Reduced basis methods for the resolution of parameter-dependent PDEs
MS13

Elise Grosjean

Ensta
Institut Polytechnique de Paris

Reduced Basis Methods (RBM)

Linear space

Manifold

Kolmogorov

Reduce

Approximation

TP RB+FEM

- ◇ **Offline** Construction of a reduced space V_N spanned by a reduced basis $\rightarrow$???
- ◇ **Online** Computation of the reduced coefficients $\alpha \rightarrow$???

TP RB+FEM

◇ **Offline** Construction of a reduced space V_N spanned by a reduced basis $\rightarrow$???

$V_N = Span\{u_1, \ldots, u_N\}$

We approximate $u(\mathbf{x}, \boldsymbol{\mu})$ by $\sum_{k=1}^{Ntrain} \alpha_k(\boldsymbol{\mu}) u_k(\mathbf{x})$

$$\boxed{u_i = \mathrm{argmax}_{u \in \mathcal{M}} \|u - P_{V^{i-1}}(u)\|,}$$

$$\boxed{V^i = \mathrm{Span}(V^{i-1}, u_i).}$$

◇ **Online** Computation of the reduced coefficients $\alpha \rightarrow$???

TP RB+FEM

◇ **Offline** Construction of a reduced space V_N spanned by a reduced basis $\rightarrow$???

$V_N = Span\{u_1, \ldots, u_N\}$

We approximate $u(\mathbf{x}, \boldsymbol{\mu})$ by $\sum_{k=1}^{Ntrain} \alpha_k(\boldsymbol{\mu}) u_k(\mathbf{x})$

$$u_i = \text{argmax}_{u \in \mathcal{M}} \|u - P_{V^{i-1}}(u)\|,$$

$$V^i = \text{Span}(V^{i-1}, u_i).$$

`Construct_RB_Greedy(...)`

`Construct_RB_Weak_Greedy(...)`

◇ **Online** Computation of the reduced coefficients $\alpha \rightarrow$???

TP RB+FEM

◇ **Offline** Construction of a reduced space V_N spanned by a reduced basis $\rightarrow$???

$V_N = Span\{u_1, \ldots, u_N\}$

We approximate $u(\mathbf{x}, \boldsymbol{\mu})$ by $\sum_{k=1}^{Ntrain} \alpha_k(\boldsymbol{\mu}) u_k(\mathbf{x})$

$$u_i = \text{argmax}_{u \in \mathcal{M}} \|u - P_{V^{i-1}}(u)\|,$$

$$V^i = \text{Span}(V^{i-1}, u_i).$$

`Construct_RB_Greedy(...)`

`Construct_RB_Weak_Greedy(...)`

◇ **Online** Computation of the reduced coefficients $\boldsymbol{\alpha} \rightarrow$???

TP RB+FEM

◇ **Offline** Construction of a reduced space V_N spanned by a reduced basis $\rightarrow$???

$V_N = Span\{u_1, \ldots, u_N\}$

We approximate $u(\mathbf{x}, \boldsymbol{\mu})$ by $\sum_{k=1}^{Ntrain} \alpha_k(\boldsymbol{\mu}) u_k(\mathbf{x})$

$$u_i = \text{argmax}_{u \in \mathcal{M}} \|u - P_{V^{i-1}}(u)\|,$$

$$V^i = \text{Span}(V^{i-1}, u_i).$$

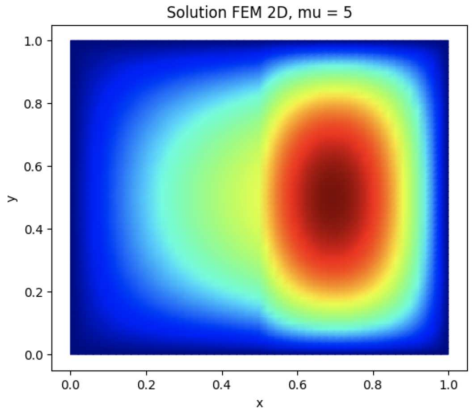
```
Construct_RB_Greedy(...)
Construct_RB_Weak_Greedy(...)
```

◇ **Online** Computation of the reduced coefficients $\boldsymbol{\alpha} \rightarrow$???

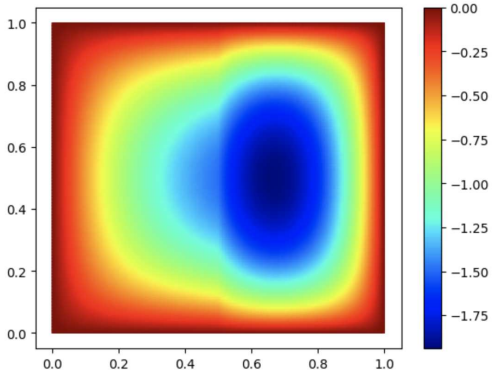
$\mathbf{P}^T \mathbf{A}(\boldsymbol{\mu}) \mathbf{P}\ \boldsymbol{\alpha}(\boldsymbol{\mu}) = \mathbf{P}^T \mathbf{l}(\boldsymbol{\mu})$

```
solve_fem_rom(A,b,mu, Phi,m)
```

Greedy like ...

