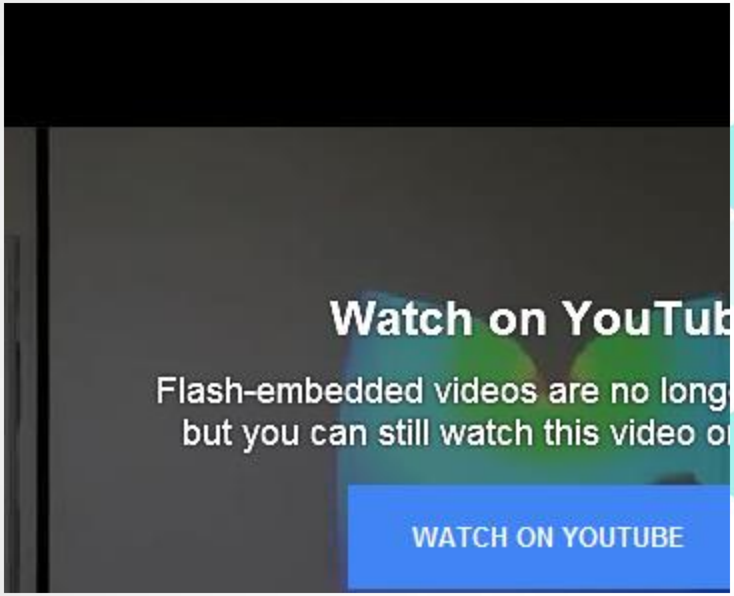
Structural Wing Model Reduction in Fluid-Structure Interactions



[Amsallem, International Journal for Numerical Methods in Engineering, 2014]

STUDENT:

Oriol CHANDRE VILA

TUTORS:

Joseph MORLIER and Sylvain DUBREUIL (ONERA)

Project of Innovation and Research (PIR)

Spring 2018

ISAE SUPAERO

ONERA THE FRENCH AEROSPACE LAB

Table of Contents

Introduction

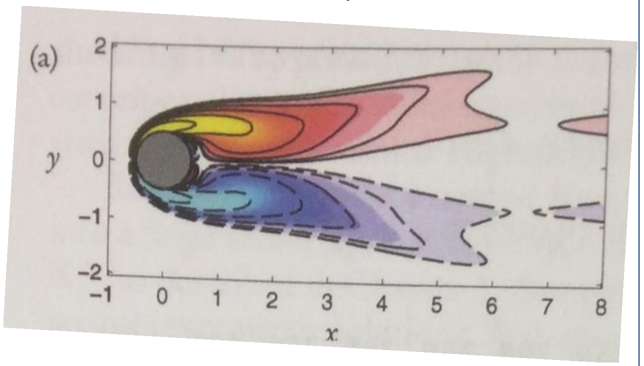
Aeroelasticity

Coupling between aerodynamic forces, inertial forces and elastic forces.

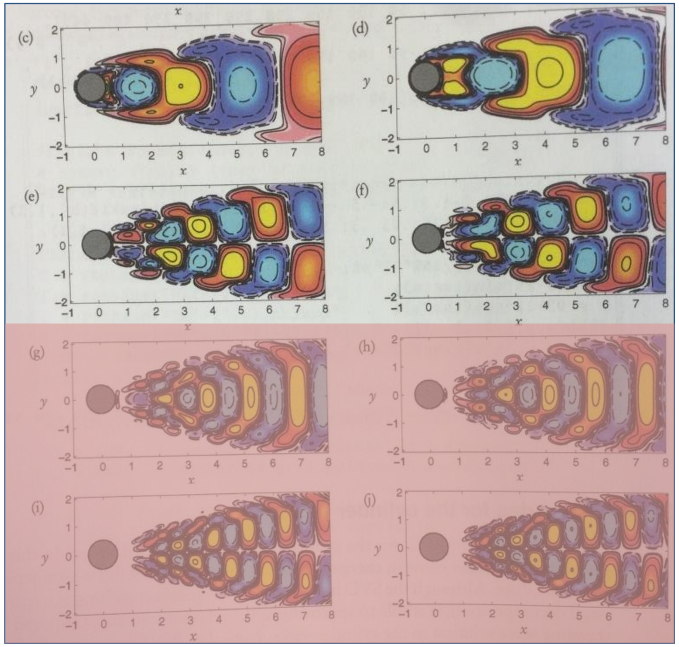
Current issues due to models dimension. Why the ROM?

