{\beta_h(\mu)} = \frac{\|r(\cdot;\mu)\|_{X'_h}}{\beta_h(\mu)} \le \frac{\|r(\cdot;\mu)\|_{X'_h}}{\beta_{\mathrm{LB}}(\mu)} := \Delta_N(\mu)$.

$\bigstar_1$ Only $N-$dependent online evaluation of $\|r(\cdot; \mu)\|_{X'_h}$ for any μ!

$\bigstar_2$ Successive Constraint Method for efficient evaluation of $\beta_{\mathrm{LB}}(\mu)$.[5]

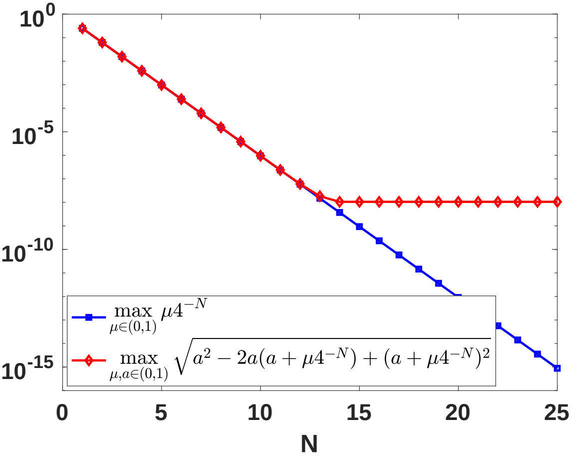
[5] Huynh, Rozza, Sen, Patera 2007; C., Hesthaven, Maday, Rodriguez 2009; Huynh, Knezevic, C., Hesthaven, Patera 2010

Accurate (and efficient) evaluation of $\|r(\cdot;\mu)\|_{X_h'}$ is tricky (and resp. messy)

$$|a-b| \stackrel{?}{=} \sqrt{a^2 - 2ab + b^2}$$

Accurate (and efficient) evaluation of $\|r(\cdot;\mu)\|_{X_h'}$ is tricky (and resp. messy)

$$|a-b| \stackrel{?}{=} \sqrt{a^2-2ab+b^2}$$



Challenge 2: EIM is powerful but leads to more terms

EIM for non-affine operator, as an example:

$$\mathbb{L}(\mu) = \sum_{q=1}^{Q_a} a_q^{\mathbb{L}}(x,\mu)\mathbb{L}_q \quad \bigoplus \quad a_q^{\mathbb{L}}(x,\mu) = \sum_{m=1}^{M_q} \phi_m^q(\mu) a_{aff,m}^q(x)$$

The online efficiency is, at best, inversely proportional to $\sum_{q=1}^{Q_a} M_q$.

Response to challenge 1

L1-RBM: residual-free replacement of the error estimate $\Delta_N(\mu)$

L1-RBM: residual-free replacement of $\Delta_N(\mu)$

$$\widetilde{\Delta}_N(\boldsymbol{\mu}) = \left(\sum_{m=1}^{N} |c_m(\boldsymbol{\mu})| \right), \quad \boldsymbol{\mu}^{N+1} = \operatorname{argmax} \widetilde{\Delta}_N(\boldsymbol{\mu}).$$

Lemma (C., Jiang, Narayan 2018)

Let $e_N(\boldsymbol{\mu}) = \left\| u^N(\boldsymbol{\mu}) - u^{\mathcal{N}}(\boldsymbol{\mu}) \right\|_{X^{\mathcal{N}}}$ be the RBM error committed at parameter value $\boldsymbol{\mu}$. Then

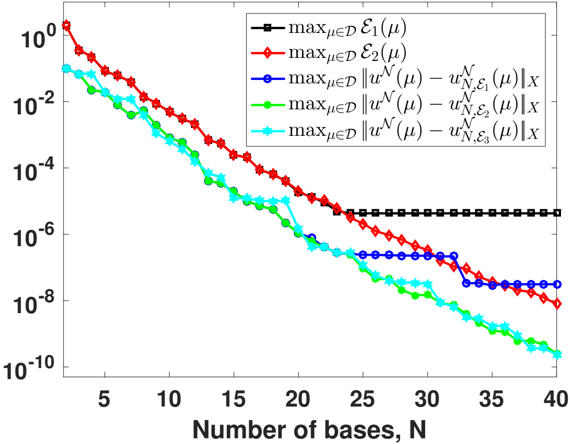
$$e_N(\boldsymbol{\mu}) \le \left(1 + \widetilde{\Delta}_N(\boldsymbol{\mu}) \right) \epsilon_N(u^{\mathcal{N}}),$$

where ϵ_N is the $\boldsymbol{\mu}$-independent quantity,

$$\epsilon_N(u^{\mathcal{N}}) := \inf_{v \in U_N} \left\| u^{\mathcal{N}} - v \right\|_{L^\infty(\mathcal{D}, X^{\mathcal{N}})}$$

