Convex Displacement Interpolation for Nonlinear Approximation and Data Augmentation

Angelo Iollo
work with J.-C. Chapelier, J. Labatut, T. Taddei

MONHADE Team Inria – Onera
Institut de Mathématiques – Université de Bordeaux

Non-linear interpolation between solutions for two different geometries

$$\tilde{\rho}(.,\mu) = \left(\left(1 - s(\mu)\right)\rho_0 \circ \Psi_\mu^{-1} \circ X_\mu + s(\mu)\rho_1 \circ \Psi_\mu^{-1} \circ X_\mu^{-1}\right) \circ \Psi_\mu$$

Why we need non-linear interpolations: Parametric model problem in 1D

$$\begin{cases} -\dfrac{\partial^2 u_\mu}{\partial x^2} = f_\mu & \text{in } \Omega = (-1,1), \\ u_\mu(-1) = u_\mu(1) = 0; & \end{cases} \qquad f_\mu(x) = \frac{1}{\sigma}\exp\left(\frac{(x-\mu)^2}{\sigma^2}\right),$$

with $\mu \in \mathscr{P} = [-0.9, 0.9]$ and $\sigma > 0$.

Why we need non-linear interpolations: Parametric model problem in 1D

$$\begin{cases} -\dfrac{\partial^2 u_\mu}{\partial x^2} = f_\mu & \text{in } \Omega = (-1,1), \\ u_\mu(-1) = u_\mu(1) = 0; \end{cases} \qquad f_\mu(x) = \frac{1}{\sigma}\exp\left(\frac{(x-\mu)^2}{\sigma^2}\right),$$

with $\mu \in \mathscr{P} = [-0.9, 0.9]$ and $\sigma > 0$.

We introduce the space $\mathcal{U} = H_0^1(\Omega)$ endowed with the inner product $(w,v) = \int_{-1}^{1} \partial_x w \partial_x v + wv\, dx$

$$\text{find } u_\mu \in \mathcal{U} \ : \ a(u_\mu, v) = F_\mu(v) \quad \forall\, v \in \mathcal{U},$$

$$a(w,v) = \int_{-1}^{1} \partial_x w\, \partial_x v\, dx, \quad F_\mu(v) = \int_{-1}^{1} f_\mu v\, dx.$$

Why we need non-linear interpolations: Parametric model problem in 1D

We denote by $\mathscr{Z} \subset \mathcal{U}$ an n-dimensional linear subspace of $\mathcal{U}$; we further introduce the Galerkin reduced-order model:

$$\text{find } \widehat{u}_\mu \in \mathscr{Z} \ : \ a(\widehat{u}_\mu, v) = F_\mu(v) \ \ \forall\, v \in \mathscr{Z}.$$

Why we need non-linear interpolations: Parametric model problem in 1D

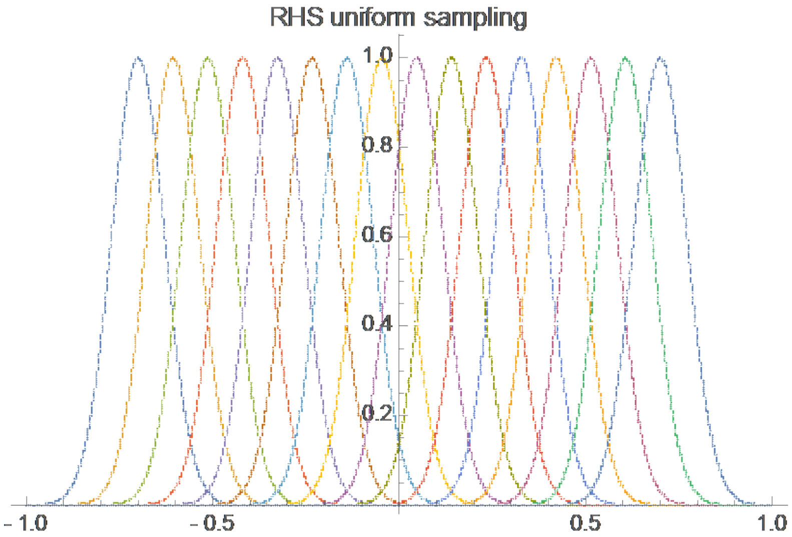
We denote by $\mathcal{Z} \subset \mathcal{U}$ an n-dimensional linear subspace of $\mathcal{U}$; we further introduce the Galerkin reduced-order model:

$$\text{find } \widehat{u}_\mu \in \mathcal{Z} \ : \ a(\widehat{u}_\mu, v) = F_\mu(v) \ \ \forall\, v \in \mathcal{Z}.$$

Céa lemma:

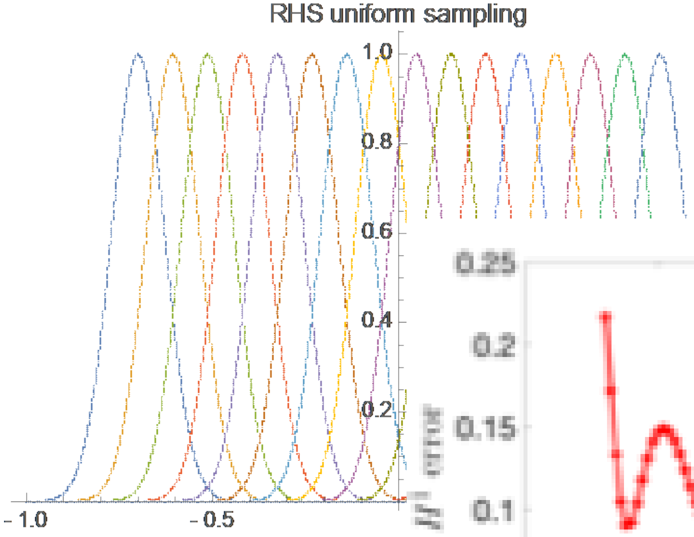
$$\|\widehat{u}_\mu - u_\mu\| \leq \sqrt{1 + \frac{2}{\pi}} \min_{v \in \mathcal{Z}} \|v - u_\mu\|, \quad \forall\, \mu \in \mathcal{P}.$$

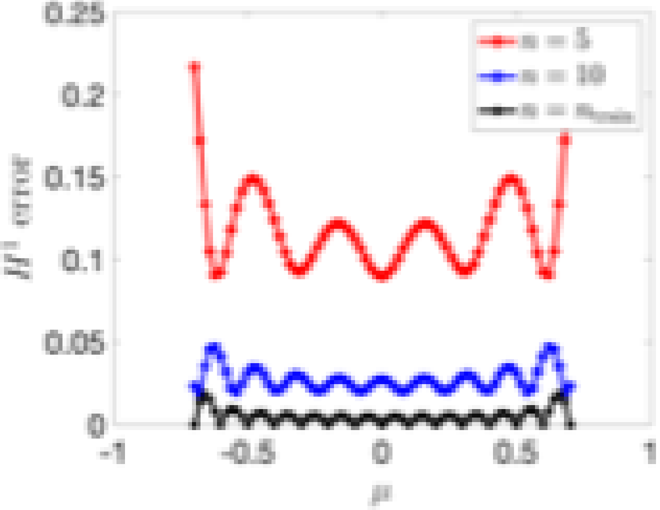
Why we need non-linear interpolations: Parametric model problem in 1D