The calculations have a high cost in time and computational power.

Mean vorticity field



[Kutz et al., Chpt 9, SIAM, 2016]



[Kutz et al., Chpt 9, SIAM, 2016)

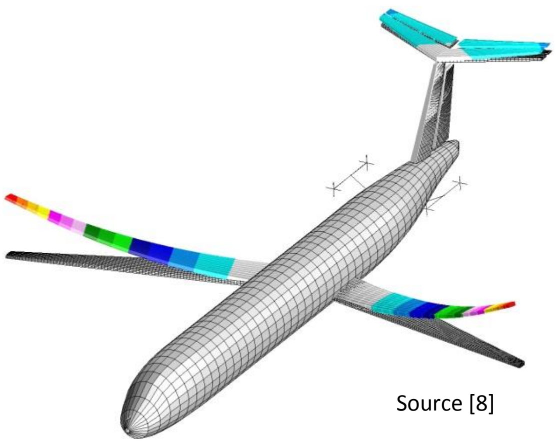
Aim & Objectives

Aim

To study the use of Reduced Order Models (hereafter ROM) in order **to lighten** the computational requirement in fluid-structure problems simulations

Objectives

- To study different ROM methodologies.
- To apply ROM techniques to a simple problem.
- To define a strategy for fluid-structure interactions problems.



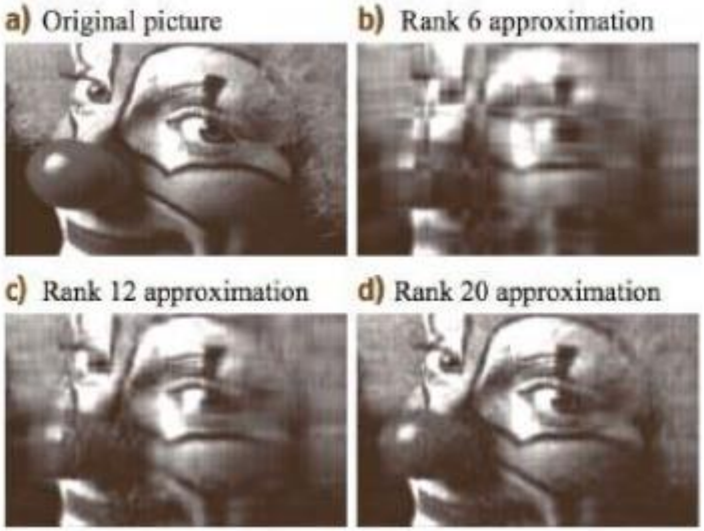
Source [8]

State of the Art: ROM

Principal Component Analysis (PCA) [Kutz, Chpt 15, Oxford University Press, 2013]

Possible correlated variables (Observations) → Orthogonal Transformation → Values of linearly uncorrelated variables (Principal Components)

- Widely used in the actuality.
- Limitations:
  - The results depend on the scaling variables.
  - The applicability is limited by certain assumptions.
  - No differentiation between classes.
- Used in different disciplines:
  - SVD in Algebra
  - POD in Mechanical Engineering.



SVD approach: $A \approx U\Sigma V^*$

In Matlab:

```
[U,S,V] = svd(A)
[Ar] = U(:,1:k)*S(1:k,1:k)*V(:,1:k)'
```

State of the Art: ROM

Dynamic Mode Decomposition (DMD)

[Kutz, SIAM, 2016]
[Kutz, Chpt 20, Oxford University Press, 2013]

AIM:

To take advantage of the low dimensionality in the experimental data.