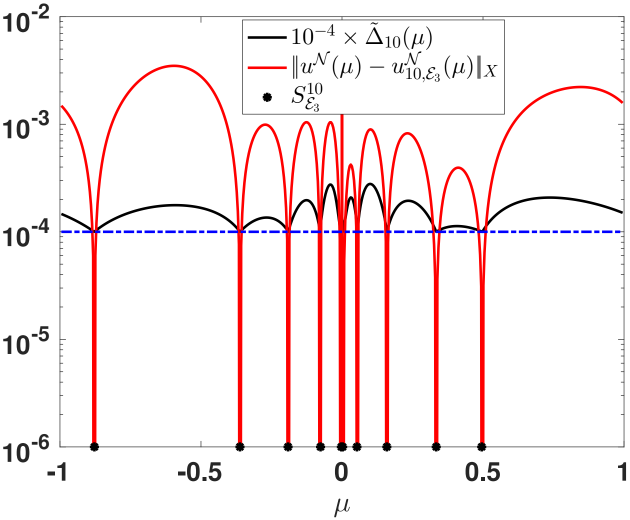
RBM error est.: stagnation ($\mathcal{E}_1$) and its resolutions (a robust $\|r(\cdot;\mu)\|_{X_h'}$ evaluation $\mathcal{E}_2$, replacement by $\widetilde{\Delta}_N(\mu)$ $\mathcal{E}_3$)

$$-u_{xx} - \mu_1 u_{yy} - \mu_2 u = -10\sin(8x(y-1)) \quad \text{on} \quad \Omega.$$



L1-RBM: handling parametric discontinuity

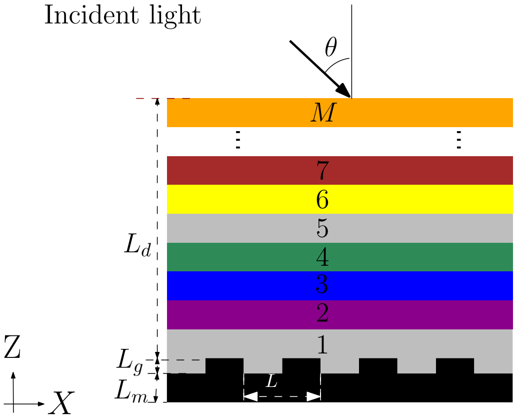
$$(1+\ell(\mu)x)u_{xx}+u_{yy} = e^{4xy} \quad \text{on} \quad \Omega, \ell(\mu) = \sin\left((\mu - \text{sign}(\mu))\frac{\pi}{2}\right), \quad \mu \in \mathcal{D}$$



Time for · · ·

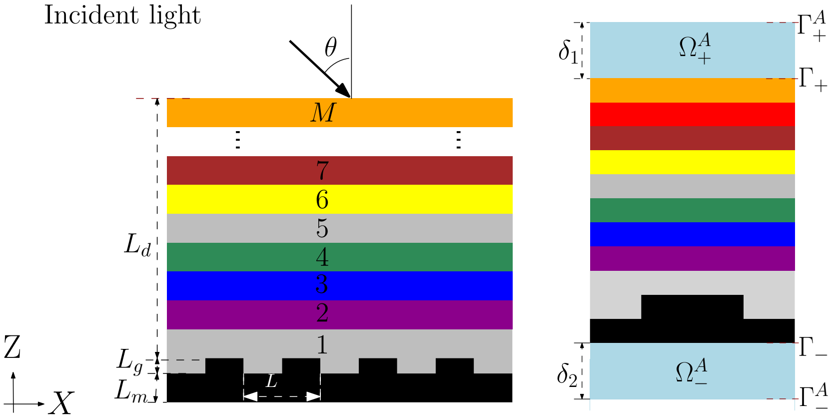

1958
University of Science and Technology of China
Lectures Series on High-Order Numerical Methods
2020

L1-RBM for Solar cell simulation, setup [6]



