A non intrusive reduced basis method for fluid dynamic

Rachida CHAKIR

4^{th} Workshop on Freefem++ and its Applications

IFSTTAR

Laboratoire Instrumentation, Simulation et Informatique Scientifique (LISIS)

In collaborations with

Laboratoire Jacques Louis Lions, UPMC

- Yvon Maday - maday@ann.jussieu.fr
- Pascal Joly - joly@ann.jussieu.fr
- Philippe Parnaudeau - parnaudeau@ann.jussieu.fr

LISIS - IFSTTAR

- Erick Merliot - erick.merliot@ifsttar.fr
- Julien Waeytens - julien.waeytens@ifsttar.fr

Eco-city

ENERGY SAVING

ENERGY EFFICIENTY

Data center hots spots

Needs

- 2D/3D Numerical Modeling
- Fast and reliable methods
- Control of quantity of interest
- Uncertainties quantifications

Challenges

- Multiphysics Modeling
- Non-Linearities and Coupling
- Complex geometries
- Optimization

Reduced basis methods

CONTEXT : optimization process or characterization in real-time of systems governed by a parameters dependent PDEs.

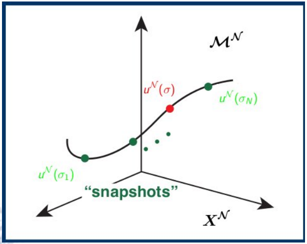
$\rightarrow$ Classicals discretization techniques such as finite element methods are generally too expensives

σ : parameters (boundaries conditions, physicals parameters, ...)

Given σ in $\mathcal{D} \subset \mathbb{R}^d$
$\rightarrow$ Find $u^{\mathcal{N}}(\sigma)$ in $X^{\mathcal{N}}$ s.t

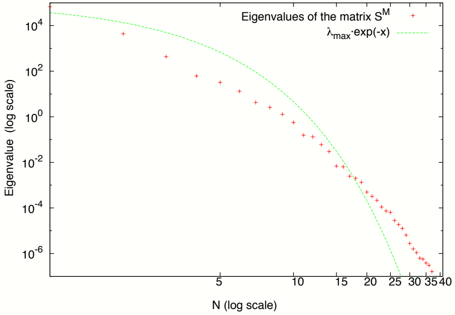
$$a(u^{\mathcal{N}}(\sigma), v; \sigma) = f(v, \sigma)$$

$X^{\mathcal{N}}$: finite-dimensional space



$\rightarrow$ The reduced basis (R.B.) methods exploits the parametric structure of the governing PDE to construct rapidly convergent and computationally efficient approximations.

$\rightarrow$ Assume that $\mathcal{M}^{\mathcal{N}}(\mathcal{D}) = \{u^{\mathcal{N}}(\sigma), \sigma \in \mathcal{D}\}$ has a small (kolmogorov) dimension ...



Evaluation of the dimension of $\mathcal{M}^{\mathcal{N}}(\mathcal{D})$?

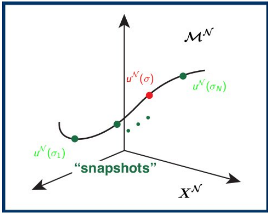
Principal Analysis Component

in appropriate norms :

$S^M_{k,\ell} = \langle u^{\mathcal{N}}(\sigma_k), u^{\mathcal{N}}(\sigma_\ell)\rangle$,

$1 \leq k, \ell \leq M$, M : number of snapshots

Assuming that $\mathcal{M}^{\mathcal{N}}(\mathcal{D}) = \{u^{\mathcal{N}}(\sigma), \sigma \in \mathcal{D}\}$ has a small kolmogorov dimension :



$\rightarrow$ we can select a set of parameters $(\sigma^1, \cdots, \sigma^N)$ in such way that $\mathcal{M}^{\mathcal{N}}(\mathcal{D})$ can be approximated by $W^{\mathcal{N}}_N = \mathrm{span}\{u^{\mathcal{N}}(\sigma^n), 1 \leq n \leq N\}$.

→ The R.B. method is based on the fact that for any $\varepsilon_N > 0$, there exist a set of parameters $(\sigma_1, \cdots, \sigma_N) \in \mathcal{D}^N$ such that :

$$\boxed{\forall \mu \in \mathcal{D}, \exists (\alpha_i(\sigma)) \in \mathbb{R}^N, \quad \|u(\sigma) - \sum_{i=1}^{N} \alpha_i(\sigma)\, u(\sigma^i)\|_{H^1(\Omega)} \leq \varepsilon.}$$

→ The R.B. method is a Galerkin approach within the space $W_N^{\mathcal{N}}$.

THE REDUCED BASIS METHOD	vs	A CLASSICAL DISCRETIZATION METHOD
Find $u_N(\sigma)$ in $W_N^{\mathcal{N}}$ s.t. : $a(u_N(\sigma), v_N; \sigma) = (f, v_N)$, $\forall v_N \in W_N^{\mathcal{N}}$ $\Rightarrow \mathcal{O}(N)$		Find $u^{\mathcal{N}}(\sigma)$ in $X^{\mathcal{N}}$ s.t : $a(u^{\mathcal{N}}(\sigma), v^h; \sigma) = (f, v^h)$. $\forall v^h \in X^{\mathcal{N}}$ $\Rightarrow \mathcal{O}(\mathcal{N})$

→The reduced basis is promising if N is small ! $(N << \mathcal{N})$

Requirements of the reduced basis method :