WHAT DOES DMD PROVIDE?

A decomposition of experimental data into a set of dynamic modes.

HOW DOES IT WORK?

DMD computes the eigenvalues and eigenvectors of the linear model.

ADVANTAGES:

- No equations are needed.
- The future state is known for all time.

WHEN DOES IT FAIL?

- Data matrix is full rank
- No suitable low dimensional structure.

DMD is not a ROM itself, it is more a strategy. However, it uses ROM (usually SVD) to perform a low-rank truncation of the data.

State of the Art: ROM

Proper Generalized Decomposition (PGD) [Chinesta et al., Springer, 2014]

PGD is a numerical method for solving **boundary value** problems.

Its **biggest advantage** is the ability to work in a **Parameterized** approach.

The vectors ***R(x)***, ***S(y)***, ***T(z)***,⋯ , ***Z(n)*** are robustly computed **offline.**

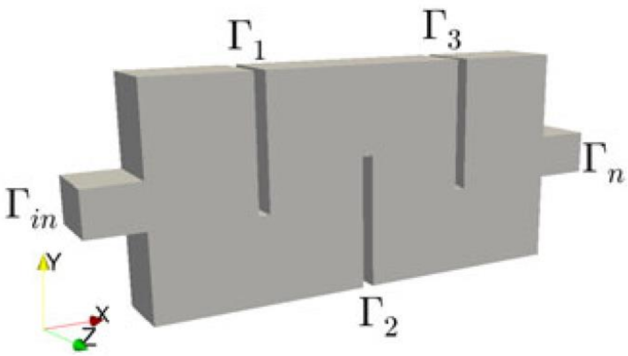
Fast ONLINE solution of a **recurring problem**.

PGD is not a ROM itself, it is more a strategy. However, it can use ROM to economize the vectors solutions ***R***, ***S***, ***T***,⋯, ***Z*** .

State of the Art: ROM

Analysis of Performance of ROM [Quarteroni et al., Springer, 2016]

- Galerkin Reduced Basis (G-RB) + POD or Greedy Algorithm.
- The parametric dependence of the PDE solution is exploited.
- **PROBLEM**: Steady heat conduction-convection problem.



Error bounds :

1) To compute the norm: $\|u_h(\mu) - u_N(\mu)\|_V$
2) To compute the stability factor (β_h) with *Interpolatory Radial Basis function*.
3) To compute the error estimator:

$$\Delta_N = \frac{\|u_h(\mu) - u_N(\mu)\|_V}{\beta_h}$$

Table 2. Computational details for HF and RB models. [Quarteroni et al., Chpt 3, Springer, 2016]

High-fidelity model		Reduced-Order model	
# FE dofs (N_h)	44171	# RB dofs	29
Affine operator components (Q_a)	2	Dofs reduction	1520:1
Affine RHs components (Q_f)	6	Offline CPU time	≈ 5 min
FE solution time	≈ 3,5 s	Online CPU time	1 ms

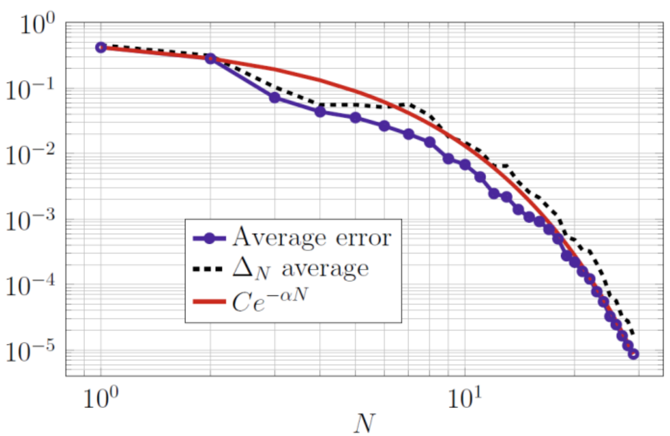


Figure 1. Comparison of the average error and bound error estimator computed on a set of 350 random values. [Quarteroni et al., Chpt 3, Springer, 2016]