[6] C., Monk, Solano, Work in Progress

L1-RBM for Solar cell simulation, setup [6]



[6] C., Monk, Solano, Work in Progress

L1-RBM for Solar cell simulation, setup

$$
\begin{aligned}
\nabla \cdot (A \nabla u) + k^2 n u &= 0 && \text{in } \Omega \\
u &= u^i + u^s && \text{in } \Omega \\
\text{Quasi-periodicity: } \; u(L, z) &= \exp(i\alpha L) u(0, z) && \text{for } z \in \mathbb{R} \\
A \frac{\partial u}{\partial x}(L, z) &= \exp(i\alpha L) A \frac{\partial u}{\partial x}(0, z) && \text{for } z \in \mathbb{R}.
\end{aligned}
$$

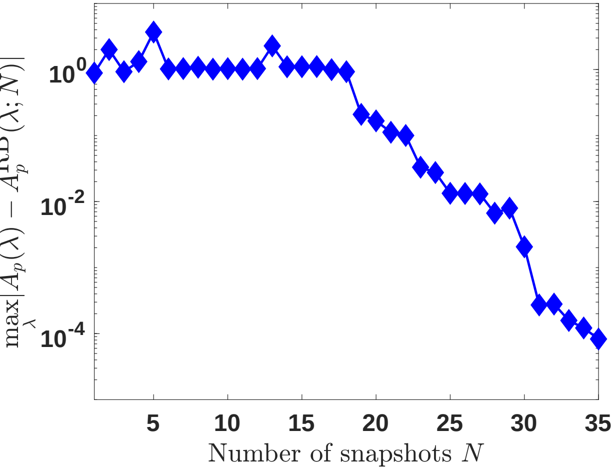
L1-RBM for Solar cell simulation, setup

$$\begin{aligned} \nabla \cdot (A \nabla u) + k^2 n u &= 0 && \text{in } \Omega \\ u &= u^i + u^s && \text{in } \Omega \\ \text{Quasi-periodicity: } u(L, z) &= \exp(i\alpha L) u(0, z) && \text{for } z \in \mathbb{R} \\ A \frac{\partial u}{\partial x}(L, z) &= \exp(i\alpha L) A \frac{\partial u}{\partial x}(0, z) && \text{for } z \in \mathbb{R}. \end{aligned}$$

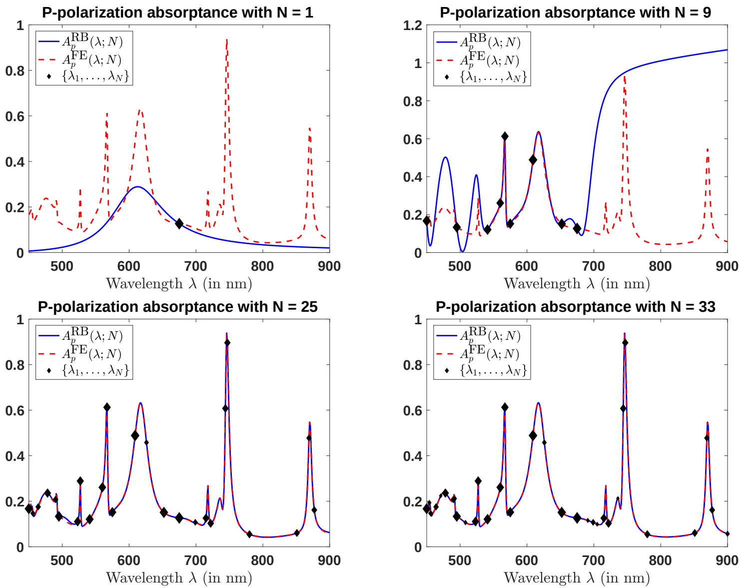
$$\text{TM: } A = \frac{1}{\epsilon_r(\lambda)} \text{ and } n = 1$$
$$u_{\text{inc}}(x, z) = \exp(ik(d_1 x + d_2 z))$$
$$\alpha := k d_1 = k \sin\theta$$

L1-RBM for Solar cell simulation, QoI convergence

$A_p(\lambda)$: FE approximation of absorption rate of sunlight with wavelength λ
$A_p^{\mathrm{RB}}(\lambda; N)$: RB approximation of $A_p(\lambda)$, using N bases



L1-RBM for Solar cell, QoI convergence in action, λ-sweep



Generalized SPDE [7]