- How to select the good sampling set $(\sigma^1, \cdots, \sigma^N)$?
  - $\rightarrow$ Random
  - $\rightarrow$ P.O.D
  - $\rightarrow$ Greedy's algorithm

Algorithm 1 Example of a Greedy's algorithm

Given $\Xi_{train} = (\sigma_1, \cdots, \sigma_{n_{train}}) \in \mathcal{D}^{n_{train}}$, $n_{train} >> 1$
Choose randomly σ_1, $\rightarrow S_1 = \{\sigma_1\}$ and $W_1^{\mathcal{N}} = \{u^{\mathcal{N}}(\sigma_1)\}$
for $N = 2$ to N_{max} **do**
$\quad \sigma_N = \arg\max_{\sigma \in \Xi_{train}} \|u^{\mathcal{N}}(\sigma) - u^h_{N-1}(\sigma)\|_X$
$\quad S_N = S_{N-1} \cup \sigma_N$ and $W_N^{\mathcal{N}} = W_{N-1}^{\mathcal{N}} + \mathrm{span}\{u^{\mathcal{N}}(\sigma_N)\}$
end for

$\rightarrow$This version of the Greedy's algorithm is quite expensive !

Requirements of the reduced basis method :

- How to select the good sampling set $(\sigma^1, \cdots, \sigma^N)$?
  $\rightarrow$ Random
  $\rightarrow$ P.O.D
  $\rightarrow$ Greedy's algorithm

Algorithm 2 Example of a Greedy's algorithm

Given $\Xi_{train} = (\sigma_1, \cdots, \sigma_{n_{train}}) \in \mathcal{D}^{n_{train}}$, $n_{train} >> 1$
Choose randomly σ_1, $\rightarrow S_1 = \{\sigma_1\}$ and $W_1^{\mathcal{N}} = \{u^{\mathcal{N}}(\sigma_1)\}$
for $N = 2$ to N_{max} **do**
  $\sigma_N = \arg \max_{\sigma \in \Xi_{train}} \Delta_{N-1}(\sigma)$
  $S_N = S_{N-1} \cup \sigma_N$ and $W_N^{\mathcal{N}} = W_{N-1}^{\mathcal{N}} + \mathrm{span}\{u^{\mathcal{N}}(\sigma_N)\}$
end for

$\Delta_N(\sigma)$: sharp, inexpensive *a posteriori* error bound of $\|u^{\mathcal{N}}(\sigma) - u_N^h(\sigma)\|_X$
$\rightarrow$ Only the actual $u^{\mathcal{N}}(\sigma_N)$ are computed by the Greedy's algorithm.

Requirements of the reduced basis method :

- How to select the good set of $(\sigma^1, \cdots, \sigma^N)$?
  $\rightarrow$ Random
  $\rightarrow$ P.O.D
  $\rightarrow$ Greedy algorithm

- How to actually computes the reduced solution $u_N(\sigma)$ for a given σ ?
  $\rightarrow$ Get the classical solution $(u^{\mathcal{N}}(\sigma^n))_{1\leq n\leq N}$ (for example using a FEM code), from which the orthogonal basis function $(\xi_1^{RB}, \cdots, \xi_N^{RB})$ of $W_N^{\mathcal{N}}$ will be computed.

  For each new value of σ :
  $\rightarrow$ build the matrix $[A^N(\sigma)]_{k,\ell} = a(\xi_k^{RB}, \xi_\ell^{RB}; \sigma)_{1\leq k,\ell\leq N}$ and the vector $[F^N(\sigma)]_\ell = f(\xi_\ell^{RB}; \sigma)_{1\leq \ell\leq N}$

Requirements of the reduced basis method :

- How to select the good set of $(\sigma^1, \cdots, \sigma^N)$?
  $\rightarrow$ Random
  $\rightarrow$ P.O.D
  $\rightarrow$ Greedy algorithm
- How to actually computes the reduced solution $u_N(\sigma)$ for a given σ ?
  $\rightarrow$ Get the classical solution $(u^{\mathcal{N}}(\sigma^n))_{1\leq n\leq N}$ (for example using a FEM code), from which the orthogonal basis function $(\xi_1^{RB}, \cdots, \xi_N^{RB})$ of $W_N^{\mathcal{N}}$ will be computed.

  For each new value of σ :
  $\rightarrow$ build the matrix $[A^N(\sigma)]_{k,\ell} = a(\xi_k^{RB}, \xi_\ell^{RB}; \sigma)_{1\leq k,\ell\leq N}$ and the vector $[F^N(\sigma)]_\ell = f(\xi_\ell^{RB}; \sigma)_{1\leq \ell\leq N}$
  $\rightarrow$ solve the system $A^N(\sigma)\,\alpha^{N,h}(\sigma) = F^N(\sigma)$ and build output :

$$s(u_h^N(\sigma)) = \sum_{\ell=1}^{N} \alpha_\ell^{N,h}(\sigma)\, s(\xi_\ell^{BR})$$

Requirements of the reduced basis method :

- How to select the good set of $(\sigma^1, \cdots, \sigma^N)$?
  $\rightarrow$ Random
  $\rightarrow$ P.O.D
  $\rightarrow$ Greedy algorithm

- How to actually computes the reduced solution $u_N(\sigma)$ for a given σ ?
  $\rightarrow$ Get the classical solution $(u^{\mathcal{N}}(\sigma^n))_{1\leq n\leq N}$ (for example using a FEM code), from which the orthogonal basis function $(\xi_1^{RB}, \cdots, \xi_N^{RB})$ of $W_N^{\mathcal{N}}$ will be computed.