State of the Art: ROM

POD approach using:
- ❑ Error estimator $\Delta_N(\mu)$
- ❑ Latin Hypercube Sampling (LHS)

Greedy algorithm critical aspects:
- ❑ Use of $\|u_h(\mu) - u_N(\mu)\|_V$
- ❑ It is not necessarily faster than POD

Greedy algorithm + POD:
- ❑ Error estimator $\Delta_N(\mu)$
- ❑ Latin Hypercube Sampling (LHS)
- ❑ Potentially faster

$n_s = 100$
$N = 26$
$\varepsilon_{POD} = 10^{-5}$

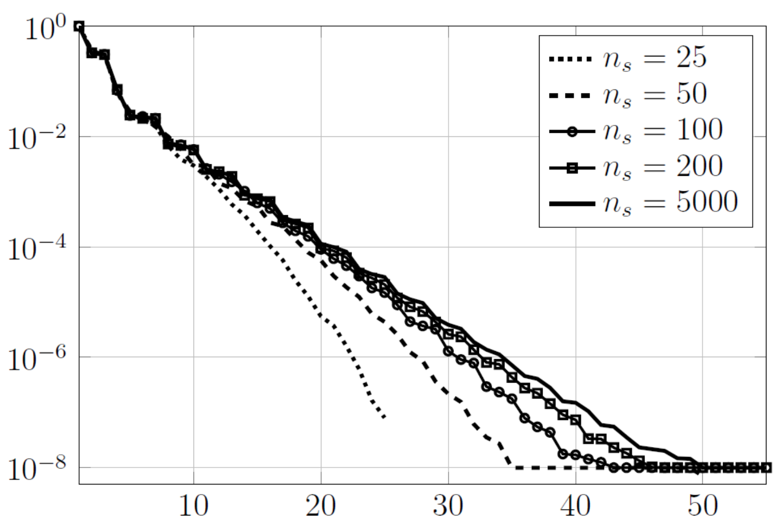


Figure 2. First 55 singular values of the correlation matrix obtained by LHS. [Quarteroni et al., Chpt 6, Springer, 2016]

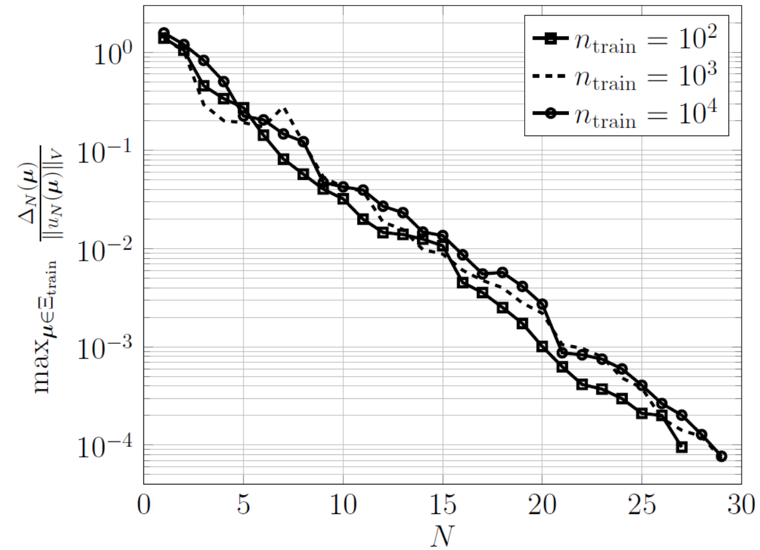


Figure 3. Convergence history of the greedy algorithm. [Quarteroni et al., Chpt 7, Springer, 2016]