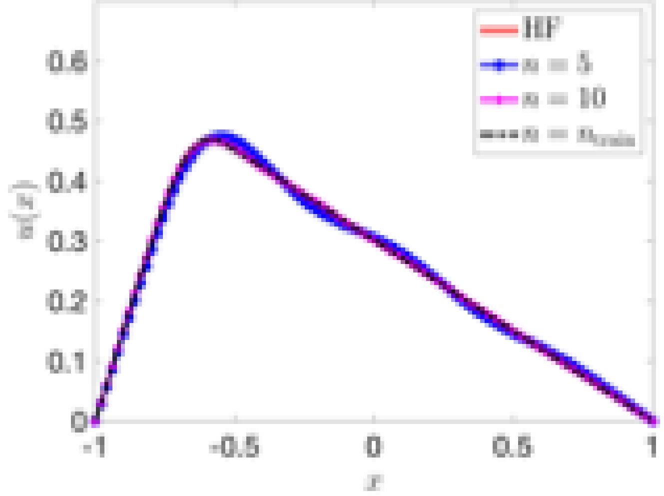
$\sigma = \mathbf{0.1}$

Why we need non-linear interpolations: Parametric model problem in 1D

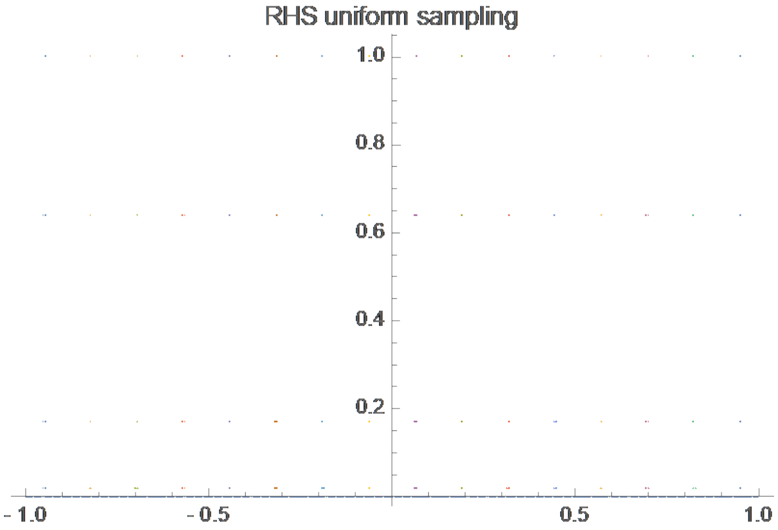




$\sigma = \mathbf{0.1}$

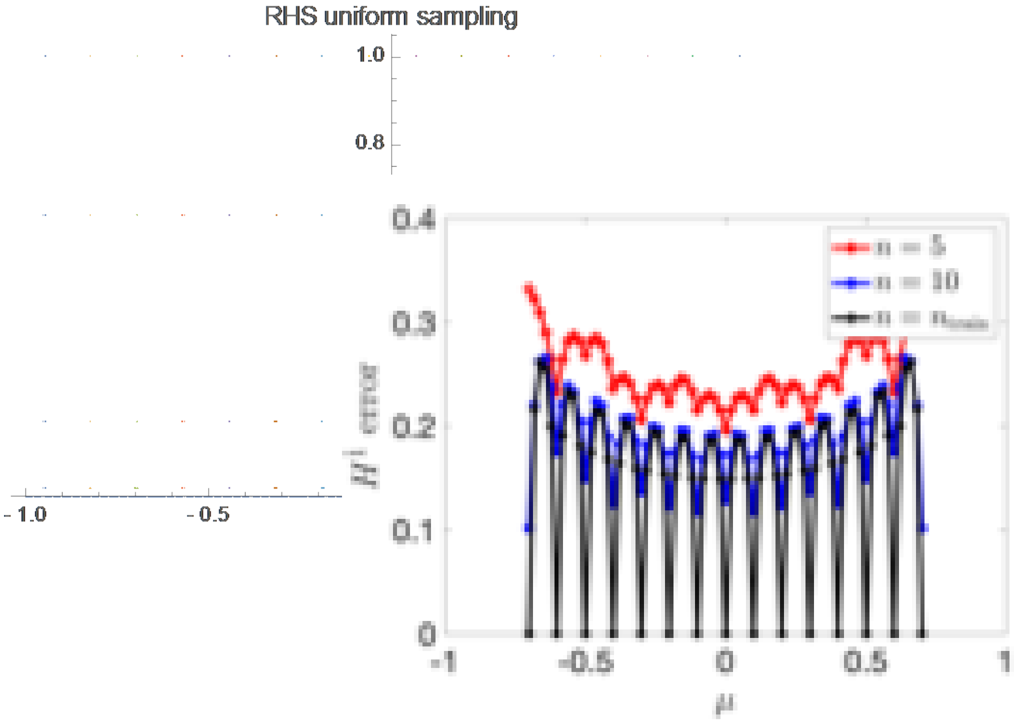


Why we need non-linear interpolations: Parametric model problem in 1D

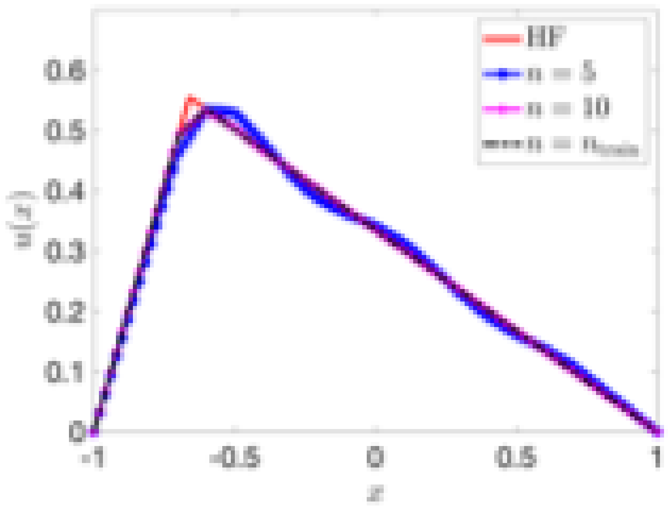


$\sigma = 0.001$

Why we need non-linear interpolations: Parametric model problem in 1D

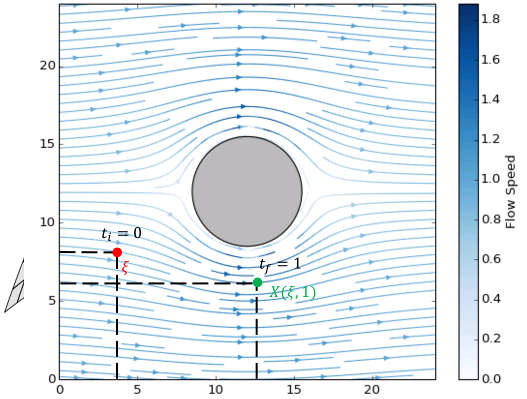


σ =0.001



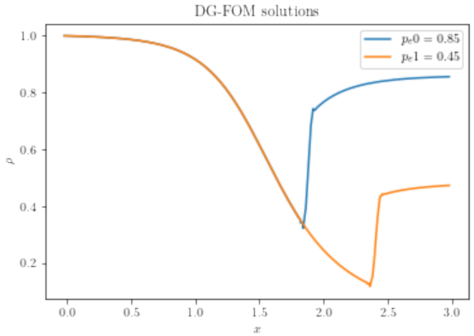
Convex Displacement Interpolation (CDI)

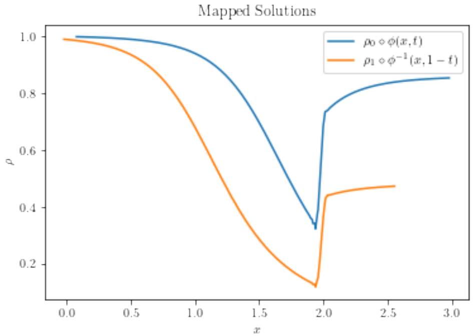
$$\begin{cases} \frac{\partial X}{\partial t}(\xi, t) = v \circ X(\xi, t) \\ X(\xi, 0) = \xi \end{cases} \qquad \begin{cases} \frac{\partial Z}{\partial t}(\xi, t) = -v \circ Z(\xi, t) \\ Z(\xi, 0) = \xi \end{cases}$$

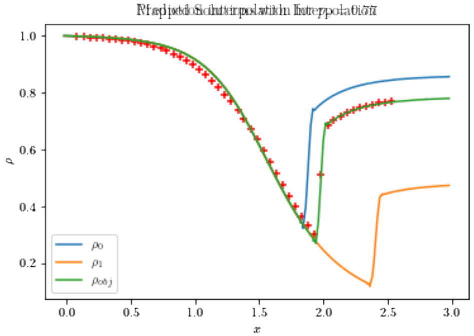


$$\tilde{U}_{CDI}(\xi, \mu) = (1 - s(\mu))U_0(Z(\xi, s(\mu))) + s(\mu)U_1(X(\xi, 1 - s(\mu)))$$

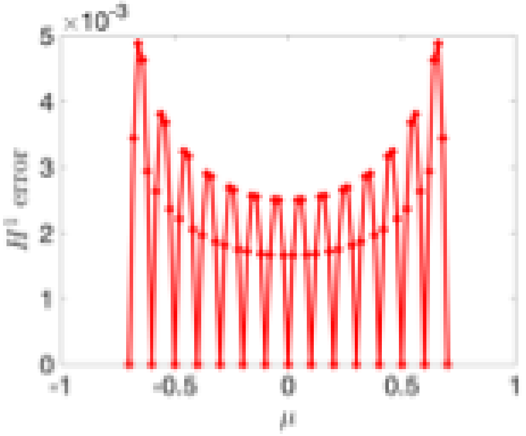
$$\tilde{U}_{CI}(\xi, \mu) = (1 - s(\mu))U_0(\xi) + s(\mu)U_1(\xi)$$