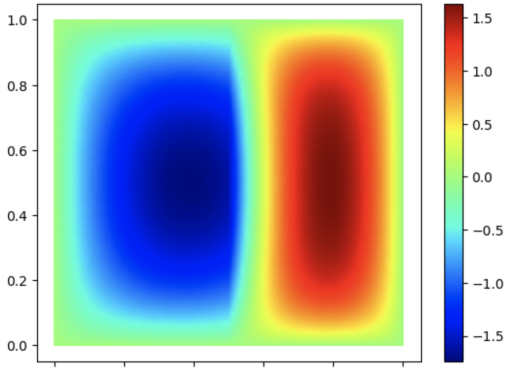


TP Kolmogorov



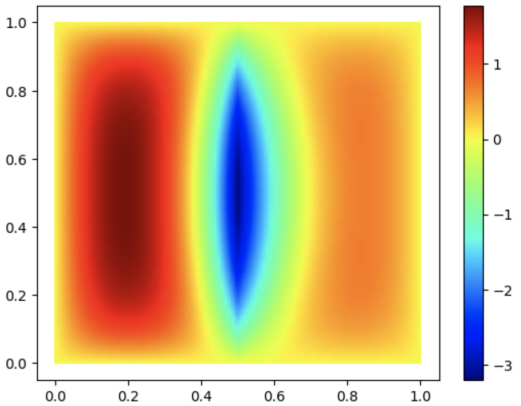
mode 1

TP Kolmogorov



mode 2

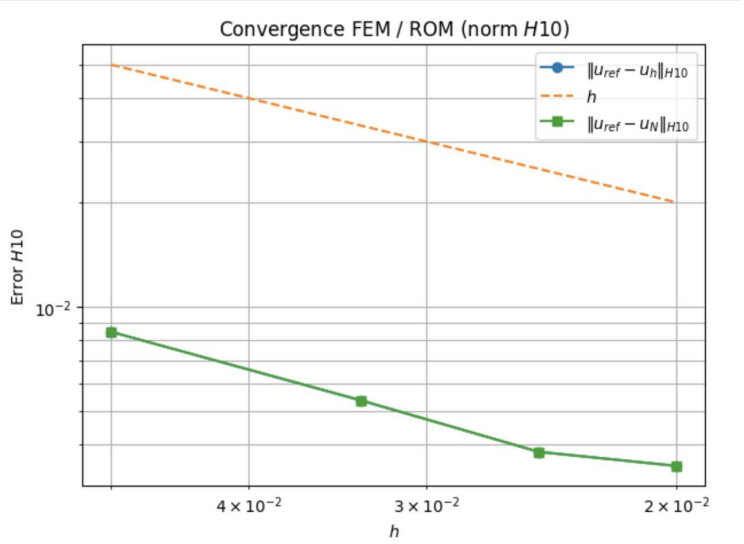
TP Kolmogorov



mode 3

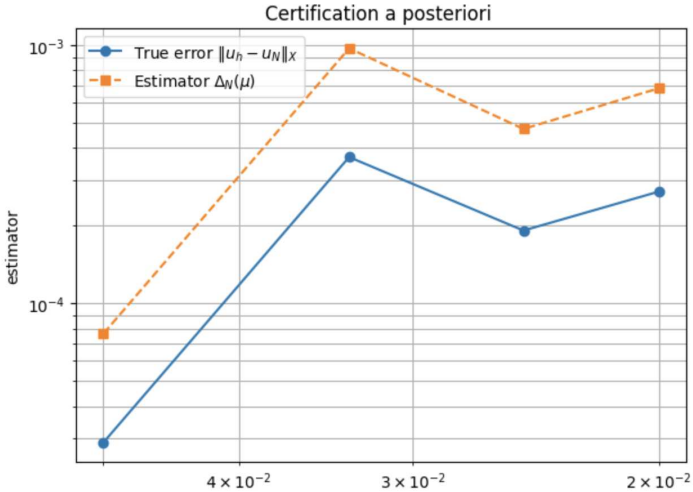
TP RB+FEM

Expected results: Convergence in $\mathcal{O}(h)$!!