Results

Advection-Diffusion problem definition

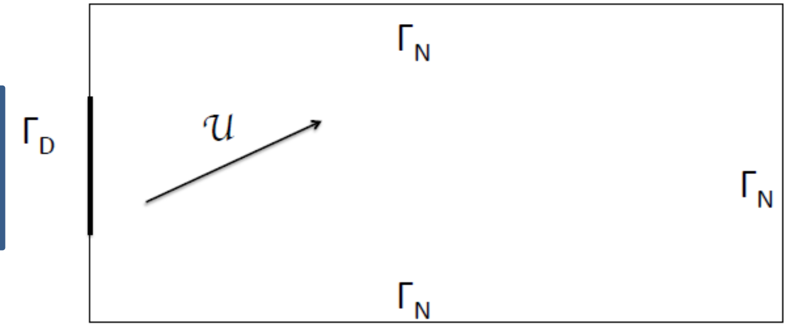
[Amsallem, International Journal for Numerical Methods in Engineering, 2014]

Steady Advection-Diffusion

$$\mathcal{U} \cdot \nabla\mathcal{T} - \kappa \cdot \Delta\mathcal{T} = 0$$

- $\boldsymbol{x} = (x, y) \in [0,1] \times [0,1]$
- $\mathcal{U} = \begin{bmatrix} \mathcal{U}_1 \\ \mathcal{U}_2 \end{bmatrix} = \begin{bmatrix} \mathcal{U}_1 \\ \cancel{0} \end{bmatrix}; \mathcal{U}_1 \epsilon\ [0; 0{,}5]$ → $u_2 \in [0; 0{,}5]$
- $\kappa\ \epsilon\ [0; 0{,}025]$

BOUNDARY CONDITIONS



$$\begin{cases} \mathcal{T}(\boldsymbol{x}; \boldsymbol{\mu}) = \mathcal{T}_D(y; \bar{y}) \\ \nabla\mathcal{T}(\boldsymbol{x}; \boldsymbol{\mu}) \cdot \boldsymbol{n}(\boldsymbol{x}) = 0 \end{cases}$$

$$\mathcal{T}_D(y, t; \bar{y}) = \begin{cases} 300 & \text{if } y \in [0; 1/3] \\ 300 + 325(\sin(3 \cdot \pi \cdot |y - \bar{y}|) + 1) & \text{if } y \in [1/3; 2/3] \\ 300 & \text{if } y \in [2/3; 1] \end{cases}$$

$\bar{y} \in [0\cancel{,1; 0,}6]$ → $\bar{y} = 0{,}4$

Results

Advection-Diffusion problem: POD

Algorithm *Greedy* POD

[Amsallem, International Journal for Numerical Methods in Engineering, 2014]

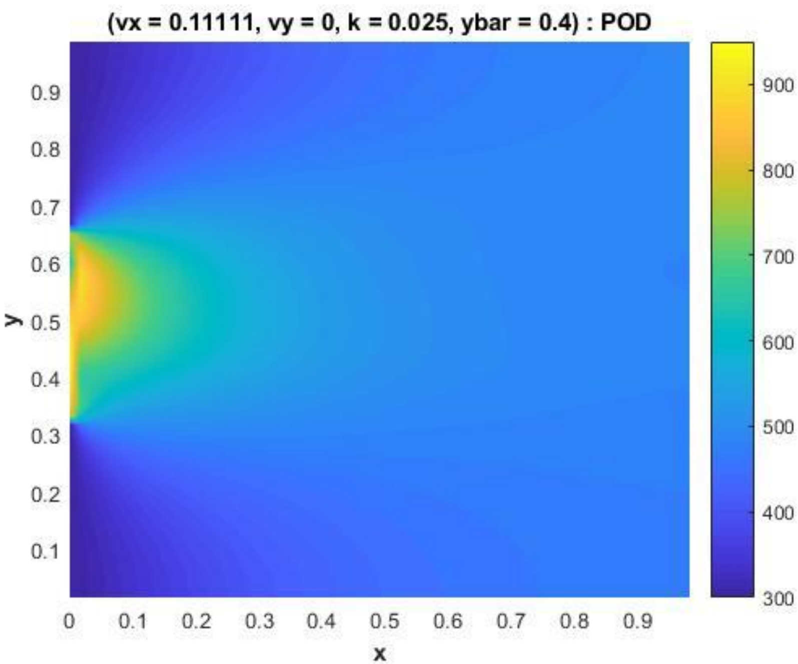
1. Select randomly a first sample: $\boldsymbol{\mu}^{(1)}$
2. Solve the HDM: $f\left(\boldsymbol{w}\left(\boldsymbol{\mu}^{(1)}; \boldsymbol{\mu}^{(1)}\right)\right) = 0$
3. Build a corresponding ROB: $\boldsymbol{V}$
4. For i = 2 ,⋯, s
   a) Solve:
   $\mu^{(1)} = argmax_{\mu \in \{\mu_1, \cdots, \mu_c\}} \|\boldsymbol{r}(\boldsymbol{\mu})\|$
   b) Solve the HDM: $f\left(\boldsymbol{w}\left(\boldsymbol{\mu}^{(i)}; \boldsymbol{\mu}^{(i)}\right)\right) = 0$
   c) Build a ROB $\boldsymbol{V}$ based on the samples $\{\boldsymbol{w}(\boldsymbol{\mu}^{(1)}), \cdots, \boldsymbol{w}(\boldsymbol{\mu}^{(i)})\}$