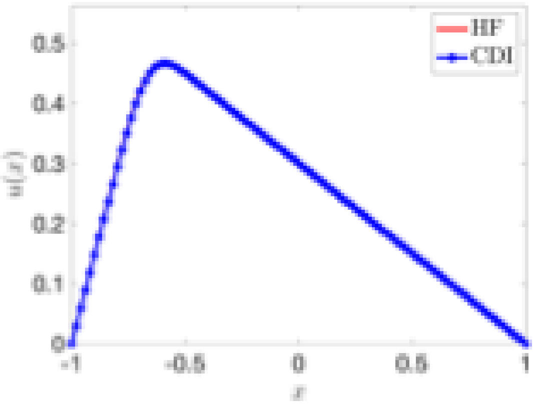






CDI interpolation error & worst error prediction field

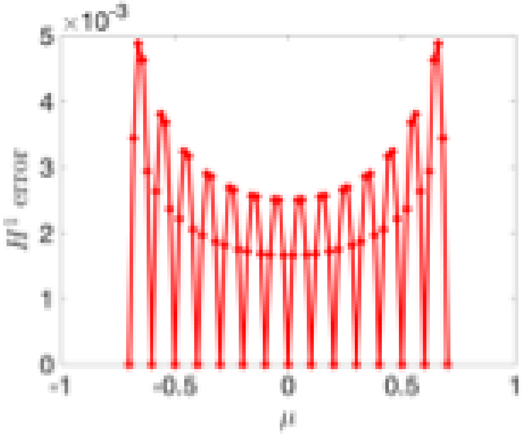


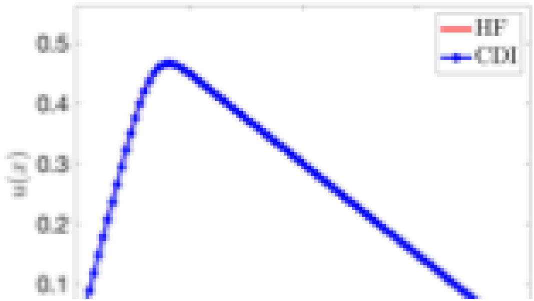


σ =0.1

we define the bijective maps Φ_ν for $\nu \in \mathscr{P}^{\mu}_{\mathrm{nn}}$ so that $\Phi_\nu(\Omega) = \Omega$, Φ_ν is piecewise linear in the intervals $(-1, \mu)$ and $(\mu, 1)$ and $\Phi_\nu(\mu) = \nu$. We notice that the error is below 0.5% for both values of σ.

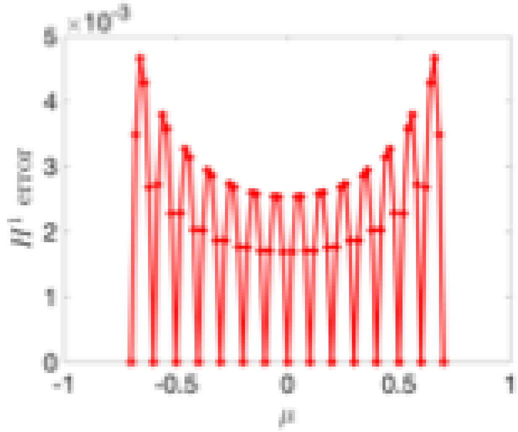
CDI interpolation error & worst error prediction field

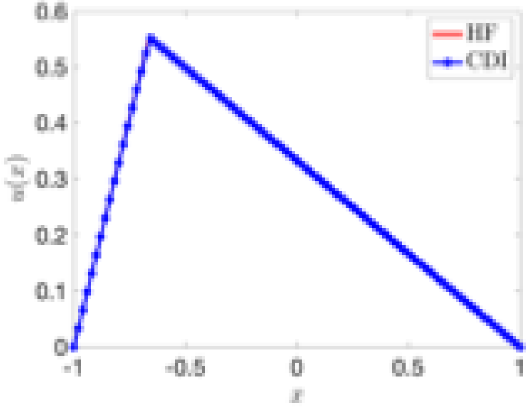




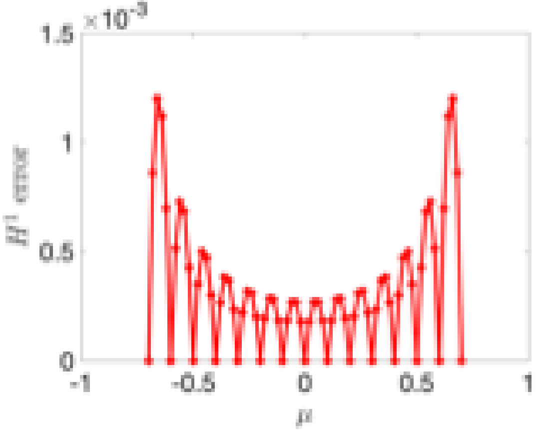
σ =0.1

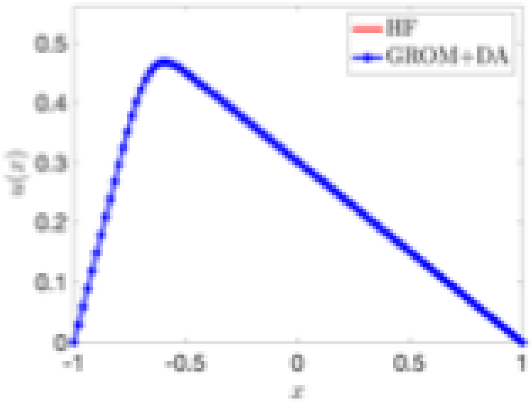
σ =0.001





ROM predictions based on CDI data augemntation

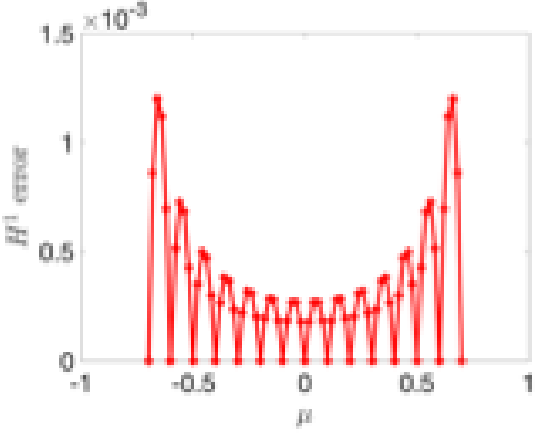


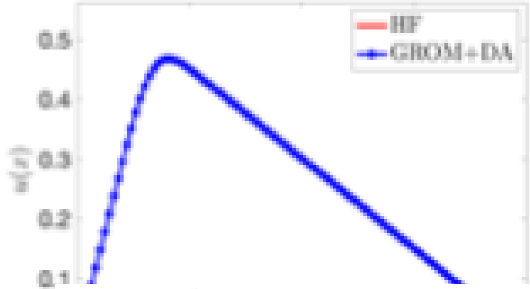


σ =**0.1**

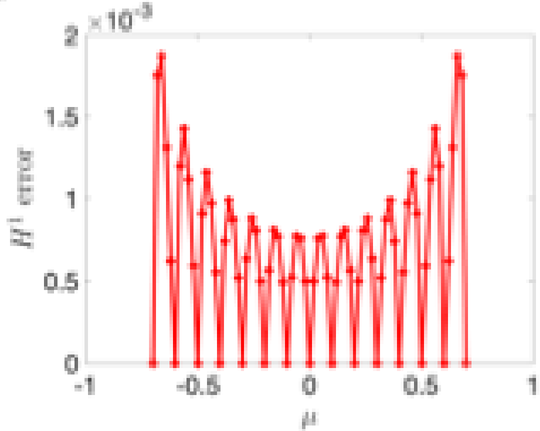
$$\mathcal{Z}_{\mu} = \text{span}\left\{\tilde{u}_{\mu}^{\text{cdi}}, u_{\nu^1}, u_{\nu^2}\right\}, \quad \text{with } \mathscr{P}_{\text{nn}}^{\mu} = \{\nu^1, \nu^2\}.$$

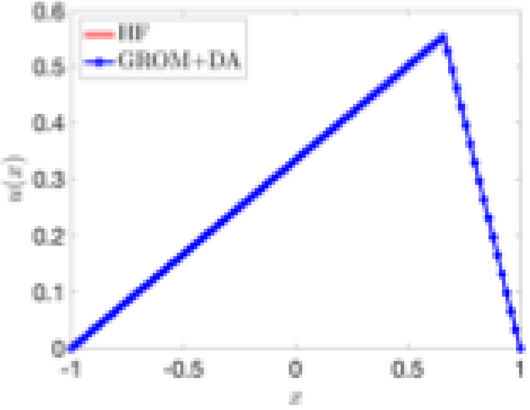
ROM predictions based on CDI data augemntation





σ =0.1

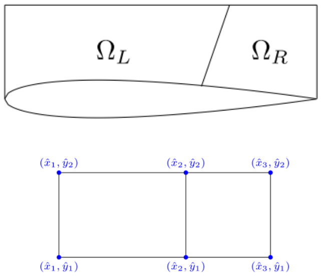




σ =0.001

Parametric Mappings in Bounded Domains

Parametric mappings



credit: Cagniart et al. 2017

Lagrangian-based ROM

$$\tilde{U}_\mu = \boldsymbol{\Psi}\boldsymbol{\alpha}_\mu \circ \Phi_\mu$$

AMD

[Iollo et al. 2014]

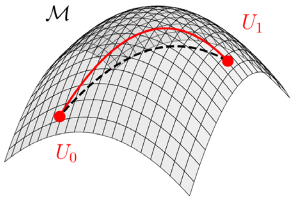
Non-linear interpolations in bounded domains