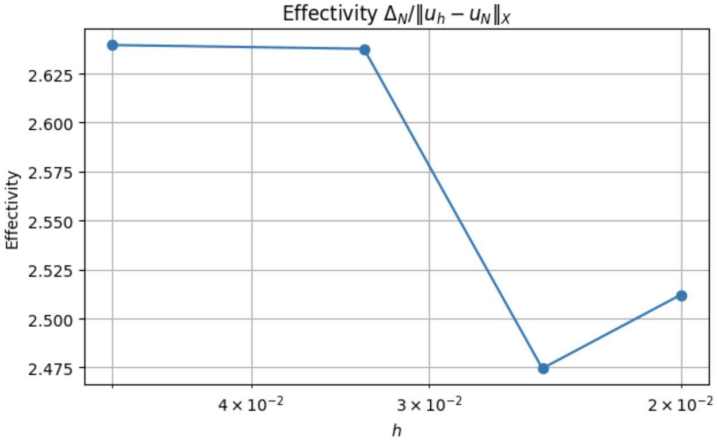


TP RB+FEM

Expected results:
$\mu = [1, 2, 3, 4]$



TP RB+FEM


Effectivity $\Delta_N/\|u_h - u_N\|_X$
Effectivity
2.625
2.600
2.575
2.550
2.525
2.500
2.475
4×10^{-2}
3×10^{-2}
2×10^{-2}
h

TP RB+FEM

- ◇ Keep HF precision $\surd$
- ◇ Reduce computational costs ?

TP RB+FEM

```
import time
start = time.time()
```

- ◇ POD
- ◇ Greedy
- ◇ Weak Greedy (with a posteriori estimator)

We are going to use $\|u_h(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_V \leq \frac{1}{\alpha_{sta}(\boldsymbol{\mu})} \|r_N(\boldsymbol{\mu})\|_{V'}$

```
end = time.time()
print("Time :", end - start, "secondes")
```

Aposteriori

Two key ingredients:

- ◇ Dual norm of the residual: Offline-online computation strategy
- ◇ Inf-sup $\alpha_{sta}(\boldsymbol{\mu})$ not efficiently computable but one can compute $\alpha_{LB}(\boldsymbol{\mu})$ such that

$$\forall \boldsymbol{\mu} \in \mathcal{G}, \alpha_{sta}(\boldsymbol{\mu}) \geq \alpha_{LB}(\boldsymbol{\mu})$$

EIM

Non-linear parameters

EIM = Empirical Interpolation Method

EIM

Non-linear parameters
EIM = Empirical Interpolation Method

Model problem

Let's get back to our sheep

linear second-order parameter dependent problem

$$\begin{cases} -\nabla \cdot \big(A(\boldsymbol{\mu})\nabla u\big) = f(\boldsymbol{\mu}) & \text{dans } \Omega, \\ u = 0 & \text{sur } \partial\Omega. \end{cases}$$

In weak form: find $u(\boldsymbol{\mu}) \in V := H_0^1(\Omega)$ such that

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = \ell(v), \qquad \forall v \in V,$$

where

$$a(w, v; \boldsymbol{\mu}) = \int_\Omega A(\mathbf{x}; \boldsymbol{\mu})\, \nabla w \cdot \nabla v \, d\mathbf{x}, \qquad \ell(v; \boldsymbol{\mu}) = \int_\Omega f(\boldsymbol{\mu}) v \, d\mathbf{x}.$$

Why EIM?

Definition: Affine operator

the bilinear form $a(\cdot,\cdot;\ \boldsymbol{\mu}) : V \times V \to \mathbb{R}$ is affine if there exist

- $\diamond$ M_a functions $\theta_m^a : \mathcal{G} \to \mathbb{R},\ 1 \leq m \leq M_a$ bounded (i.e. $\theta_m^a(\mathcal{G})$),
- $\diamond$ M_a functions $a_m : V \times V \to \mathbb{R},\ 1 \leq m \leq M_a$

such that

$$\forall \boldsymbol{\mu} \in \mathcal{G},\ \forall (u,v) \in V \times V, a(u,v;\ \boldsymbol{\mu}) = \sum_{m=1}^{M_a} \theta_m^a(\boldsymbol{\mu}) a_m(u,v).$$

$\mathbf{A}(\boldsymbol{\mu}) = \sum_{m=1}^{M_a} \theta_m^a(\boldsymbol{\mu}) \mathbf{A}_m.$

Why EIM?

e.g. Tp4: $\Omega = \Omega_1 \cup \Omega_2$,
e.g. Tp5: $\Omega = \Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4$,

Reduced basis Galerkin approximation

Online assembling cost with the affine operators:
$\mathcal{O}(N^2 M_a + N M_l)$ with

$$\mathbf{P}^T\mathbf{A}(\boldsymbol{\mu})\mathbf{P} = \sum_{m=1}^{M_a} \theta_m^a(\boldsymbol{\mu}) \underbrace{\mathbf{P}^T\mathbf{A}_m\mathbf{P}}_{\text{precomputed offline}}, \quad \mathbf{P}^T\mathbf{l}(\boldsymbol{\mu}) = \sum_{m=1}^{M_l} \theta_m^l(\boldsymbol{\mu}) \underbrace{\mathbf{P}^T\mathbf{l}_m}_{\text{precomputed offline}}.$$

Instead of $\mathcal{O}(\mathcal{N}^2\ N)$ for $\mathbf{A}$ and of $\mathcal{O}(\mathcal{N}\ N)$ for $\mathbf{l}$

Why EIM?