V size:
3487 x 10 =
34870 terms

$u_1 = 0{,}11 \ {}^{m}/_{s}$
$u_2 = 0 \ {}^{m}/_{s}$
$\kappa = 0{,}025 \ {}^{m^2}/_{s}$

Time computing **performance**: -85%

Solution **accuracy**: $E_{max} = 7\%$ $E_{avg} = 1\%$



Results

Advection-Diffusion problem: PGD

5 coordinates: x, y, u_1, u_2, κ

$$\int_{\Omega} \mathcal{T}^{*}(u\,\nabla\mathcal{T} - \kappa\,\Delta\mathcal{T})\,dx dy du_1 du_2 d\kappa = 0$$

$$\mathcal{T}^{n}(x, y, u_1, u_2, \kappa) = \sum_{i=1}^{n-1} X_i(x) Y_i(y) U_i(u_1) V_i(u_2) K_i(\kappa) + R(x) S(y) T(u_1) W(u_2) Z(\kappa)$$

1. The search of a less intrusive method
2. A non-symmetry induced by the convection terms.

Residual Minimization Technique for PGD

$$\min \; \|A \cdot X - F\|^2$$

[Chinesta et al., Springer, 2014]

AA = cell(5,4) → 2 diffusion eq. + 2 advection eq.

BB = cell(5,1) → RHS: Source terms

N_NT = cell(5,1) → Mass matrices

Dirichlet = cell(5,1) → Where to apply Dirichlet BC: $x = 0$

GG = cell(5,1) → A priori known terms, to enforce the Dirichlet BC

GG {2} = $\mathcal{T}_{\mathrm{D}}\ (y;\ \bar{y})$

Neumann BC

D1x(end,end) = 0;

D1y(1,1) = 0; D1y(end,end) = 0;

Results

Advection-Diffusion problem: PGD

OFFLINE

Loop	Stopping Criterion	Tolerance
Fixed point	$\frac{\lVert FF_n - FF_{n-1}\rVert}{\lVert FF_{n-1}\rVert}$	10^{-8}
PGD enrichment	$\frac{\lVert FF_n\rVert}{\lVert FF_1\rVert}$	10^{-6} **!!!**