Model equation, Wick product, Multi-index

$$\partial_t u(t,x) = \mathcal{L}u(t,x) + \mathcal{M}u(t,x) \diamond \dot{\mathfrak{N}}(t), \quad (t,x) \in (0,T] \times D,$$
$$u(0,x) = u_0(x), \quad x \in D$$

$$u = \sum_{\alpha\in\mathcal{J}} u_\alpha \Phi_\alpha, \quad \Phi_\alpha \diamond \Phi_\beta = \Phi_{\alpha+\beta}, \quad u \diamond v = \sum_{\alpha\in\mathcal{J}} \sum_{\beta\in\mathcal{J}} u_\alpha v_\beta \Phi_{\alpha+\beta}$$

$$\mathcal{J} = \{\alpha = (\alpha_k)_{k=1}^{\infty} : \alpha_k \geq 0, \ |\alpha| < \infty\}. \quad \xi^\alpha := \prod_{k=1}^{\infty} \xi_k^{\alpha_k}, \ \alpha! := \prod_{k=1}^{\infty} \alpha_k!.$$

$$\Phi_\alpha := \prod_{k=1}^{\infty} \varphi_{\alpha_k}(\xi_k) \quad \Phi_{\varepsilon_0} = 1, \ \Phi_{\varepsilon_k} = \xi_k, \ \mathbb{E}[\Phi_\alpha \Phi_\beta] = \alpha! \delta_{\alpha\beta}.$$

[7] Mikulevicius, Rozovskii, 2016

Generalized SPDE [7]

Model equation, Wick product, Multi-index

$$\partial_t u(t,x) = \mathcal{L}u(t,x) + \mathcal{M}u(t,x) \diamond \dot{\mathfrak{N}}(t), \quad (t,x) \in (0,T] \times D,$$
$$u(0,x) = u_0(x), \quad x \in D$$

$$u = \sum_{\alpha\in\mathcal{J}} u_\alpha \Phi_\alpha, \quad \Phi_\alpha \diamond \Phi_\beta = \Phi_{\alpha+\beta}, \quad u \diamond v = \sum_{\alpha\in\mathcal{J}} \sum_{\beta\in\mathcal{J}} u_\alpha v_\beta \Phi_{\alpha+\beta}$$

$$\mathcal{J} = \{\alpha = (\alpha_k)_{k=1}^{\infty} : \alpha_k \geq 0, \ |\alpha| < \infty\}. \quad \xi^\alpha := \prod_{k=1}^{\infty} \xi_k^{\alpha_k}, \ \alpha! := \prod_{k=1}^{\infty} \alpha_k!.$$

$$\Phi_\alpha := \prod_{k=1}^{\infty} \varphi_{\alpha_k}(\xi_k) \quad \Phi_{\varepsilon_0} = 1, \ \Phi_{\varepsilon_k} = \xi_k, \ \mathbb{E}[\Phi_\alpha \Phi_\beta] = \alpha! \delta_{\alpha\beta}.$$

Driving noise $\dot{\mathfrak{N}}(t)$ and stochastic process

$$\dot{\mathfrak{N}}(t) = \sum_{k=1}^{\infty} m_k(t)\xi_k, \ \mathfrak{N}(t) = \int_0^t \dot{\mathfrak{N}}(s)\, ds = \sum_{k=1}^{\infty} \left(\int_0^t m_k(s)\, ds\right) \xi_k.$$

[7] Mikulevicius, Rozovskii, 2016

Discretizing the generalized SPDE and COF-RB[8]

The propagator system

$$\partial_t u_\alpha(t,x) = \mathcal{L}u_\alpha(t,x) + \sum_{\varepsilon_k \le \alpha} \mathcal{M}u_{\alpha-\varepsilon_k}(t,x)m_k(t), \ (t,x) \in (0,T] \times D,$$

$$u_\alpha(0,x) = u_0(x)\mathbb{1}_{\{\alpha=\varepsilon_0\}}, \quad x \in D.$$

[8] Chen, Rozovskii, Shu 2019; Liu, Chen, C., Shu, 2019

Discretizing the generalized SPDE and COF-RB[8]

The propagator system

$$\partial_t u_\alpha(t,x) = \mathcal{L} u_\alpha(t,x) + \sum_{\varepsilon_k \le \alpha} \mathcal{M} u_{\alpha-\varepsilon_k}(t,x) m_k(t), \ (t,x) \in (0,T] \times D,$$

$$u_\alpha(0,x) = u_0(x) \mathbb{1}_{\{\alpha=\varepsilon_0\}}, \quad x \in D.$$

Truncation

$$\mathcal{J}_{M,K} := \{\alpha \in \mathcal{J} : |\alpha| \le M, \ d(\alpha) \le K\}, \quad \mathcal{N} = \#(\mathcal{J}_{M,K}) = \binom{M+K}{M}.$$

Convergence: State of the art [Chen, Rozovskii, Shu 2019]

Exponential with respect to M, and cubic with respect to K.