Multi-Frame ROM

$$\tilde{U}_\mu = \sum_{k=1}^{n} \boldsymbol{\alpha}_{\mu,k} \boldsymbol{\Psi}_k \circ \Phi_{\mu,k}$$

Data Augmentation

[Cucchiara et al. 2023]

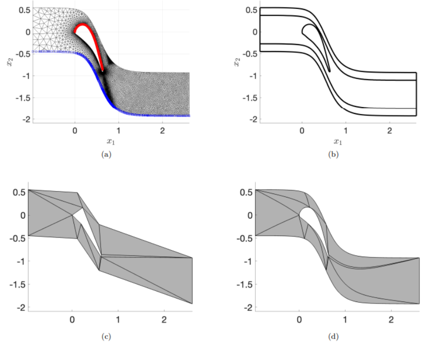


CDI

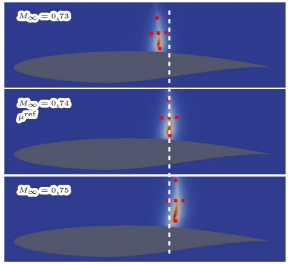
[Iollo et al. 2022]

Registration for ROM

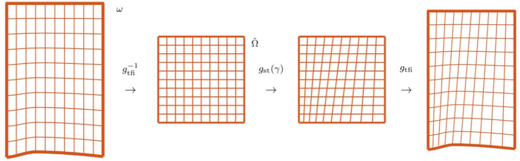
$$\Phi = \Psi \circ \mathrm{N} \circ \Psi^{-1}$$



credit: Taddei 2025

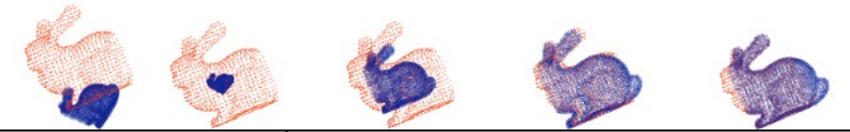


credit: Razavi et al. 2025



Registration

Objective: *Find a transformation* Φ *to align the two objects*



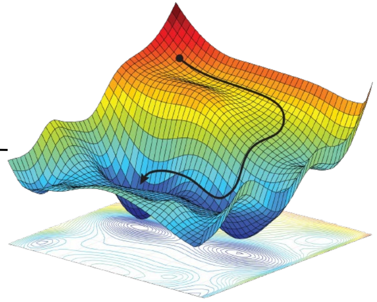
$$\Phi^* = \arg \min_{\Phi \in \mathrm{Diff}(\Omega)} \mathfrak{f}^{\mathrm{obj}}(\Phi) = \mathfrak{f}^{\mathrm{tg}}(\Phi) + \lambda \mathfrak{f}_{\mathrm{pen}}(\Phi)$$

Modeling

Energy Functional

Search Space

Minimization strategy

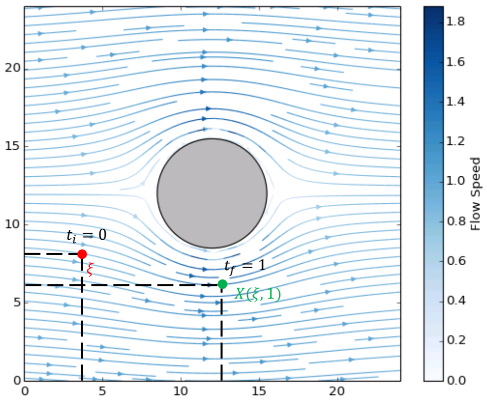


$$\Phi \in \mathcal{W}_M \subset \mathrm{Diff}(\Omega)$$

Registration in bounded domains: velocity based mapping

Mapping is defined as the solution of the following ODE:

$$\frac{\partial X}{\partial t}(\xi, t) = v \circ X(\xi, t)$$
$$X(\xi, 0) = \xi$$



Theorem :

if $v \in C^1(\Omega, \mathbb{R}^d)$ ***and*** $v.n = 0$ ***then*** $X(.,t)$ ***is a diffeomorphism and*** $X(\Omega, t) = \Omega, \quad \forall t$

We seek diffeomorphism of the form:

$$\Phi = X(., 1; v) \quad \text{with } v(x; \boldsymbol{a}) = \sum_{i=1}^{M} a_i \phi_i(x) \quad \forall x \in \Omega$$

The minimization aims at finding the coordinates :

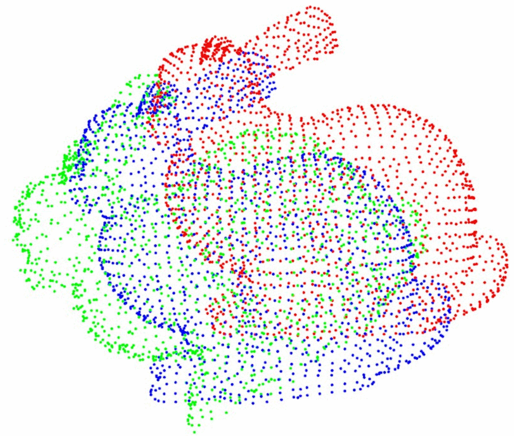
$$\boldsymbol{a} \in \mathbb{R}^M$$

Energy functional: point set registration

Source points: $\Xi = \{\xi_i\}_{i=1}^{N_0}$

Target points: $Y = \{y_j\}_{j=1}^{N_1}$

$$\mathfrak{f}^{\text{obj}}(\boldsymbol{a}) = \underbrace{\frac{1}{2}\sum_{i=1}^{N_0}\sum_{j=1}^{N_1} P_{ij}\|X(\xi_i, 1; \boldsymbol{a}) - y_j\|_2^2}_{\text{Data discrepancy}} + \underbrace{\frac{1}{2}\lambda \boldsymbol{a}^T \boldsymbol{K} \boldsymbol{a}}_{\text{Regularization}}$$



Data discrepancy

$P_{ij} \in [0, 1]$: weight informing whether ξ_i should be mapped to y_j

Assumed to be known

Regularization

$\boldsymbol{K}$: SPD Matrix built from a regularizing operator as:

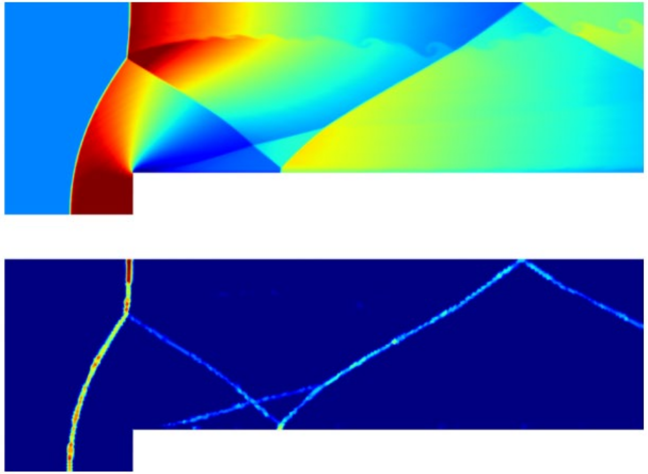
$$K_{ij} = \langle \mathcal{K}[\phi_i], \phi_j \rangle_{L^2}$$

Mapped points: $X(\Xi, 1; \boldsymbol{a}) = \{X(\xi_i, 1; \boldsymbol{a})\}_{i=1}^{N_0}$

Modeling: sensor based extraction

Point set registration for shocks:

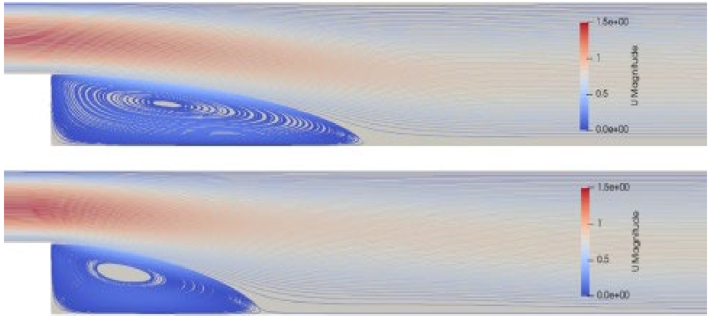
$$\varsigma_s(U) = \frac{1}{a\|\nabla p\|_2}\langle V, \nabla p\rangle_2$$



Persson. 2016

Iso contour extraction:

$$\varsigma_{iso}(U) = \begin{cases} \frac{\rho-\rho_\infty}{\max \rho-\rho_\infty} & \text{if } \rho > \rho_\infty \\ \frac{\rho-\rho_\infty}{\min \rho-\rho_\infty} & \text{if } \rho < \rho_\infty \end{cases}$$