  For each new value of σ :
  $\rightarrow$ build the matrix $[A^N(\sigma)]_{k,\ell} = a(\xi_k^{RB}, \xi_\ell^{RB}; \sigma)_{1\leq k,\ell\leq N}$ and the vector $[F^N(\sigma)]_\ell = f(\xi_\ell^{RB}; \sigma)_{1\leq \ell\leq N}$
  $\rightarrow$ solve the system $A^N(\sigma)\,\alpha^{N,h}(\sigma) = F^N(\sigma)$ and build output :

$$s(u_h^N(\sigma)) = \sum_{\ell=1}^{N} \alpha_\ell^{N,h}(\sigma)\, s(\xi_\ell^{BR})$$

Requirements of the reduced basis method :

- How to select the good set of $(\sigma^1, \cdots, \sigma^N)$?
  $\rightarrow$ Random
  $\rightarrow$ P.O.D
  $\rightarrow$ Greedy algorithm
- How to actually computes the reduced solution $u_N(\sigma)$ for a given σ ?
  $\rightarrow$ Get the classical solution $(u^{\mathcal{N}}(\sigma^n))_{1\leq n\leq N}$ (for example using a FEM code), from which the orthogonal basis function $(\xi_1^{RB}, \cdots, \xi_N^{RB})$ of $W_N^{\mathcal{N}}$ will be computed.

  For each new value of σ :
  $\rightarrow$ build the matrix $[A^N(\sigma)]_{k,\ell} = a(\xi_k^{RB}, \xi_\ell^{RB}; \sigma)_{1\leq k,\ell\leq N}$ and the vector $[F^N(\sigma)]_\ell = f(\xi_\ell^{RB}; \sigma)_{1\leq \ell\leq N}$
  $\rightarrow$ solve the system $A^N(\sigma)\,\alpha^{N,h}(\sigma) = F^N(\sigma)$ and build output :

$$s(u_h^N(\sigma)) = \sum_{\ell=1}^{N} \alpha_\ell^{N,h}(\sigma)\, s(\xi_\ell^{BR})$$

Requirements of the reduced basis method :

- How to select the good set of $(\sigma^1, \cdots, \sigma^N)$? (OFFLINE)
  $\rightarrow$ Random
  $\rightarrow$ P.O.D
  $\rightarrow$ Greedy algorithm
- How to actually computes the reduced solution $u_N(\sigma)$ for a given σ ?
  $\rightarrow$ Get the classical solution $(u^{\mathcal{N}}(\sigma^n))_{1\leq n\leq N}$ (for example using a FEM code), from which the orthogonal basis function $(\xi_1^{RB}, \cdots, \xi_N^{RB})$ of $W_N^{\mathcal{N}}$ will be computed. (OFFLINE)

  For each new value of σ :
  $\rightarrow$ build the matrix $[A^N(\sigma)]_{k,\ell} = a(\xi_k^{RB}, \xi_\ell^{RB}; \sigma)_{1\leq k,\ell\leq N}$ and the vector $[F^N(\sigma)]_\ell = f(\xi_\ell^{RB}; \sigma)_{1\leq \ell\leq N}$ (OFFLINE + ONLINE)
  $\rightarrow$ solve the system $A^N(\sigma)\,\alpha^{N,h}(\sigma) = F^N(\sigma)$ and build output :

$$s(u_h^N(\sigma)) = \sum_{\ell=1}^{N} \alpha_\ell^{N,h}(\sigma)\, s(\xi_\ell^{BR})$$ (ONLINE)

$\rightarrow$ One of the keys of the R.B method is the decomposition of the computational work into an OFFLINE and an ONLINE stage

Requirements of the reduced basis method :

- How to select the good set of $(\sigma^1, \cdots, \sigma^N)$? (OFFLINE)

  $\rightarrow$ Random

  $\rightarrow$ P.O.D

  $\rightarrow$ Greedy algorithm

- How to actually computes the reduced solution $u_N(\sigma)$ for a given σ ?

  $\rightarrow$ Get the classical solution $(u^{\mathcal{N}}(\sigma^n))_{1\leq n\leq N}$ (for example using a FEM code), from which the orthogonal basis function $(\xi_1^{RB}, \cdots, \xi_N^{RB})$ of $W_N^{\mathcal{N}}$ will be computed. (OFFLINE)

  For each new value of σ :

  $\rightarrow$ build the matrix $[A^N(\sigma)]_{k,\ell} = a(\xi_k^{RB}, \xi_\ell^{RB}; \sigma)_{1\leq k,\ell\leq N}$ and the vector $[F^N(\sigma)]_\ell = f(\xi_\ell^{RB}; \sigma)_{1\leq \ell\leq N}$ (OFFLINE + ONLINE)

  $\rightarrow$ solve the system $A^N(\sigma)\,\alpha^{N,h}(\sigma) = F^N(\sigma)$ and build output :

  $$s(u_h^N(\sigma)) = \sum_{\ell=1}^{N} \alpha_\ell^{N,h}(\sigma)\, s(\xi_\ell^{BR})$$ (ONLINE)

  $\rightarrow$ One of the keys of the R.B method is the decomposition of the computational work into an OFFLINE and an ONLINE stage

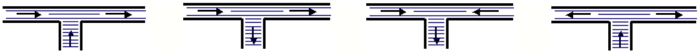


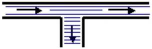
FIGURE: examples of T -junctions

An adimensioned problem

Find $\mathbf{u} \in (H^1(\Omega))^d$ and $p \in L^2(\Omega)$ such that :