[8] Chen, Rozovskii, Shu 2019; Liu, Chen, C., Shu, 2019

Challenges and plan

Challenges

1. No obvious parameterization.
2. Coupling $\longrightarrow$ traditional greedy algorithm not amenable.

Challenges and plan

Challenges

1. No obvious parameterization.
2. Coupling $\longrightarrow$ traditional greedy algorithm not amenable.

Plan

1. Use of the multi-index as a "parameter".
2. Keep-or-toss greedy algorithm to cope with coupling.

"Keep-or-toss" to deal with the Hierarchical coupling

1. Order all multi-indices α according to $|\alpha|$ to obtain $\{\alpha_1, \ldots\ldots \alpha_{\mathcal{N}}\}$
2. Set RB space $W_1 = \text{span}\left\{\vec{U}_{\alpha_1}\right\}$.
3. For $i = 2, \ldots, \mathcal{N}$ (semi-sequentially) do:
   a. If $\vec{U}^{\text{RB}}_{\alpha_i}$ is accurate enough, toss α_i.
   b. Otherwise, keep α_i and augment RB space by $\vec{U}_{\alpha_i}$.

1. Order all multi-indices α according to $|\alpha|$ to obtain $\{\alpha_1, \ldots\ldots \alpha_{\mathcal{N}}\}$

2. Set RB space $W_1 = \text{span}\left\{\vec{U}_{\alpha_1}\right\}$.

3. For $i = 2, \ldots, \mathcal{N}$ (semi-sequentially) do:

- **a.** If $\vec{U}^{\text{RB}}_{\alpha_i}$ is accurate enough, toss α_i.
- **b.** Otherwise, keep α_i and augment RB space by $\vec{U}_{\alpha_i}$.

Elman, Liao 2013

Numerical result: SODE

$$
\begin{aligned}
u'(t) &= u(t) + 1 + u(t) \diamond \dot{\mathfrak{N}}(t), \quad t \in [0, T] \\
u(0) &= 1.
\end{aligned}
$$

Method	(M, K)	N	$\mathcal{N}$	e_2^{RBM}	e_2^{ORI}	CPU Time COFRB_ODE	CPU Time Full
FE	$(8, 8)$	10	12870	4.00E-03	1.50E-02	1.12E-01	1
	$(9, 9)$	11	48620	3.30E-03	1.50E-02	3.43E-02	1
CN	$(8, 8)$	14	12870	2.86E-05	7.71E-05	3.41E-01	1
	$(9, 9)$	14	48620	2.98E-05	3.67E-05	1.85E-02	1

Numerical result: 2D SPDE

$$\partial_t u(t,x,y) = \left(\frac{1}{2}\partial_x^2 + \cos(x)\partial_y\right) u + \partial_x u \diamond \dot{\mathfrak{N}}(t), \text{ in } [0,T] \times [0,2\pi]^2$$

$$u(0,x,y) = \sin(2x)\sin(y), \quad (x,y) \in [0,2\pi]^2.$$

M.	$\mathcal{N}_x \times \mathcal{N}_x$	(M,K)	N	$\mathcal{N}$	e_2^{RBM}	e_2^{ORI}	CPU Time COFRB_PDE	CPU Time Full
FE	32×32	$(9,9)$	48	48620	2.01E-04	3.51E-04	1.37E-02	1
		$(10,10)$	52	184756	2.04E-04	3.49E-04	4.60E-03	1
	64×64	$(8,8)$	42	12870	1.91E-04	3.50E-04	1.15E-01	1
		$(9,9)$	48	48620	2.01E-04	3.50E-04	2.92E-02	1
CN	32×32	$(6,6)$	38	924	4.62E-05	3.89E-05	1.49E-02	1
		$(7,7)$	67	3432	3.35E-06	4.29E-06	1.03E-02	1
	64×64	$(6,6)$	48	924	7.24E-05		7301.45	NAN
		$(7,7)$	54	3432	3.69E-05		12399.77	NAN