Cucchiara et al. 2023

Minimization strategy: Gradient based

Functionnal to minimize:

$$\mathfrak{f}^{\text{obj}}(\boldsymbol{a}) = \frac{1}{2}\sum_{i=1}^{N_0}\sum_{j=1}^{N_1} P_{ij}\|X(\xi_i, 1; \boldsymbol{a}) - y_j\|_2^2 + \frac{1}{2}\lambda \boldsymbol{a}^T \boldsymbol{K}\boldsymbol{a}$$

$$v(x) = \sum_{i=1}^{M} a_i \phi_i(x)$$

Optimality condition:

$$\frac{\partial \mathfrak{f}^{\text{obj}}}{\partial \boldsymbol{a}}(\boldsymbol{a}^*) = 0$$

Equivalent to solve a system of non-linear equations :

$$\boldsymbol{f}(\boldsymbol{a}^*) + \boldsymbol{K}\boldsymbol{a}^* = \boldsymbol{0}$$

$$f_i(\boldsymbol{a}) = \int_0^1 \sum_{i=1}^{N_0}\sum_{j=1}^{N_1} P_{ij}\langle X(\xi_i, 1; \boldsymbol{a}) - y_j, \nabla X(\xi_i, 1; \boldsymbol{a})\nabla X(\xi_i, \tau; \boldsymbol{a})^{-1}\phi_i(X(\xi_i, \tau; \boldsymbol{a}))\rangle_2 d\tau$$

Gradient descent:

$$\boldsymbol{a}_{k+1} = \boldsymbol{a}_k - \eta_k(\boldsymbol{a}_k + \boldsymbol{K}^{-1}\boldsymbol{f}(\boldsymbol{a}_k))$$

η_k determined by line search (critical point)

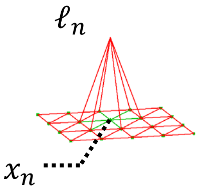
Discretization challenges

System of non-linear equations to solve: $\boldsymbol{f}(\boldsymbol{\alpha}^*) + \boldsymbol{K_h}\boldsymbol{\alpha}^* = \mathbf{0}$

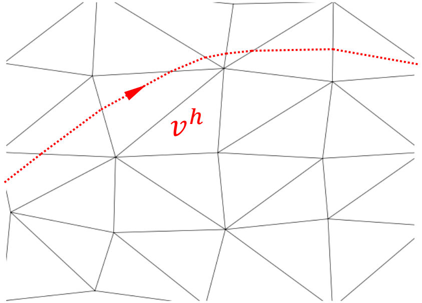
$$\hat{f}_n(\boldsymbol{\alpha}) = \int_0^1 \sum_{i=1}^{N_0} \sum_{j=1}^{N_1} P_{ij} \langle \widehat{X}(\xi_i, 1; \boldsymbol{\alpha}) - y_j, \widehat{\nabla X}(\xi_i, 1; \boldsymbol{\alpha}) \widehat{\nabla X}(\xi_i, \tau; \boldsymbol{\alpha})^{-1} \ell_n(\widehat{X}(\xi_i, \tau; \boldsymbol{\alpha})) \rangle_2 d\tau$$

Spatial discretization: FEM

$$v^h = \sum_{n=1}^{N_{\text{dof}}} \alpha_n \ell_n$$



Lagrange p_1



Time discretization: RK

Needed to solve ODE :

$$\frac{\partial X}{\partial t}(\xi, t) = v\left(X(\xi, t)\right)$$

$$\frac{\partial \nabla X}{\partial t}(\xi, t) = \nabla v\left(X(\xi, t)\right) \nabla X(\xi, t)$$

Ex: Euler integration

$$\hat{X}(\xi, t_{i+1}) = \hat{X}(\xi, t_i) + \Delta t v^h\left(\hat{X}(\xi, t_i)\right)$$

$$\widehat{\nabla X}(\xi, t_{i+1}) = \widehat{\nabla X}(\xi, t_i) + \Delta t \nabla v^h\left(\widehat{X}(\xi, t_i)\right) \widehat{\nabla X}(\xi, t_i)$$

Regularization matrix

Diffusion regularization

$$\mathcal{K}[v] = f$$

$$(Id - \beta\Delta)^s[v] = f$$

$$\langle v, \boldsymbol{n} \rangle = 0$$
$$\langle \nabla[v]\boldsymbol{n}, \boldsymbol{t} \rangle = 0$$

Following Sobolev injection $s > \frac{d}{2} + 1,\ H^s(\Omega) \hookrightarrow C^1(\Omega)$

Regularization matrix

Diffusion regularization

$$\mathcal{K}[v] = f$$

$$(Id - \beta\Delta)^s[v] = f$$

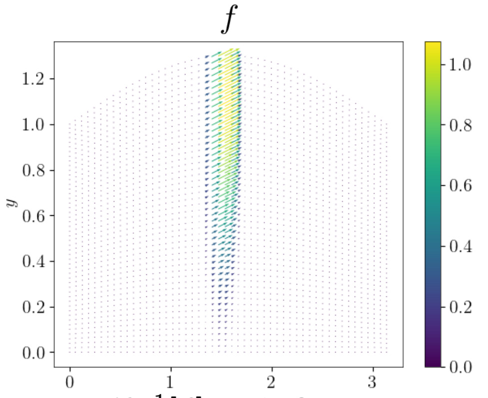
$$\langle v, \boldsymbol{n} \rangle = 0$$
$$\langle \nabla[v]\boldsymbol{n}, \boldsymbol{t} \rangle = 0$$

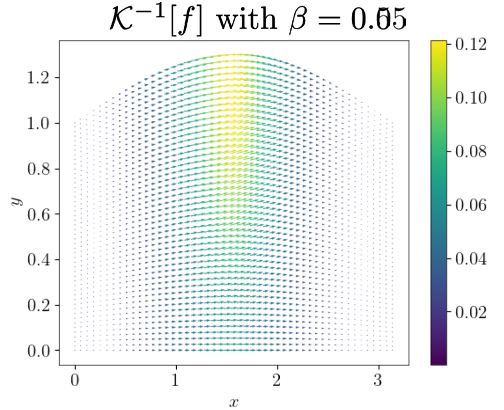
Following Sobolev injection $s > \frac{d}{2} + 1,\ H^s(\Omega) \hookrightarrow C^1(\Omega)$
Weak form built with Nitsche's method

$$b(u,v) = \int_\Omega v.u + \frac{1}{\kappa_0^2}\nabla v : \nabla u d\Omega + \frac{C_{pen}}{h}\int_{\partial\Omega}(v.\boldsymbol{n})(u.\boldsymbol{n})dS$$
$$- \int_{\partial\Omega}\frac{1}{\kappa_0^2}\nabla_{nn}[u]v.\boldsymbol{n}dS - \int_{\partial\Omega}\frac{1}{\kappa_0^2}\nabla_{nn}[v]u.\boldsymbol{n}dS$$

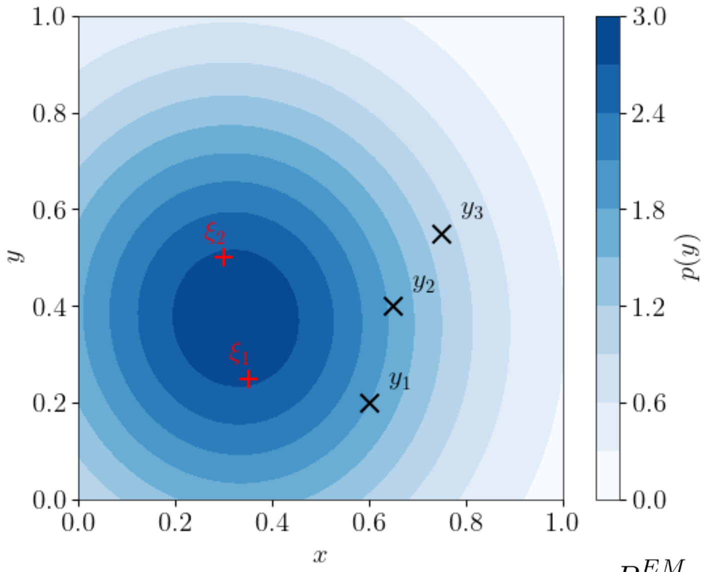
Regularization matrix $\boldsymbol{K}$ defined as:

$$\boldsymbol{K} = \left(\boldsymbol{L_h}\boldsymbol{M_h}^{-1}\right)^{s-1}\boldsymbol{L_h} \text{ with } \boldsymbol{L_h} = (b(\ell_i,\ell_j))_{i,j}^{N_{\text{dof}}\times N_{\text{dof}}} \text{ and } \boldsymbol{M_h} = (\langle \ell_i,\ell_j\rangle_{L^2})_{i,j}^{N_{\text{dof}}\times N_{\text{dof}}}$$





Matrix P: Expectation-Maximization



[Myronenko et al. 2010]