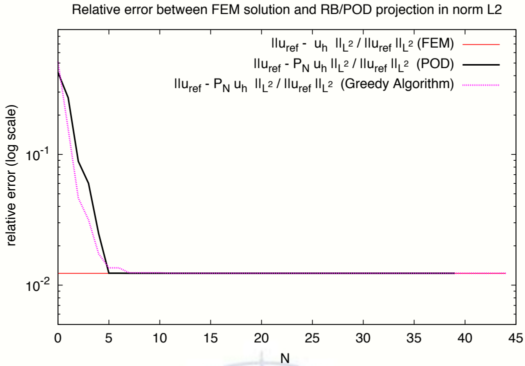
$$\left\{\begin{array}{rcl} (\mathbf{u}.\nabla)\mathbf{u} + \nabla p - \dfrac{1}{\mathrm{Re}}\Delta\mathbf{u} & = & \mathbf{0}, \\ \mathrm{div}\ \mathbf{u} & = & 0, \\ \dfrac{1}{\mathrm{Re}}\dfrac{\partial \mathrm{u}_n}{\partial n} - p = 1, \qquad \mathrm{u}_\tau & = & 0 \text{ on } \Gamma_{in}, \\ & & (\mathrm{u}_n, \mathrm{u}_\tau) : \text{normal and tangential component of } \mathbf{u}\,; \\ \dfrac{1}{\mathrm{Re}}\dfrac{\partial \mathrm{u}_n}{\partial n} - p = 0, \qquad \mathrm{u}_\tau & = & 0 \text{ on } \Gamma_{out}, \\ \mathbf{u} & = & \mathbf{0} \text{ on } \Gamma_{wall}, \end{array}\right. \tag{1}$$

with $\sigma = \mathrm{Re} = \dfrac{\rho D\, v_{in}}{\mu}$, D : pipe's diameter, v_{in} : inlet velocity, ρ : fluid's density and and μ : dynamic viscosity.



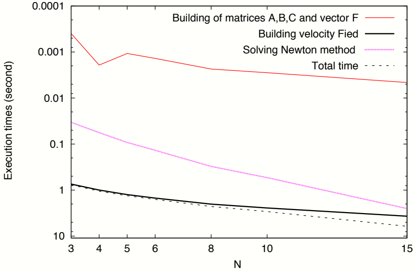
Selection of the set of $(\sigma^1, \cdots, \sigma^N)$ and construction of the reduced basis

$Re \in [10, 500]$ and $d = 2$. P_N : L^2-projection into $W_N^{\mathcal{N}}$

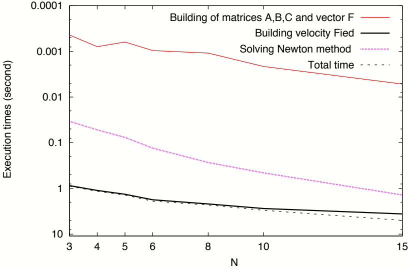


$u_h \rightarrow \mathbb{P}_2$-FE solution Ndof $= 34313$, $u_{\mathrm{ref}} \rightarrow \mathbb{P}_2$-FE solution Ndof $= 136145$

Time computation during the *online* stage of the reduced basis method

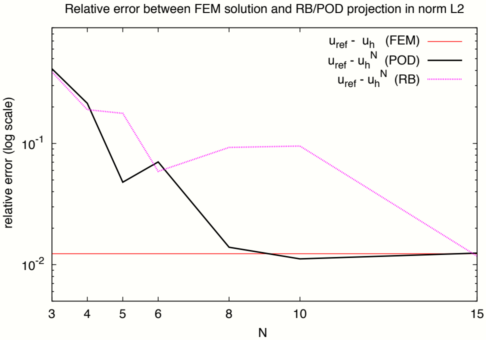


POD



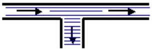
Greedy algorithm

Comparaison between FEM and RB method for $Re = 400$



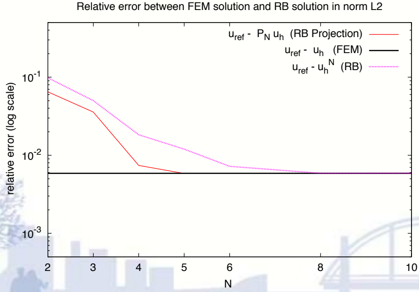
<table>
<tr><td rowspan="2">Finite element method</td><td colspan="2">online stage of the RB method</td></tr>
<tr><td>POD</td><td>Greedy</td></tr>
<tr><td>640</td><td>2.2</td><td>4.2</td></tr>
</table>

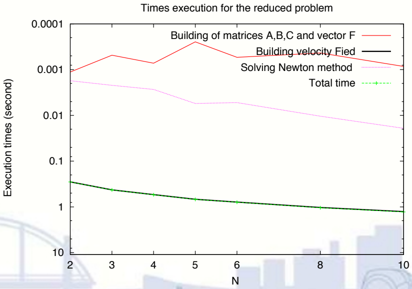
Time in seconds



An example with varying boundaries conditions $\sigma = v_{in} \in [0.01; 0.5] cm.s^{-1}$

$$\left\{\begin{array}{rcl} \rho(\mathbf{u}.\nabla)\mathbf{u} + \nabla p - \mu\Delta\mathbf{u} & = & 0, \\ \mathrm{div}(\mathbf{u}) & = & 0, \\ \mathbf{u} & = & v_{in} * (f(\mathbf{x}) \cdot \overrightarrow{n}) \text{ on } \Gamma_{in}, \\ \dfrac{\partial \mathbf{u}}{\partial n} & = & 0 \text{ on } \Gamma_{out}, \\ \mathbf{u} & = & 0 \text{ on } \Gamma_{wall}, \end{array}\right.$$