$M = M_a$

$$a(w, v; \boldsymbol{\mu}) = \sum_{m=1}^{M} \theta_m(\boldsymbol{\mu}) a_m(u, v).$$

Then:

- Offline: preassemble the parameter-independent operators,
- Online: only evaluate scalar coefficients and combine small matrices.

If $A(x; \boldsymbol{\mu})$ is non-affine in $\boldsymbol{\mu}$, this separation is not directly available!

Idea: approximate the coefficient itself by a separated expansion using EIM.

Why EIM?

$M = M_a$

$$a(w, v; \boldsymbol{\mu}) = \sum_{m=1}^{M} \theta_m(\boldsymbol{\mu}) a_m(u, v).$$

Then:

- Offline: preassemble the parameter-independent operators,
- Online: only evaluate scalar coefficients and combine small matrices.

If $A(x; \boldsymbol{\mu})$ is non-affine in $\boldsymbol{\mu}$, this separation is not directly available!

Idea: approximate the coefficient itself by a separated expansion using EIM.

Why EIM?

What if e.g. $A(\mathbf{x}, \boldsymbol{\mu}) = \frac{1}{(x_1-\mu_1)^2+(x_2-\mu_2)^2}$?

We approximate the coefficient by

$$A_M(\mathbf{x}; \boldsymbol{\mu}) := \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu})\, q_m(\mathbf{x})$$

Why EIM?

What if e.g. $A(\mathbf{x}, \boldsymbol{\mu}) = \frac{1}{(x_1-\mu_1)^2+(x_2-\mu_2)^2}$?

We approximate the coefficient by

$$A_M(\mathbf{x}; \boldsymbol{\mu}) := \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu})\, q_m(\mathbf{x})$$

EIM: Abstract setting

Assume Ω is compact. Consider continuous functions:

$$A \in C(\Omega \times \mathcal{G}; \mathbb{R})$$

and

$$X = C(\Omega; \mathbb{R})$$

Given basis functions $q_1, \ldots, q_M \in X$, define

$$\mathcal{X}_M := \text{span}\{q_1, \ldots, q_M\} \subset X.$$

Kolmogorov width?

$$\mathcal{M}_A = \{A(\cdot, \boldsymbol{\mu}), \boldsymbol{\mu} \in \mathcal{G}\}$$

Choose a training set of parameters $\mathcal{G}_{train}$ and corresponding "snapshots" $A(\cdot; \boldsymbol{\mu})$.

EIM

We approximate the coefficient by

$$A_M(\mathbf{x};\boldsymbol{\mu}) := I_M(A)(\mathbf{x};\boldsymbol{\mu}) = \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu})\, q_m(\mathbf{x})$$

where

- I_M is an interpolation operator,
- M is the number of selected "magic" points. Interpolation conditions are imposed at these selected points

$$\mathbf{x}_1, \ldots, \mathbf{x}_M \in \Omega : \quad A(\mathbf{x}_n;\boldsymbol{\mu}) = I_M(A)(\mathbf{x}_n;\boldsymbol{\mu}), \quad n = 1, \ldots, M.$$

- $(q_1, \ldots, q_M)$ is the set of EIM basis functions,
- $\beta_m(\boldsymbol{\mu})$ are parameter-dependent coefficients.

Matrix form and online coefficients

Define the interpolation matrix

$$\mathbf{Q}_M = \big(q_j(\mathbf{x}_i)\big)_{1\le i,j\le M}$$

and the vector of sampled values

$$\mathbf{G}(\boldsymbol{\mu}) = \big(A(\mathbf{x}_1;\boldsymbol{\mu}),\dots,A(\mathbf{x}_M;\boldsymbol{\mu})\big)^T.$$

Then

$$\boxed{I_M(A)(\mathbf{x};\boldsymbol{\mu}) = \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu})\, q_m(\mathbf{x})} \Leftrightarrow \boxed{\mathbf{Q}_M\, \boldsymbol{\beta}(\boldsymbol{\mu}) = \mathbf{G}(\boldsymbol{\mu}),}$$

$$\boldsymbol{\beta}(\boldsymbol{\mu}) = \big(\beta_1(\boldsymbol{\mu}),\dots,\beta_M(\boldsymbol{\mu})\big)^T.$$