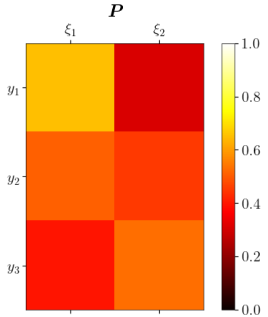
$$\hat{f}^{\text{obj}}(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{i=1}^{N_0}\sum_{j=1}^{N_1} \boxed{P_{ij}} \|\hat{X}(\xi_i, 1; \boldsymbol{\alpha}) - y_j\|_2^2 + \frac{1}{2}\lambda \boldsymbol{\alpha}^T \boldsymbol{K_h} \boldsymbol{\alpha}$$

$$p(y) = \frac{1}{N_1} w + \frac{1}{N_0}(1 - w)\sum_{j=1}^{N_0} p(y|j)$$

$$p(y) = \frac{1}{N_1} w + \frac{1}{N_0(2\pi\sigma^2)^{d/2}}(1 - w)\sum_{j=1}^{N_0} \exp\left(-\frac{1}{2}\frac{||\Phi(\xi_j; \boldsymbol{\alpha}) - y||_2^2}{\sigma^2}\right)$$

$$L(\boldsymbol{\alpha}, \sigma^2) = -\sum_{i=1}^{N_0} \log \sum_{j=1}^{N_1+1} P(j) p(y_i|j)$$

$$P_{ij}^{EM} = \frac{\exp\left(-\frac{1}{2}\left(\frac{||\Phi(\xi_j) - y_i||_2^2}{\sigma_{k-1}^2}\right)\right)}{\sum_{n=1}^{N_0} \exp\left(-\frac{1}{2}\left(\frac{||\Phi(\xi_n) - y_i||_2^2}{\sigma_{k-1}^2}\right)\right) + (2\pi\sigma_{k-1}^2)^{d/2}\frac{w}{1-w}\frac{N_0}{N_1}}$$



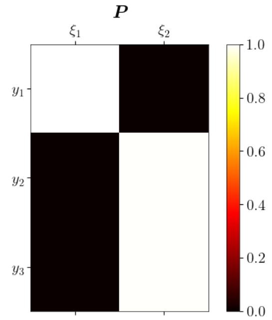
$$\boldsymbol{P} = \boldsymbol{P}^{\varsigma} \odot \boldsymbol{P}^{EM}$$

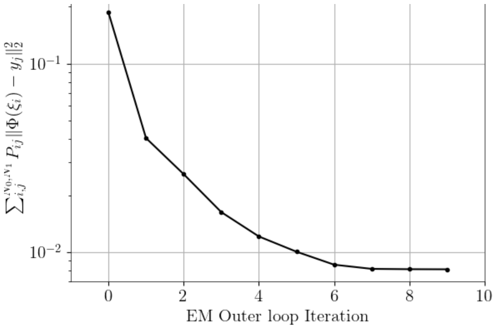
Minimization algorithm

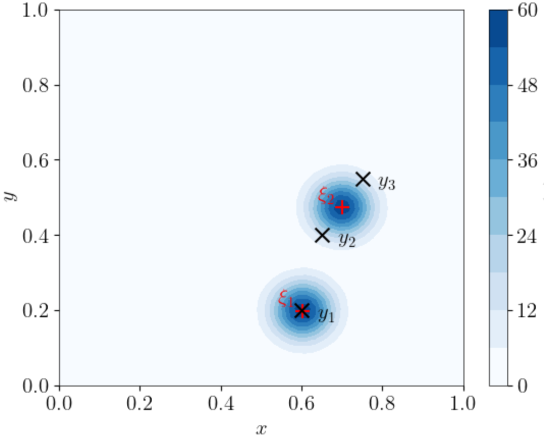
Data: $\Xi, Y, \varepsilon, \sigma_0, \boldsymbol{P}^{\varsigma}$
Result: velocity FE dof $\boldsymbol{\alpha}$
Initialization;
$\boldsymbol{\alpha} \leftarrow 0$;
$\sigma \leftarrow \sigma_0$;
$k \leftarrow 0$;
Optimization;
while $|\Delta\sigma_k^2| > \varepsilon$ **do**
  E-step;
    $P_{ij} \leftarrow (2.24)$;
  M-step ;
    $v^h = \sum_{p=1}^{N_{\text{dof}}} \alpha_p \ell_p$;
    $\hat{X}(\xi_i, 1) \leftarrow \xi_i + \sum_{m=1}^{N_{RK}} \omega_m v^h(\hat{X}(y_m, t_m), t_m)$;
    $\widehat{\nabla X}(\xi_i, 1) \leftarrow \text{Id} + \sum_{m=1}^{N_{RK}} \omega_m \nabla v^h(\hat{X}(y_m, t_m), t_m)\widehat{\nabla X}(\hat{X}(y_m, t_m), t_m)$;
    $\hat{f}_n \leftarrow \int_0^1 \sum_{i,j=1}^{N_0,N_1} P_{ij} \left\langle X(\xi_i, 1) - y_j, \nabla X(\xi_i, 1)(\nabla X(\xi_i, \tau))^{-1} \ell_n(X(\xi_i, \tau)) \right\rangle_2 d\tau$;
    $\boldsymbol{\alpha}_{k+1} \leftarrow \boldsymbol{\alpha}_k - \eta_k \left(\boldsymbol{K}_h^{-1} \boldsymbol{f}(\boldsymbol{\alpha}_k) + \lambda \boldsymbol{\alpha}_k \right)$; /* η from line search */
    $\sigma_{k+1}^2 \leftarrow (2.28)$;
    $\Delta\sigma_{k+1}^2 \leftarrow \sigma_{k+1}^2 - \sigma_k^2$;
    $k \leftarrow k + 1$;
end

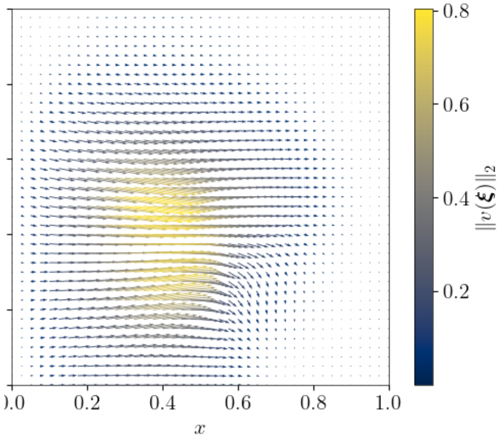
Example

Data: $\Xi, Y, \varepsilon, \sigma_0, \boldsymbol{P}^\varsigma$
Result: velocity FE dof $\boldsymbol{\alpha}$
Initialization;
$\boldsymbol{\alpha} \leftarrow 0$;
$\sigma \leftarrow \sigma_0$;
$k \leftarrow 0$;
Optimization;
while $|\Delta\sigma_k^2| > \varepsilon$ **do**
  E-step ;
    $P_{ij} \leftarrow (2.24)$;
  M-step ;
    $v^h = \sum_{p=1}^{N_{\text{dof}}} \alpha_p \ell_p$;
    $\hat{X}(\xi_i, 1) \leftarrow \xi_i + \sum_{m=1}^{N_{RK}} \omega_m v^h(\hat{X}(y_m, t_m), t_m)$;
    $\widehat{\nabla X}(\xi_i, 1) \leftarrow \mathrm{Id} + \sum_{m=1}^{N_{RK}} \omega_m \nabla v^h(\hat{X}(y_m, t_m), t_m)\widehat{\nabla X}(\hat{X}(y_m, t_m), t_m)$;
    $\hat{f}_n \leftarrow \int_0^1 \sum_{i,j=1}^{N_0,N_1} P_{ij} \left\langle X(\xi_i, 1) - y_j, \nabla X(\xi_i, 1)(\nabla X(\xi_i, \tau))^{-1} \ell_n(X(\xi_i, \tau)) \right\rangle_2 d\tau$;
    $\boldsymbol{\alpha}_{k+1} \leftarrow \boldsymbol{\alpha}_k - \eta_k \left(\boldsymbol{K}_h^{-1} \boldsymbol{f}(\boldsymbol{\alpha}_k) + \lambda \boldsymbol{\alpha}_k \right)$; /* η from line search */
    $\sigma_{k+1}^2 \leftarrow (2.28)$;
    $\Delta\sigma_{k+1}^2 \leftarrow \sigma_{k+1}^2 - \sigma_k^2$;
    $k \leftarrow k + 1$;
end