Requirements of the reduced basis method :

$\boxed{\text{How } A^N(\sigma) \text{ is generated ?}}$

Direct affine's decomposition

$$a(\xi_i^{RB}, \xi_j^{RB}; \sigma) = \sum_{k=1}^{\mathcal{K}} \theta_k(\sigma)\, a_k(\xi_i^{RB}, \xi_j^{RB})$$

Empirical interpolation method

$$a(\xi_i^{RB}, \xi_j^{RB}; \sigma) = \sum_{k=1}^{\mathcal{K}} \Phi_k(\sigma)\, a(\xi_i^{RB}, \xi_j^{RB}; q_k)$$

OFFLINE : $a_k(\xi_i^{RB}, \xi_j^{RB})$ (or $a(\xi_i^{RB}, \xi_j^{RB}; q_k)$) are precomputed

ONLINE :
- $A^N(\sigma)$ generation's requires only $\mathcal{K} \times N^2$ operations instead of $\mathcal{N}^2$.
- $A^N(\sigma)$ inversion's is done in N^3 operations instead of $\mathcal{N}^3$.(direct inversion)

What happens when the FEM simulation code is used as black box ?

$\rightarrow$ It's not possible to use this code to perform all the OFFLINE computations required for an efficient performance of the R.B method

(Since we want the online computation to be done with a N complexity and not with a complexity of the finite element method)

$\rightarrow$ An alternative : a non intrusive reduced basis methodS

blanc ici

Requirements of the reduced basis method :

How $A^N(\sigma)$ is generated ?

Direct affine's decomposition

$$a(\xi_i^{RB}, \xi_j^{RB}; \sigma) = \sum_{k=1}^{\mathcal{K}} \theta_k(\sigma)\, a_k(\xi_i^{RB}, \xi_j^{RB})$$

Empirical interpolation method

$$a(\xi_i^{RB}, \xi_j^{RB}; \sigma) = \sum_{k=1}^{\mathcal{K}} \Phi_k(\sigma)\, a(\xi_i^{RB}, \xi_j^{RB}; q_k)$$

OFFLINE : $a_k(\xi_i^{RB}, \xi_j^{RB})$ (or $a(\xi_i^{RB}, \xi_j^{RB}; q_k)$) are precomputed

ONLINE :
- $A^N(\sigma)$ generation's requires only $\mathcal{K} \times N^2$ operations instead of $\mathcal{N}^2$.
- $A^N(\sigma)$ inversion's is done in N^3 operations instead of $\mathcal{N}^3$.(direct inversion)

What happens when the FEM simulation code is used as black box ?

→ It's not possible to use this code to perform all the OFFLINE computations required for an efficient performance of the R.B method

(since we want the online computation to be done with a N complexity and not with a complexity of the finite element method)

→ An alternative : a non intrusive reduced basis methodS

blanc ici

A non intrusive reduced basis method : How ?

Let $\tilde{u}_h^N(\sigma)$ be the L^2-projection of $u^{\mathcal{N}}(\sigma)$ in $W_N^{\mathcal{N}}$ defined by

$$\boxed{\tilde{u}_h^N(\sigma) = \sum_{i=1}^{N} \beta_i^{N,h}(\sigma)\,\xi_i^{RB} \qquad \text{with} \qquad \beta_i^{N,h}(\sigma) = \int_\Omega u^{\mathcal{N}}(\sigma)\,\xi_i^{RB}}$$

$\longrightarrow$ The standard R.B. method aims at evaluating the coefficients $\alpha_i^{N,h}(\sigma)$ those can appear as a substitute to the optimal coefficients $\beta_i^h(\sigma)$.

Since, the computation of $u^{\mathcal{N}_H}(\sigma)$, for $H >> h$ and $X_{\mathcal{N}_H} \subset X_{\mathcal{N}}$, is less expensive than the one of $u^{\mathcal{N}}(\sigma)$.

$\longrightarrow$ Our alternative method [1,2] consists in proposing an another surrogate to $\beta_i^{N,h}(\sigma)$ defined by

$$\boxed{\beta_i^{N,H}(\sigma) = \int_\Omega u^{\mathcal{N}_H}(\sigma)\,\xi_i^{RB}}$$

[1] R. Chakir Y. Maday, *A two-grid finite-element/reduced basis scheme for the approximation of the solution of parameter dependent P.D.E.* Actes de congrès du 9ème colloque national en calcul des structrures, Giens 2009.
[2] R. Chakir, Y. Maday, *Une méthode combinée d'éléments finis à deux grilles/bases réduites pour l'approximation des solutions d'une EDP paramétrique*, C. R. Acad. Sci. Paris, Ser. I 347 (2009) 435 – 440.

A non intrusive reduced basis method : How ?

Let $\tilde{u}_h^N(\sigma)$ be the L^2-projection of $u^{\mathcal{N}}(\sigma)$ in $W_N^{\mathcal{N}}$ defined by

$$\boxed{\tilde{u}_h^N(\sigma) = \sum_{i=1}^{N} \beta_i^{N,h}(\sigma)\, \xi_i^{RB} \qquad \text{with} \qquad \beta_i^{N,h}(\sigma) = \int_\Omega u^{\mathcal{N}}(\sigma)\, \xi_i^{RB}}$$

$\longrightarrow$ The standard R.B. method aims at evaluating the coefficients $\alpha_i^{N,h}(\sigma)$ those can appear as a substitute to the optimal coefficients $\beta_i^h(\sigma)$.