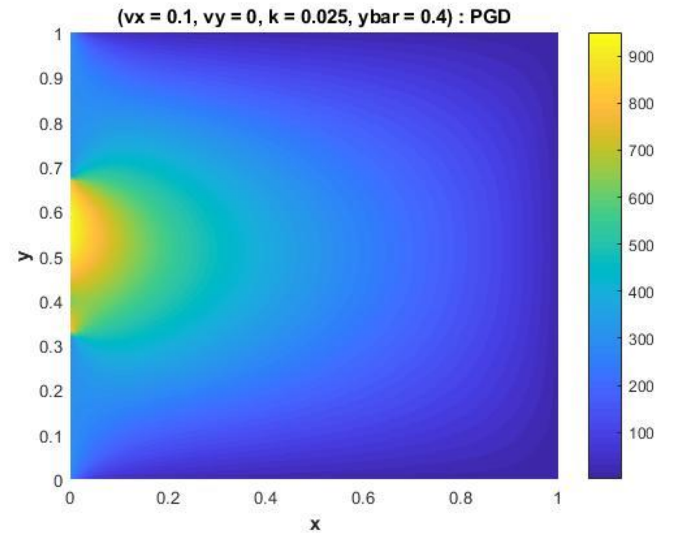
Computing time: 43min 25s

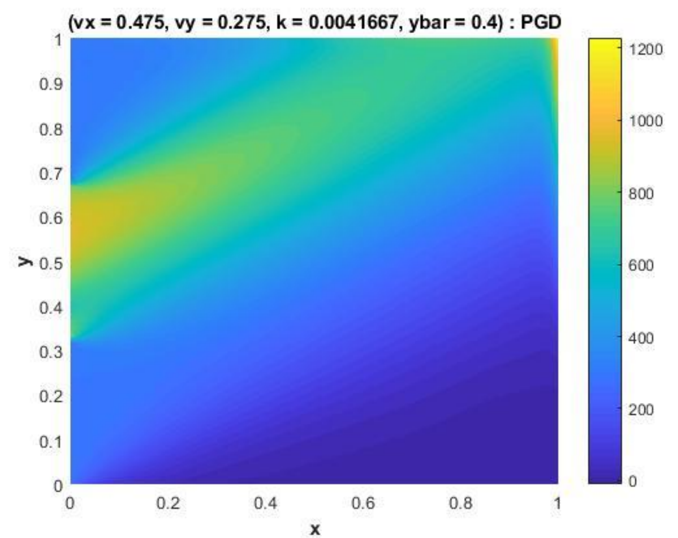
FF size: 292500 terms

ONLINE

```
alph = ones(size(FF{1},2),1);
alpha = FF{3}(5,:) .* FF{4}(1,:) .* FF{5}(15,:)
.* alph';
T = FF{2} * diag(alpha) * FF{1}';
```

Computing time: 0,1 – 0,5 s





Conclusions

- PGD → a **lot of cases** of the same (normally expensive) problem are expected.

⇩

POD

- DMD → simulations in **time domain**. We need a film of snapshots to create the reduced model.

PGD

- The most interesting PGD's approach is the **Parametric PGD**.
- PGD allows to introduce **Boundary Conditions** as extra-coordinates of the problem.
- The **Stopping Criterion** and **Tolerance** effectiveness should be evaluated for each problem.

Future work

Modification of the Stopping Criterion.

SUMMER INTERSHIP IN ONERA

Application of the strategy into *OpenAeroStruct.*

SUMMER INTERSHIP IN ONERA

Link to the GitHub Project

References

[1] Kutz, J.N., Brunton, S.L., Brunton, B.W. and Proctor, J.L., "Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems", 1st ed., SIAM, Philadelphia, 2016.

[2] Kutz, J. N., "Data-Driven Modeling & Scientific Computation: Methods for Complex Systems & Big Data", 1st ed., Oxford University Press, Oxford, 2013.

[3] Chinesta, F., Keunings, R. and Leygue, A., "The Proper Generalized Decomposition for Advanced Numerical Simulations: A primer", 1st ed., Springer, 2014.

[4] Quarteroni, A., Manzoni, A. and Negri, F., "Reduced Basis Methods for Partial Differential Equations. An Introduction", Unitext, vol. 92. Springer, 2016.

[5] Amsallem, D., “An Adaptive and Efficient Greedy Procedure for the Optimal Training of Parametric Reduced-Order Models”, International Journal for Numerical Methods in Engineering, 2014.

[6] http://www.dlr.de/ae/en/desktopdefault.aspx/tabid-1596/