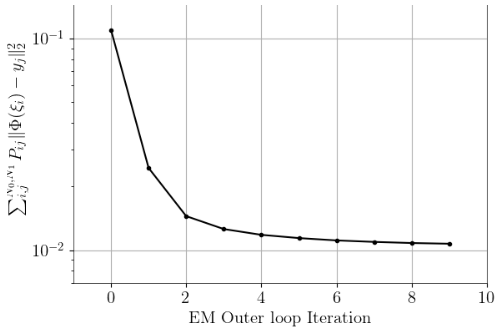




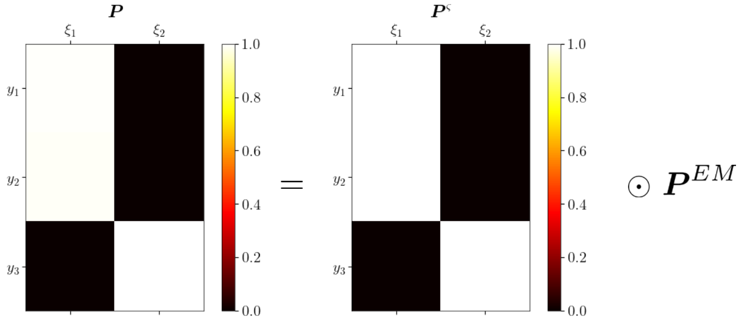


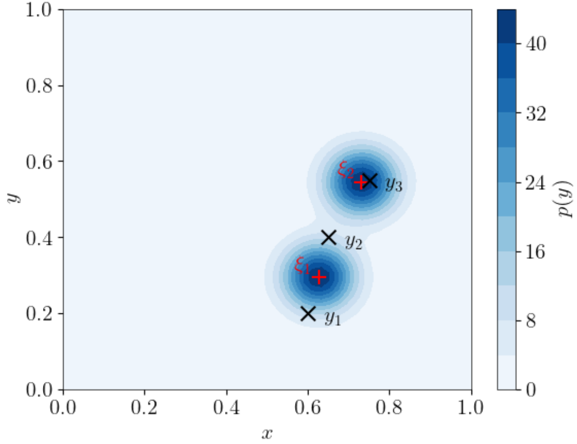
Example II

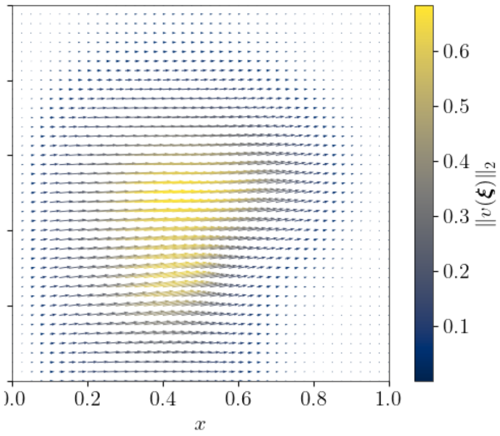
Data: $\Xi, Y, \varepsilon, \sigma_0, \boldsymbol{P}^\varsigma$
Result: velocity FE dof $\boldsymbol{\alpha}$
Initialization;
$\boldsymbol{\alpha} \leftarrow 0$;
$\sigma \leftarrow \sigma_0$;
$k \leftarrow 0$;
Optimization;
while $|\Delta\sigma_k^2| > \varepsilon$ **do**
  E-step ;
    $P_{ij} \leftarrow (2.24)$;
  M-step ;
    $v^h = \sum_{p=1}^{N_{\mathrm{dof}}} \alpha_p \ell_p$;
    $\hat{X}(\xi_i, 1) \leftarrow \xi_i + \sum_{m=1}^{N_{RK}} \omega_m v^h(\hat{X}(y_m, t_m), t_m)$;
    $\widehat{\nabla X}(\xi_i, 1) \leftarrow \mathrm{Id} + \sum_{m=1}^{N_{RK}} \omega_m \nabla v^h(\hat{X}(y_m, t_m), t_m) \widehat{\nabla X}(\hat{X}(y_m, t_m), t_m)$;
    $\hat{f}_n \leftarrow \int_0^1 \sum_{i,j=1}^{N_0,N_1} P_{ij} \left\langle X(\xi_i, 1) - y_j, \nabla X(\xi_i, 1)(\nabla X(\xi_i, \tau))^{-1} \ell_n(X(\xi_i, \tau)) \right\rangle_2 d\tau$;
    $\boldsymbol{\alpha}_{k+1} \leftarrow \boldsymbol{\alpha}_k - \eta_k \left(\boldsymbol{K}_h^{-1} \boldsymbol{f}(\boldsymbol{\alpha}_k) + \lambda \boldsymbol{\alpha}_k\right)$; /* η from line se
    $\sigma_{k+1}^2 \leftarrow (2.28)$;
    $\Delta\sigma_{k+1}^2 \leftarrow \sigma_{k+1}^2 - \sigma_k^2$;
    $k \leftarrow k + 1$;
end



$$\boldsymbol{P} = \boldsymbol{P}^\varsigma \odot \boldsymbol{P}^{EM}$$







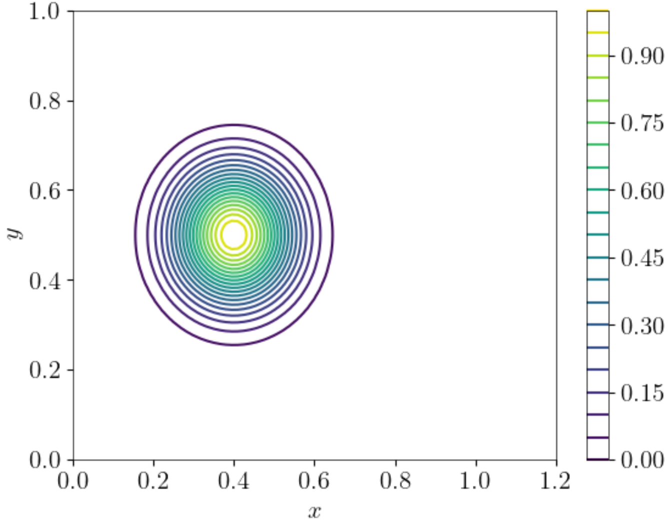
Test Case 1: Coalescing Gaussian

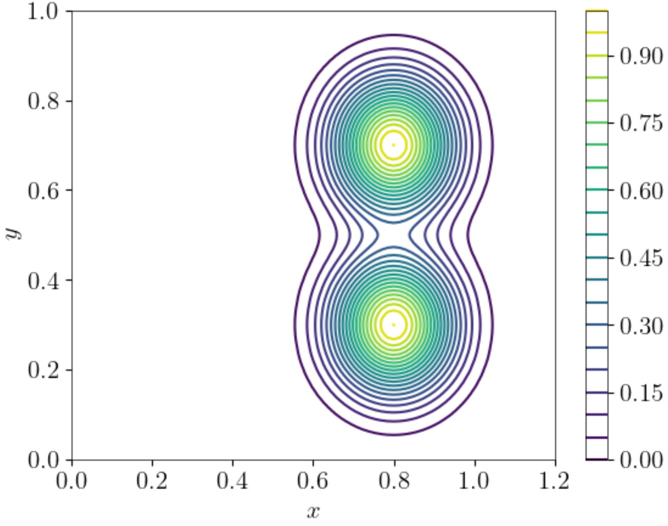
$$I_0(\boldsymbol{x}) = \exp\left(-\frac{1}{2\sigma_I^2}(\|\boldsymbol{x} - \boldsymbol{c}_0\|_2^2)\right)$$

$$\boldsymbol{c}_0 = [0.8, 0.7]^T$$

$$I_1(\boldsymbol{x}) = \exp\left(-\frac{1}{2\sigma_I^2}(\|\boldsymbol{x} - \boldsymbol{c}_1^a\|_2^2)\right) + \exp\left(-\frac{1}{2\sigma_I^2}(\|\boldsymbol{x} - \boldsymbol{c}_1^b\|_2^2)\right)$$

$$\boldsymbol{c}_1^a = [0.8, 0.7]^T, \quad \boldsymbol{c}_1^b = [0.8, 0.3]^T$$

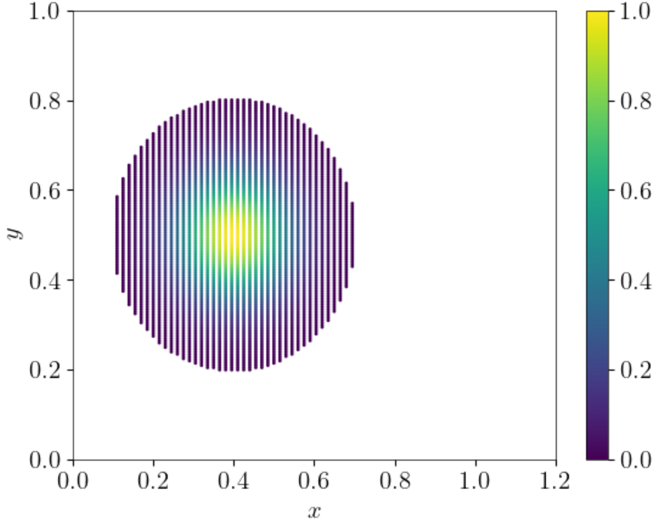


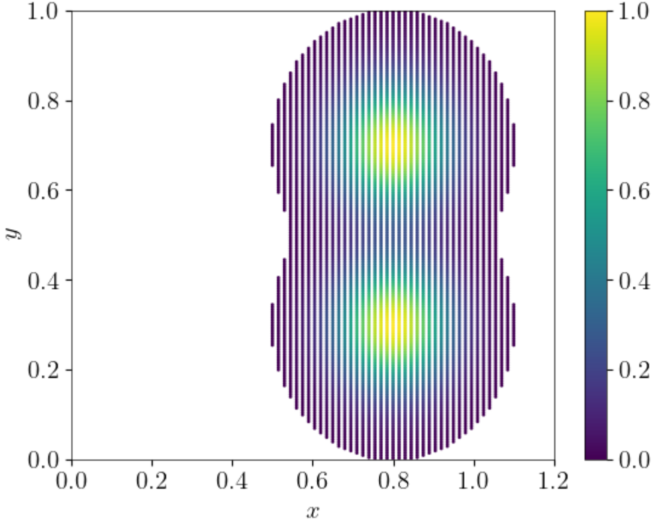


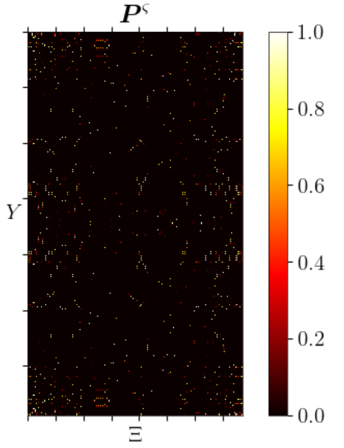
Point set construction

$$\Xi = \{(\xi_j)_{j=1}^{N_0} \in \mathcal{T}_\varsigma \;:\; I_0^h(\xi_j) > 10^{-2}\}$$