Matrix form and online coefficients

$$\boxed{\mathbf{Q}_M \, \boldsymbol{\beta}(\boldsymbol{\mu}) = \mathbf{G}(\boldsymbol{\mu}).}$$

Offline consists in

- ◇ finding the magic points,
- ◇ generating the basis functions q_j and assembling $\mathbf{Q}_M$,

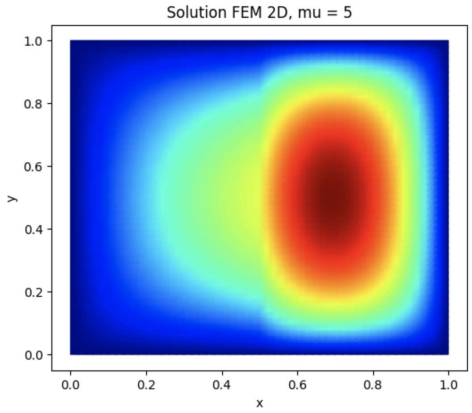
And then, the **online** stage for a new parameter value $\boldsymbol{\mu}$ only requires:

- ◇ evaluating $A(\mathbf{x}_n; \boldsymbol{\mu})$ at the M magic points to get $\mathbf{G}(\boldsymbol{\mu})$
- ◇ solving a small $M \times M$ linear system,
- ◇ forming

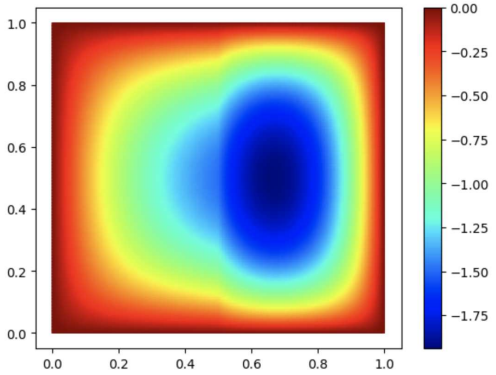
$$A_M(\mathbf{x}; \boldsymbol{\mu}) = \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu}) \; q_m(\mathbf{x}).$$

Offline greedy construction of EIM

Greedy like ...

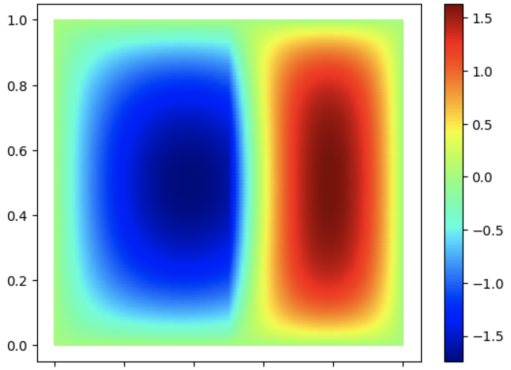


TP Kolmogorov



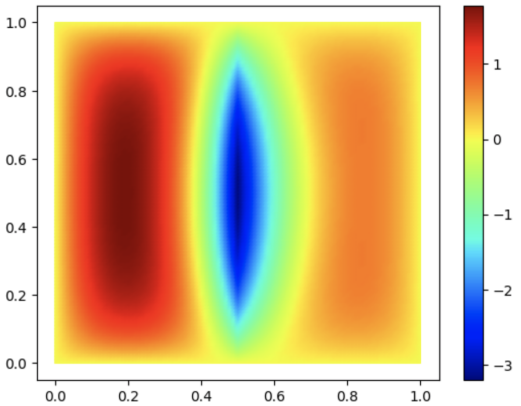
mode 1

TP Kolmogorov



mode 2

TP Kolmogorov



mode 3

Offline greedy construction of EIM

EIM builds $(q_k, \mathbf{x}_k)$ iteratively using a greedy residual criterion.

Offline greedy construction of EIM

Initialization:

- ◇ Choose $\boldsymbol{\mu}_1$ such that

$$\|A(\cdot;\boldsymbol{\mu})\|_{L^\infty(\Omega)}$$

  is maximal.
- ◇ Choose $\mathbf{x}_1$ such that

$$\mathbf{x}_1 = \arg\max_{\mathbf{x}\in\Omega} |A(\mathbf{x};\boldsymbol{\mu}_1)|.$$

- ◇ Define $q_1(\mathbf{x})$

...