Responses to challenge 2

Collocation (e.g. FDM), Reduced Collocation

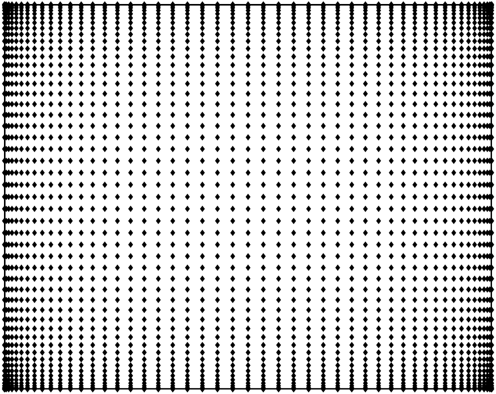
Reduced Over-Collocation (ROC)

Reduced Spectral Chebyshev Collocation: RBM → RCM [9]

Formulation for solving $\mathbb{L}(\mu)\, u_\mu(x) = f(x;\mu)$:

Seek $u_\mu^{\mathcal{N}} \in \prod_{\ell=1}^{d} \mathbb{P}_{\mathcal{N}_\ell}$ s. t.

$\mathbb{L}_{\mathcal{N}}(\mu)\, u_\mu^{\mathcal{N}}(x_j) = f(x_j;\mu)$ for $x_j \in$

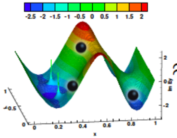
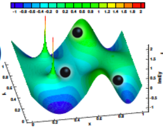
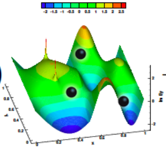
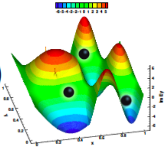


[9] C., Gottlieb 2013

Reduced Collocation Methods: Ideas

We combine $u^{\mathcal{N}}_{\mu^1}, u^{\mathcal{N}}_{\mu^2}, \ldots, u^{\mathcal{N}}_{\mu^N}$ to produce a surrogate solution $u^N(\mu^*)$:

$$u^N(\mu^*) = \sum_{j=1}^{N} c_j(\mu^*) u^{\mathcal{N}}_{\mu^j}.$$

 $\approx c_1(\mu)$  $+c_2(\mu)$  $+c_3(\mu)$ 

Key: it has to satisfy the PDE, $\mathbb{L}_{\mathcal{N}}(\mu^*)(u^N(\mu^*)) = f(x; \mu^*)$.

Reduced collocation points

A natural way is to enforce the PDE at a reduced set of points. For the linear case, this reads

$$\sum_{j=1}^{N} c_j(\mu^*)\mathbb{I}_{\mathcal{N}}^{N}\left(\mathbb{L}_{\mathcal{N}}(\mu^*)u_{\mu^j}^{\mathcal{N}}\right) = f(x;\mu^*). \quad \text{for} \quad x \in \mathbf{C_R^N}.$$

Tricky part: determination of C_R^N

First try: EIM points from the snapshots[10] X_s^N:

Hierarchical.

x^i is determined by $\{x^1, \ldots, x^{i-1}\}$ and $\{u_{\mu^1}^{\mathcal{N}}, \ldots, u_{\mu^i}^{\mathcal{N}}\}$.

[10] C., Gottlieb 2013, C., Gottlieb, Maday 2014

Reduced collocation points

A natural way is to enforce the PDE at a reduced set of points. For the linear case, this reads

$$\sum_{j=1}^{N} c_j(\mu^*)\mathbb{I}_{\mathcal{N}}^{N}\left(\mathbb{L}_{\mathcal{N}}(\mu^*)u_{\mu^j}^{\mathcal{N}}\right) = f(x;\mu^*). \quad \text{for} \quad x \in \mathbf{C_R^N}.$$