Since, the computation of $u^{\mathcal{N}_H}(\sigma)$, for $H >> h$ and $X_{\mathcal{N}_H} \subset X_{\mathcal{N}}$, is less expensive than the one of $u^{\mathcal{N}}(\sigma)$.

$\longrightarrow$ Our alternative method [1,2] consists in proposing an another surrogate to $\beta_i^{N,h}(\sigma)$ defined by

$$\boxed{\beta_i^{N,H}(\sigma) = \int_\Omega u^{\mathcal{N}_H}(\sigma)\, \xi_i^{RB}}$$

[1] R. Chakir Y. Maday, *A two-grid finite-element/reduced basis scheme for the approximation of the solution of parameter dependent P.D.E*, Actes de congrès du 9ème colloque national en calcul des structures, Giens 2009.
[2] R. Chakir, Y. Maday, *Une méthode combinée d'éléments finis à deux grilles/bases réduites pour l'approximation des solutions d'une EDP. paramétrique*, C. R. Acad. Sci. Paris, Ser. I 347 (2009) 435 - 440.

We can build a reduced solution $u^N_{H,h}(\sigma)$ and the output $s(u^N_{H,h}(\sigma))$:

$$\boxed{u^N_{H,h}(\sigma) = \sum_{i=1}^{N} \beta_i^{N,H}(\sigma)\,\xi_i^{RB}} \quad \text{and} \quad \boxed{s(u^N_{H,h}(\sigma)) = \sum_{i=1}^{N} \beta_i^{N,H}(\sigma)\,s(\xi_i^{RB})}$$

$\rightarrow$ This method is based on the fact that the error measured in the L^2-norm converge faster than the one measured H^1-norm.

Why this can still be a good approximation ?

$\longrightarrow$ The basis functions ξ_i^{RB} have to be orthonormal in H^1 and L^2 norm.

$X^{\mathcal{N}}$ and $X^{\mathcal{N}}_H$: $\mathbb{P}^k$- F.E discretization space $\to \|u(\sigma) - u^{\mathcal{N}}(\sigma)\|_X \le c(\sigma)\,\mathrm{h}^{\mathrm{k}}$

$\longrightarrow$ Using the orthogonality of ξ_i^{BR}, we easily can prove that :

$$\boxed{\|u(\sigma) - u^N_{H,h}(\sigma)\|_X \le \varepsilon + C(\sigma)\,(\mathrm{h}^{\mathrm{k}} + \mathrm{H}^{2\mathrm{k}})}$$

which is asymptotically similar to $\|u(\sigma) - u^{\mathcal{N}}(\sigma)\|_X \le \varepsilon + C(\sigma)\,\mathrm{h}^{\mathrm{k}}$ when we choose $h \sim H^2$.

We can build a reduced solution $u^N_{H,h}(\sigma)$ and the output $s(u^N_{H,h}(\sigma))$:

$$\boxed{u^N_{H,h}(\sigma) = \sum_{i=1}^{N} \beta_i^{N,H}(\sigma)\,\xi_i^{RB}} \text{ and } \boxed{s(u^N_{H,h}(\sigma)) = \sum_{i=1}^{N} \beta_i^{N,H}(\sigma)\,s(\xi_i^{RB})}$$

$\rightarrow$ This method is based on the fact that the error measured in the L^2-norm converge faster than the one measured H^1-norm.

Why this can still be a good approximation ?
$\longrightarrow$ The basis functions ξ_i^{RB} have to be orthonormal in H^1 and L^2 norm.

$X^{\mathcal{N}}$ and $X_H^{\mathcal{N}}$: $\mathbb{P}^k$- F.E discretization space $\rightarrow \|u(\sigma) - u^{\mathcal{N}}(\sigma)\|_X \leq c(\sigma)\,\mathrm{h}^{\mathrm{k}}$
$\longrightarrow$ Using the orthogonality of ξ_i^{BR}, we easily can prove that :

$$\boxed{\|u(\sigma) - u^N_{H,h}(\sigma)\|_X \leq \varepsilon + C(\sigma)\,(\mathrm{h}^{\mathrm{k}} + \mathrm{H}^{2\mathrm{k}})}$$

which is asymptotically similar to $\|u(\sigma) - u^{\mathcal{N}}(\sigma)\|_X \leq \varepsilon + C(\sigma)\,\mathrm{h}^{\mathrm{k}}$ when we choose $h \sim H^2$.

Rectification method to improve the computation of the $\beta_i^{N,H}(\sigma)$

We computes the matrix $T^N \in \mathbb{R}^{N\times N}$ solution of the following system :

$$T^N \times \begin{pmatrix} \beta_1^{N,H}(\sigma_1) & \cdots & \beta_1^{N,H}(\sigma_N) \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ \beta_N^{N,H}(\sigma_1) & \cdots & \beta_N^{N,H}(\sigma_N) \end{pmatrix} = \begin{pmatrix} \beta_1^{N,h}(\sigma_1) & \cdots & \beta_1^{N,h}(\sigma_N) \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \\ \beta_N^{N,h}(\sigma_1) & \cdots & \beta_N^{N,h}(\sigma_N) \end{pmatrix}$$

$\Rightarrow$ We replace $u_{H,h}^N(\sigma)$ and $s(u_{H,h}^N(\sigma))$:

$$\boxed{\tilde{u}_{H,h}^N(\sigma) = \sum_{i=1}^{N} T_{ij}^N\, \beta_i^{N,H}(\sigma)\, \xi_i^{RB}} \quad \text{and} \quad \boxed{s(\tilde{u}_{H,h}^N(\sigma)) = \sum_{i,j=1}^{N} T_{ij}^N\, \beta_i^{N,H}(\sigma)\, s(\xi_i^{RB})}$$

What do we need ?