- ◇ Define the residual

$$r_m(\mathbf{x}) = A(\mathbf{x};\boldsymbol{\mu}_m) - I_{m-1}(A)(\mathbf{x};\boldsymbol{\mu}_m).$$

- ◇ Select the next magic point at maximal residual magnitude:

$$\mathbf{x}_m = \arg\max_{\mathbf{x}\in\Omega} |r_m(\mathbf{x})|.$$

Offline greedy construction of EIM

$$q_1(\mathbf{x}_1) = 1.$$

$$\mathbf{x}_1 = \arg\max_{\mathbf{x}\in\Omega} |A(\mathbf{x}; \boldsymbol{\mu}_1)|.$$

$$q_1(\mathbf{x}) = \frac{A(\mathbf{x}, \boldsymbol{\mu}_1)}{A(\mathbf{x}_1, \boldsymbol{\mu}_1)}$$

...

$$q_m(\mathbf{x}_m) = 1.$$

$$\mathbf{x}_m = \arg\max_{\mathbf{x}\in\Omega} |r_m(\mathbf{x})|.$$

$$q_m(\mathbf{x}) = \frac{r_m(\mathbf{x})}{r_m(\mathbf{x}_m)}$$

Offline greedy construction of EIM

How do we find I_{m-1} in residual?

Offline greedy construction of EIM

For each parameter $\boldsymbol{\mu}$,

$$\boxed{I_1(A)(\cdot;\boldsymbol{\mu}) = \beta_1(\boldsymbol{\mu})\, q_1(\cdot)}$$

such that

$$I_1(A)(\mathbf{x}_1;\boldsymbol{\mu}) = A(\mathbf{x}_1;\boldsymbol{\mu})$$

Thus,

$$\beta_1(\boldsymbol{\mu}) = A(\mathbf{x}_1;\boldsymbol{\mu}), \quad q_1(\mathbf{x}_1) = 1. \text{ and therefore } \boxed{I_1(A)(\cdot;\boldsymbol{\mu}) = A(\mathbf{x}_1;\boldsymbol{\mu})\, q_1(\cdot).}$$

$I_1(A)(\cdot;\boldsymbol{\mu})$ is the unique function

- in the space $\mathrm{span}\{q_1\}$,
- such that $I_1(A)(\mathbf{x}_1;\boldsymbol{\mu}) = A(\mathbf{x}_1;\boldsymbol{\mu})$.

$$T_1 = \{x_1\}, \qquad \mathcal{X}_1 = \mathrm{span}\{q_1\}.$$

Offline greedy construction of EIM

For each parameter $\boldsymbol{\mu}$,

$$\boxed{I_2(A)(\cdot;\boldsymbol{\mu}) = \beta_1(\boldsymbol{\mu})\, q_1(\cdot) + \beta_2(\boldsymbol{\mu})\, q_2(\cdot)}$$

such that

$$I_2(A)(\mathbf{x}_i;\boldsymbol{\mu}) = A(\mathbf{x}_i;\boldsymbol{\mu}), \quad i = 1, 2$$

$$\begin{aligned}\beta_1(\boldsymbol{\mu})\, q_1(\mathbf{x}_1) + \beta_2(\boldsymbol{\mu})\, q_2(\mathbf{x}_1) &= A(\mathbf{x}_1;\boldsymbol{\mu}),\\ \beta_1(\boldsymbol{\mu})\, q_1(\mathbf{x}_2) + \beta_2(\boldsymbol{\mu})\, q_2(\mathbf{x}_2) &= A(\mathbf{x}_2;\boldsymbol{\mu}).\end{aligned}$$

where $q_2(\mathbf{x}_1) = 0, \qquad q_2(\mathbf{x}_2) = 1$. Thus,

$$\beta_1(\boldsymbol{\mu}) = A(\mathbf{x}_1;\boldsymbol{\mu}), \text{ and } \beta_2(\boldsymbol{\mu}) = A(\mathbf{x}_2;\boldsymbol{\mu}) - A(\mathbf{x}_1;\boldsymbol{\mu})\, q_1(\mathbf{x}_2).$$

$I_2(A)(\cdot;\boldsymbol{\mu})$ is the unique function

- in the space $\operatorname{span}\{q_1, q_2\}$,
- such that $I_2(A)(\mathbf{x}_i;\boldsymbol{\mu}) = A(\mathbf{x}_i;\boldsymbol{\mu})$, for $i = 1, 2$.

$$T_2 = \{\mathbf{x}_1, \mathbf{x}_2\}, \qquad \mathcal{X}_2 = \operatorname{span}\{q_1, q_2\}.$$

Offline greedy construction of EIM

For each parameter $\boldsymbol{\mu}$,

$$\boxed{I_3(A)(\cdot;\boldsymbol{\mu}) = \beta_1(\boldsymbol{\mu})\, q_1(\cdot) + \beta_2(\boldsymbol{\mu})\, q_2(\cdot) + \beta_3(\boldsymbol{\mu})\, q_3(\cdot)}$$

such that

$$I_3(A)(\mathbf{x}_i;\boldsymbol{\mu}) = A(\mathbf{x}_i;\boldsymbol{\mu}), \qquad i = 1,2,3,$$

with

$$q_3(\mathbf{x}_1) = 0, \qquad q_3(\mathbf{x}_2) = 0, \qquad q_3(\mathbf{x}_3) = 1.$$