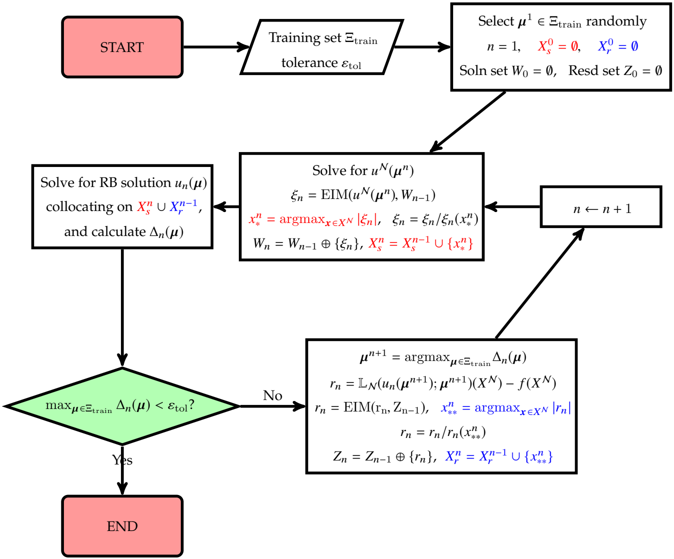
Tricky part: determination of C_R^N

First try: EIM points from the snapshots[10] X_s^N:

- Hierarchical.
- x^i is determined by $\{x^1, \ldots, x^{i-1}\}$ and $\{u_{\mu^1}^{\mathcal{N}}, \ldots, u_{\mu^i}^{\mathcal{N}}\}$.
- Works well ...... but can be better.

[10] C., Gottlieb 2013, C., Gottlieb, Maday 2014

Reduced over-collocation[11] points: $X_s^N \cup X_r^{N-1}$



[11] C., Gottlieb, Ji, Maday, Xu, arXiv: 1906.07349

(Over) collocation, in a nutshell

Goal: Removing the dependence on Q_h (from EIM) when designing RBM for nonlinear and nonaffine problems.

(Over) collocation, in a nutshell

Goal: Removing the dependence on Q_h (from EIM) when designing RBM for nonlinear and nonaffine problems. (i.e. Making $Q_h = 1$.)

1. N basis $\longrightarrow$ reduced solver is collocating at N points (taken out of a, e.g., tensorial grid of $\mathcal{N} \times \mathcal{N}$) $\longrightarrow$ $Q_h = 1$ for the reduced solver.

(Over) collocation, in a nutshell

Goal: Removing the dependence on Q_h (from EIM) when designing RBM for nonlinear and nonaffine problems. (i.e. Making $Q_h = 1$.)

1. N basis $\longrightarrow$ reduced solver is collocating at N points (taken out of a, e.g., tensorial grid of $\mathcal{N} \times \mathcal{N}$) $\longrightarrow Q_h = 1$ for the reduced solver.
2. $Q_h > 1$ for the error estimator in the greedy algorithm $\longrightarrow$ L1-RBM making $Q_h = 1$ for greedy.

(Over) collocation, in a nutshell

Goal: Removing the dependence on Q_h (from EIM) when designing RBM for nonlinear and nonaffine problems. (i.e. Making $Q_h = 1$.)

1. N basis $\longrightarrow$ reduced solver is collocating at N points (taken out of a, e.g., tensorial grid of $\mathcal{N} \times \mathcal{N}$) $\longrightarrow$ $Q_h = 1$ for the reduced solver.
2. $Q_h > 1$ for the error estimator in the greedy algorithm $\longrightarrow$ L1-RBM making $Q_h = 1$ for greedy.
3. Instability $\longrightarrow$ (analytic) preconditioning

(Over) collocation, in a nutshell

Goal: Removing the dependence on Q_h (from EIM) when designing RBM for nonlinear and nonaffine problems. (i.e. Making $Q_h = 1$.)

1. N basis $\longrightarrow$ reduced solver is collocating at N points (taken out of a, e.g., tensorial grid of $\mathcal{N} \times \mathcal{N}$) $\longrightarrow$ $Q_h = 1$ for the reduced solver.
2. $Q_h > 1$ for the error estimator in the greedy algorithm $\longrightarrow$ L1-RBM making $Q_h = 1$ for greedy.
3. Instability $\longrightarrow$ (analytic) preconditioning
4. A more elegant approach: over collocation at $2N - 1$ locations

(Over) collocation, in a nutshell

Goal: Removing the dependence on Q_h (from EIM) when designing RBM for nonlinear and nonaffine problems. (i.e. Making $Q_h = 1$.)

1. N basis $\longrightarrow$ reduced solver is collocating at N points (taken out of a, e.g., tensorial grid of $\mathcal{N} \times \mathcal{N}$) $\longrightarrow$ $Q_h = 1$ for the reduced solver.
2. $Q_h > 1$ for the error estimator in the greedy algorithm $\longrightarrow$ L1-RBM making $Q_h = 1$ for greedy.
3. Instability $\longrightarrow$ (analytic) preconditioning
4. A more elegant approach: over collocation at $2N - 1$ locations
   - One set of N points resolving the solution/basis well.
   - One set of $N - 1$ points resolving the (representative) residuals well.