F.E. code used as *black box*

Compute
snapshots $u_h(\sigma_i)$
coarse solution $u_H(\sigma)$

Return
fine mesh $\mathcal{T}_h$
coarse mesh $\mathcal{T}_H$

F.E. library

(Freefem++)

To compute
L^2 and H^1
scalar product

Interpolate
from $\mathcal{T}_H$
to $\mathcal{T}_h$

Implementation

Offline stage

1. Construction of a reduced approximation's space.
   - computation of a sample of solutions (black box software)
   - selection of N solutions to build the reduced basis (F.E. Library).
2. Orthonormalisation in L^2 and H^1-norm of the reduced basis functions (F.E. Library).
3. Preparation for the rectification.
   - computation of the N coarse solutions $u^{\mathcal{N}_H}(\sigma_i)$ (black box software)
   - construction of matrix T^N (F.E. Library).

Online stage

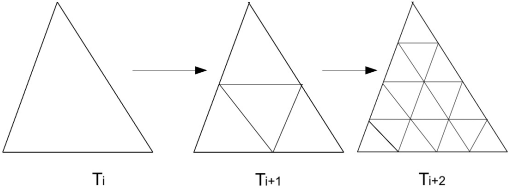
1. Computation of the coarse solution $u^{\mathcal{N}_H}(\sigma)$.(black box software)
2. Compute the coefficient $\beta_i^{N,H}(\sigma)$. (F.E. Library)
3. Apply the rectification on the $\beta_i^{N,H}(\sigma)$. (F.E. Library)
4. Build the output $s(u_N^{H,h}(\sigma))$. (F.E. Library)

A convection dominated problem : find $u \in H^1(\Omega)$ such that

$$\boxed{\begin{array}{ll} -(0.01)\Delta u + v \cdot \nabla u = 0 & \text{in } \Omega = [0,1]^2 \\ u = x^2 & \text{on } \Gamma_1 = \{(1,y), y \in [0,1]\} \\ u = y^2 & \text{on } \Gamma_2 = \{(x,1), x \in [0,1]\} \\ u = 0 & \text{on } \Gamma_3 = \partial\Omega \setminus (\Gamma_1 \cup \Gamma_2). \end{array}}$$

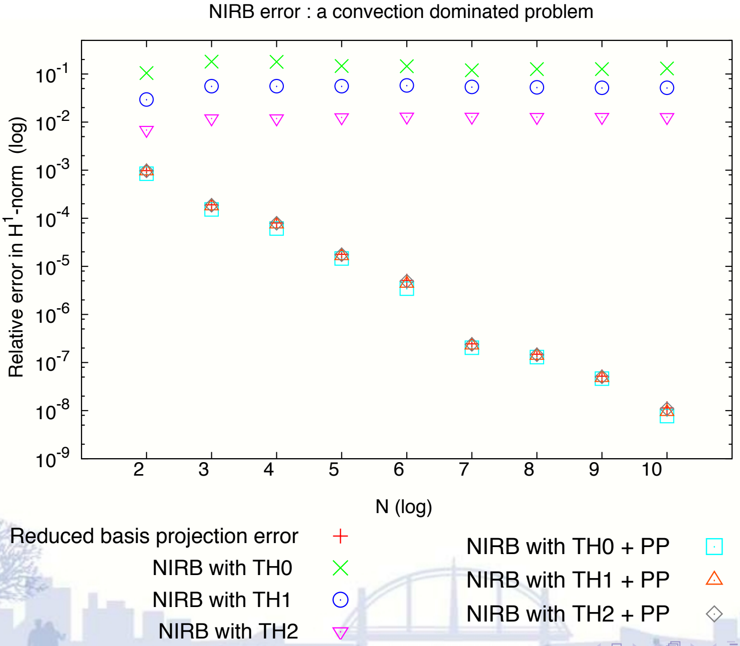
where $v = (\cos\,\sigma,\ \sin\,\sigma)$ and σ : angle of the convection flux in $[0, \frac{\pi}{2}]$.

- Construction of the meshes

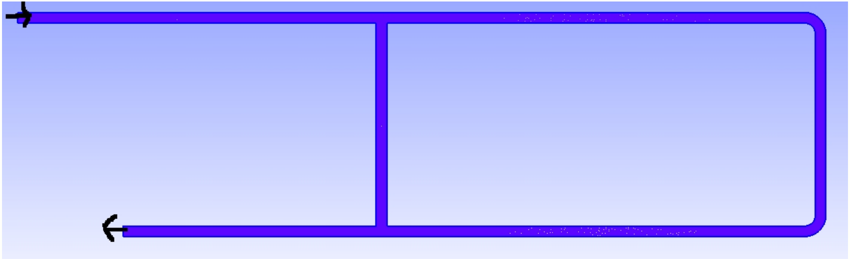


From an original coarse triangulation $\mathcal{T}_{H_0}$, we built successive refined triangulations by recursively splitting each triangle K of $\mathcal{T}_{H_{i-1}}$ into four triangles with equal diameter H_i^K such that $H_i^K = \frac{H_{(i-1)}^K}{2}$.

$\longrightarrow$ We get a superspace $X^{\mathcal{N}_{H_i}}$ about four times larger than $X^{\mathcal{N}_{H_{i-1}}}$ that satisfies $X^{\mathcal{N}_{H_{i-1}}} \subset X^{\mathcal{N}_{H_i}}$.