$\mathbf{Q}_3\boldsymbol{\beta}(\boldsymbol{\mu}) = \mathbf{G}(\boldsymbol{\mu}) \Leftrightarrow$

$\beta_1(\boldsymbol{\mu}) = A(\mathbf{x}_1;\boldsymbol{\mu}),\ \beta_2(\boldsymbol{\mu}) = A(\mathbf{x}_2;\boldsymbol{\mu}) - A(\mathbf{x}_1;\boldsymbol{\mu})\, q_1(\mathbf{x}_2),\ \beta_3(\boldsymbol{\mu}) = A(\mathbf{x}_3;\boldsymbol{\mu}) - A(\mathbf{x}_1;\boldsymbol{\mu})\, q_1(\mathbf{x}_3) - A(\mathbf{x}_2;\boldsymbol{\mu}$

$I_3(A)(\cdot;\boldsymbol{\mu})$ is the unique function

- in the space $\mathrm{span}\{q_1, q_2, q_3\}$,
- such that $I_3(A)(x_i;\boldsymbol{\mu}) = A(\mathbf{x}_i;\boldsymbol{\mu})$, for $i = 1,2,3$.

$$T_3 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}, \qquad \mathcal{X}_3 = \mathrm{span}\{q_1, q_2, q_3\}.$$

Offline greedy construction of EIM

At step M, assuming $q_1, \ldots, q_{M-1}$ and $\mathbf{x}_1, \ldots, \mathbf{x}_{M-1}$ are known:

1. interpolate the current snapshot using the first $M-1$ modes,
2. compute the residual

$$r_M(\mathbf{x}) = A(\mathbf{x}; \boldsymbol{\mu}_M) - I_{M-1}(A)(\mathbf{x}; \boldsymbol{\mu}_M),$$

3. select the next magic point at maximal residual magnitude,

$$\mathbf{x}_M = \arg\max_{\mathbf{x}\in\Omega} |r_M(\mathbf{x})|,$$

4. normalize the new basis function

$$q_M(\mathbf{x}) = \frac{r_M(\mathbf{x})}{r_M(\mathbf{x}_M)}.$$

5. find β_M for each training parameters

Offline greedy construction of EIM

Algorithm Greedy construction of the EIM basis

1: Choose $\mu_1 \in \arg\max_{\mu \in \mathcal{G}_{\text{train}}} \|A(\cdot;\mu)\|_{L^\infty(\Omega)}$
2: Choose $x_1 \in \arg\max_{x \in \Omega} |A(x;\mu_1)|$
3: Define $q_1(x) = \frac{A(x;\mu_1)}{A(x_1;\mu_1)}$
4: **for** $m = 2, \ldots, M$ **do**
5: Choose $\mu_m \in \arg\max_{\mu \in \mathcal{G}_{\text{train}}} \|A(\cdot;\mu) - I_{m-1}(A(\cdot;\mu))\|_{L^\infty(\Omega)}$
6: Define the residual $r_m(x) = A(x;\mu_m) - I_{m-1}(A(\cdot;\mu_m))(x)$
7: Choose $x_m \in \arg\max_{x \in \Omega} |r_m(x)|$
8: Define $q_m(x) = \frac{r_m(x)}{r_m(x_m)}$
9: Let $\beta_m(\mu) = A(x_m;\mu) - I_{M-1}(A)(x_M;\mu)$ for each training parameters
10: **end for**

(In practice, "maximize" is approximated using a training set.)

Offline greedy construction of EIM

Is the interpolation well defined? Is the small $M \times M$ linear system well posed?

Offline greedy construction of EIM

- $\mathbf{x}_m$ is added to $\mathbf{x}_1, \ldots, \mathbf{x}_{m-1}$ to form T_m
- The $\mathcal{X}_m$ are nested : $\mathcal{X}_m = \mathcal{X}_{m-1} \oplus \mathrm{span}\{q_m\}$

This gives

$$q_M(\mathbf{x}_M) = 1, \qquad q_M(\mathbf{x}_n) = 0 \quad (n < M),$$

so $\mathbf{Q}_M$ is lower triangular with unit diagonal.

Offline greedy construction of EIM

- $\mathbf{x}_m$ is added to $\mathbf{x}_1, \ldots, \mathbf{x}_{m-1}$ to form T_m
- The $\mathcal{X}_m$ are nested : $\mathcal{X}_m = \mathcal{X}_{m-1} \oplus \mathrm{span}\{q_m\}$

This gives

$$q_M(\mathbf{x}_M) = 1, \qquad q_M(\mathbf{x}_n) = 0 \quad (n < M),$$

so $\mathbf{Q}_M$ is lower triangular with unit diagonal.