Let's over-collocate: equations tested [12]

Steady-state problems

$$uu_x = \mu u_{xx} + f(x)$$
$$D\nabla^2 u = \sinh u + g(\boldsymbol{x})$$
$$-\mu_2 \Delta u + u(u - \mu_1)^2 = f(\boldsymbol{x})$$
$$-\mu_2 \Delta u + u(||\nabla u|| + \mu_1)^{1.5} = f(\boldsymbol{x})$$

Time-dependent problems

$$u_t + uu_x = \mu u_{xx} + f(x)$$
$$u_t + D\nabla^2 u = \sinh u + g(\boldsymbol{x})$$
$$u_t - \mu_2 \Delta u + u(u - \mu_1)^2 = f(\boldsymbol{x})$$
$$u_t - \mu_2 \Delta u + u(||\nabla u|| + \mu_1)^{1.5} = f(\boldsymbol{x})$$

[12] C., Gottlieb, Ji, Maday, Xu, arXiv: 1906.07349; C. Ji, Narayan, Xu, Preprint.

Let's over-collocate: results for cubic reaction, steady

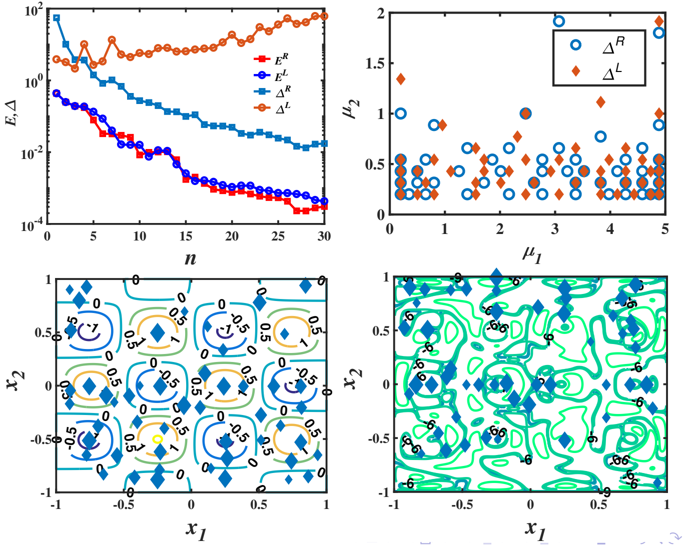
T/B: Top/Bottom **L/R:** Left/Right

TL: Convergence of error estimator/indicator and errors

TR: Chosen parameter values

BL: N points from solutions X_s^{30}

BR: $N-1$ points from residuals X_r^{29}



Let's over-collocate: results for cubic reaction, transient

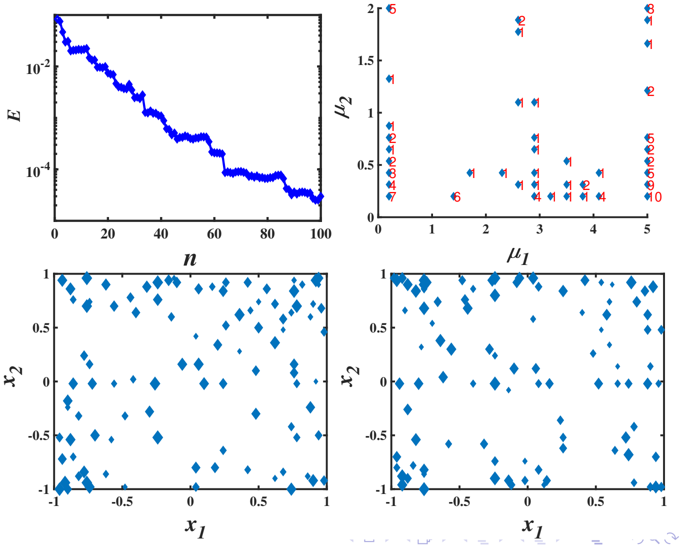
T/B: Top/Bottom **L/R:** Left/Right

TL: Convergence of error estimator/indicator and errors

TR: Chosen parameter values

BL: N points from solutions X_s^{100}

BR: $N-1$ points from residuals X_r^{99}



Thank You for your attention!

Yanlai.Chen @ umassd.edu

Certified Highly Efficient Numerics
CHEN Lab
@
UMass | Dartmouth