NIRB error : a convection dominated problem
Relative error in H1-norm (log)
10-1
10-2
10-3
10-4
10-5
10-6
10-7
10-8
10-9
2
3
4
5
6
7
8
9
10
N (log)
Reduced basis projection error
NIRB with TH0
NIRB with TH1
NIRB with TH2
NIRB with TH0 + PP
NIRB with TH1 + PP
NIRB with TH2 + PP

Small water network with low Reynolds flows - Incompressible steady Navier Stokes equations



$$\boxed{\begin{array}{cl} -\nu\Delta u + u\cdot\nabla u + \frac{1}{\rho}\nabla p = 0 & \text{in } \Omega \\ div u = 0 & \text{in } \Omega \\ u = u_{In} & \text{on } \Gamma_{In} \\ \frac{\partial u}{\partial n} = 0 & \text{on } \Gamma_{Out} \\ u = 0 & \text{on } \Gamma_{wall}. \end{array}}$$

The parameter "σ" is the inlet velocity $u_{In} \in [0.1;5]\,\mathrm{mm.s}^{-1}$.

(which correspond to a Reynolds number varying between 10 and 500)

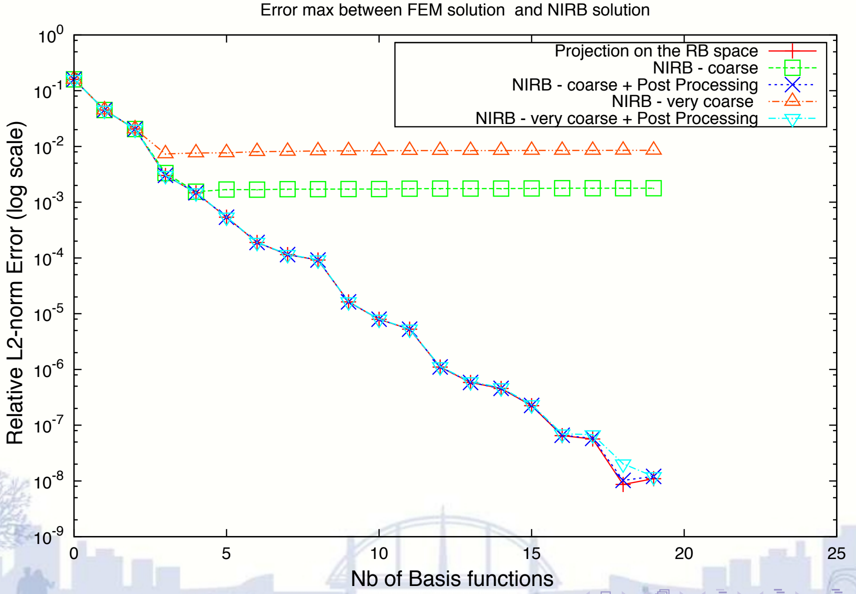
$\mathbb{P}_2/\mathbb{P}_1$ Finite Element calculation done with Freefem++

Velocity magnetude's error map between fine FEM and supercoarse FEM solutions

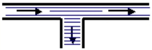
Number of degree of freedom :

- FINE mesh : 83320
- COARSE mesh : 14060

$$\|u_{ref} - u_{coarse}\|_{L^2} / \|u_{ref}\|_{L^2} = 0.004 \qquad \|u_{ref} - u_{very\ coarse}\|_{L^2} / \|u_{ref}\|_{L^2} = 0.014$$



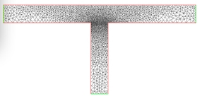
Application to a T-junction with low Reynolds



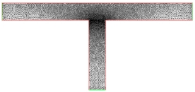
The parameter is the inlet velocity flow $\sigma = v_{in} \in [0.01; 0.5] cm.s^{-1}$

$$\left\{ \begin{array}{rcl} \rho(\mathbf{u}.\nabla)\mathbf{u} + \nabla p - \mu \Delta \mathbf{u} & = & 0, \\ \operatorname{div}(\mathbf{u}) & = & 0, \\ \mathbf{u} & = & v_{in} * (f(\mathbf{x}) \cdot \overrightarrow{n}) \text{ on } \Gamma_{in}, \\ \dfrac{\partial \mathbf{u}}{\partial n} & = & 0 \text{ on } \Gamma_{out}, \\ \mathbf{u} & = & 0 \text{ on } \Gamma_{wall}, \end{array} \right.$$

Application to a T-junction with low Reynolds

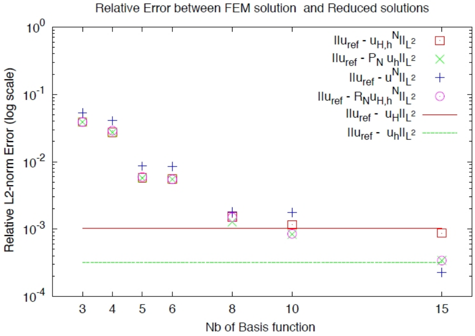


Coarse Mesh
Ndof $\mathbb{P}_2$ = 8717



Fine Mesh
Ndof $\mathbb{P}_2$ = 34313

Reference Mesh :
Ndof $\mathbb{P}_2$ = 136145



Conclusion

We note that the rectification improved even more the approximation since it allows to recover the truth error even starting from the computations of the coarsest NIRB solution.

Perspectives

- Apply to 3D model
- Apply to more complex flow using Code_Saturne as a blackbox
- Take geometry as a parameter