Abstract setting

Theorem: Well-posedness of eim

For a function $g \in X$, the empirical interpolant $I_M(g) \in \mathcal{X}_M$ is defined by

$$\forall \boldsymbol{\mu} \in \mathcal{G},\ I_M(g)(\mathbf{x}_n, \boldsymbol{\mu}) = \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu}) q_m(\mathbf{x}_n) = g(\mathbf{x}_n, \boldsymbol{\mu}), \qquad n = 1, \ldots, M.$$

Since $\mathbf{Q}_M$ is invertible, this interpolant is uniquely defined.

Abstract setting

How good is this interpolation?

Lebesgue constant

When constructing an interpolation procedure, two questions naturally arise:

- how accurate is the interpolation?
- how stable is it with respect to the choice of interpolation points?

$\Rightarrow$ **Lebesgue constant**.

EIM error

Lebesgue-type constant

$$\Lambda_M := \sup_{g \neq 0} \frac{\|I_M(g)\|_{L^\infty}}{\|g\|_{L^\infty}}.$$

Then for every $g \in X$,

$$\boxed{\|g - I_M(g)\|_\infty \leq (1 + \Lambda_M) \inf_{v \in \mathcal{X}_M} \|g - v\|_\infty.}$$

So the EIM error is controlled by:

- ◇ the approximation quality of $\mathcal{X}_M$,
- ◇ the stability factor Λ_M.

Growth of the Lebesgue constant

The Lebesgue constant is not uniformly bounded as $M \to \infty$.

In fact, with polynomials in $1D$, one has the lower bound

$$\Lambda_M \geq C \log M,$$

for some constant $C > 0$.

This logarithmic growth is essentially achieved by the Chebyshev points.

Growth of the Lebesgue constant

For more general choices of interpolation points, the situation may be much worse: the Lebesgue constant can grow exponentially with M!

Then for every $g \in X$,

$$\boxed{\|g - I_M(g)\|_\infty \leq (1 + \Lambda_M) \inf_{v \in \mathcal{X}_M} \|g - v\|_\infty.}$$

Hence, even when the best approximation error decreases, interpolation may still become unstable.

A more delicate theoretical setting

In the polynomial case, the behavior of the Lebesgue constant is well understood.

In contrast, for EIM, the theoretical analysis becomes much more delicate.

In particular:

- the interpolation space is problem-dependent,
- the interpolation points are selected by a greedy procedure,
- only limited general theoretical results are available for the growth of the associated Lebesgue constant.

Coarse bound

Then for every $g \in X$,

$$\begin{aligned} I_M(g) &= I_{M-1}(g) + (g - I_{M-1}(g))(\mathbf{x}_M) q_M \\ &= I_{M-1}(g) + \alpha_M(g) q_M \end{aligned}$$

$$|\alpha_M(g)| \leq |g(\mathbf{x}_M)| + |I_{M-1} g(\mathbf{x}_M)| \leq \|g\|_{L^\infty} + \|I_{M-1} g\|_{L^\infty}$$

thus,

$$\|I_M(g)\|_{L^\infty} \leq \|g\|_{L^\infty} + 2\|I_{M-1} g\|_{L^\infty}$$

And thus,

$$\Lambda_M \leq 1 + 2\Lambda_{M-1}, \text{ with } \Lambda_1 = 1$$

Connection with EIM

$$\Lambda_M \leq 1 + 2\Lambda_{M-1}, \text{ with } \Lambda_1 = 1$$

yields

$$\Lambda_M \leq 2^M - 1$$

Back to the PDE: recovered affine form

Applying EIM to the coefficient gives

$$A(\mathbf{x};\boldsymbol{\mu}) \approx A_M(\mathbf{x};\boldsymbol{\mu}) = \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu})\, q_m(\mathbf{x}).$$

Hence

$$a(w, v;\boldsymbol{\mu}) = \int_\Omega A(\mathbf{x};\boldsymbol{\mu})\, \nabla w \cdot \nabla v \, d\mathbf{x} \approx \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu})\, a_m(w, v),$$

with

$$a_m(w, v) = \int_\Omega q_m(\mathbf{x})\, \nabla w \cdot \nabla v \, d\mathbf{x}.$$

This is exactly the affine decomposition needed for efficient reduced-order modeling.

Conclusion

EIM replaces a non-affine coefficient by an interpolated separated expansion

$$A_M(\mathbf{x}, \boldsymbol{\mu}) = I_M(A) = \sum_{m=1}^{M} \beta_m(\boldsymbol{\mu}) q_m(\mathbf{x}).$$

The magic points $x_1, \ldots, x_M$ determine the online coefficients through interpolation.

For reduced models, EIM restores an affine decomposition and enables offline/online splitting.

Conclusion

Basic knowledge on EIM but also :

- DEIM
- GEIM ...