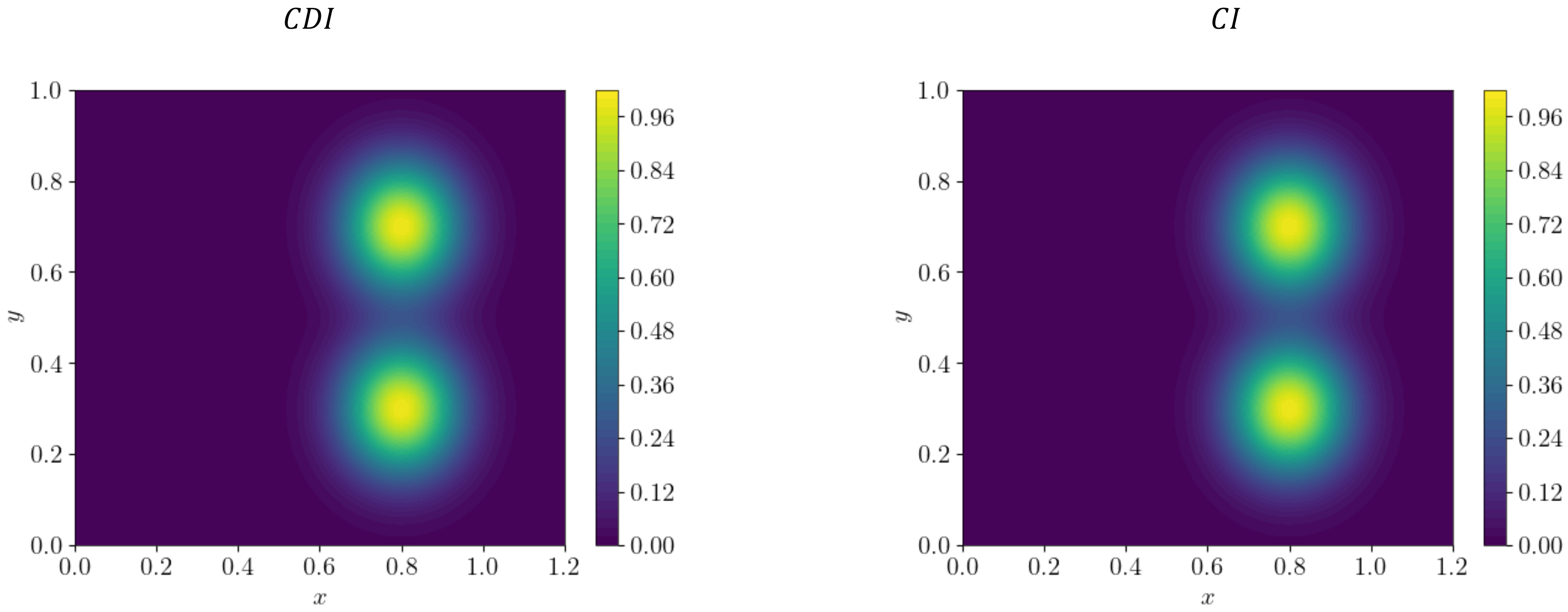
$$Y = \{(y_i)_{i=1}^{N_1} \in \mathcal{T}_\varsigma \;:\; I_1^h(y_i) > 10^{-2}\}$$







Coalescing structures : interpolation



RESULTS

Application of mapping computation to industrial-like ROM problems

Test Case presentation: NACA0012

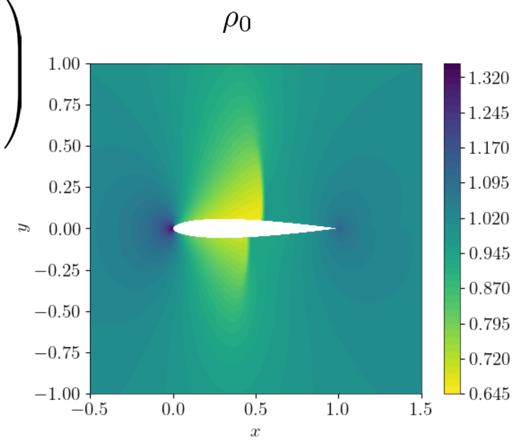
Model : 2D Euler

$$\frac{\partial U}{\partial t} + \frac{\partial f_x(U)}{\partial x} + \frac{\partial f_y(U)}{\partial y} = 0$$

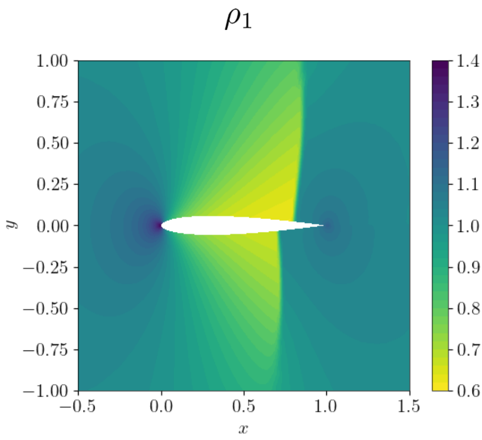
$$U = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho E \end{pmatrix} \quad f_x(U) = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p \\ \rho v_y v_x \\ v_x(\rho E + p) \end{pmatrix} \quad f_y(U) = \begin{pmatrix} \rho v_y \\ \rho v_y v_x \\ \rho v_y^2 + p \\ v_y(\rho E + p) \end{pmatrix}$$

Numerical solver : Aghora

DG p2 : DoFs/eq ~ 150,000
Convective flux: ROE
Linear solver: Restarted GMRES with left preconditioning
Time stepping : Implicit back Euler Jacobian free



$M_\infty = 0.8$



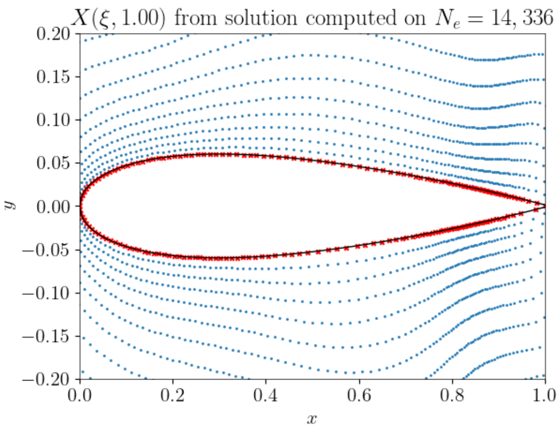
$M_\infty = 0.85$

Accuracy of domain boundaries $\Phi(\Omega) = \Omega$?

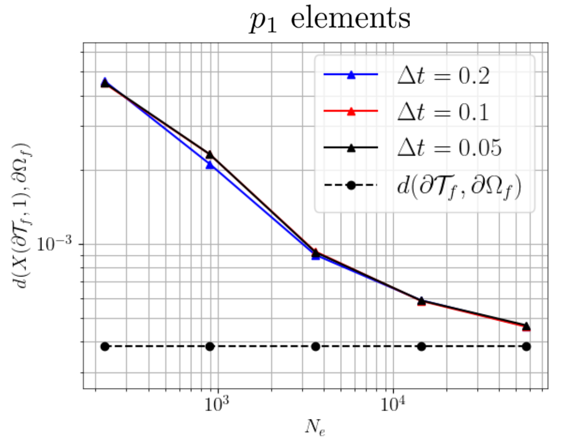
$\partial\Omega_f = \{\boldsymbol{S}_{CR}(t)\ \forall t \in [0,1]\}$ with:

$$\boldsymbol{S}_{CR}(t_s) = \frac{1}{2}\begin{bmatrix} t_s^3 & t_s^2 & t_s & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p}_{i-1} \\ \mathbf{p}_i \\ \mathbf{p}_{i+1} \\ \mathbf{p}_{i+2} \end{bmatrix}$$

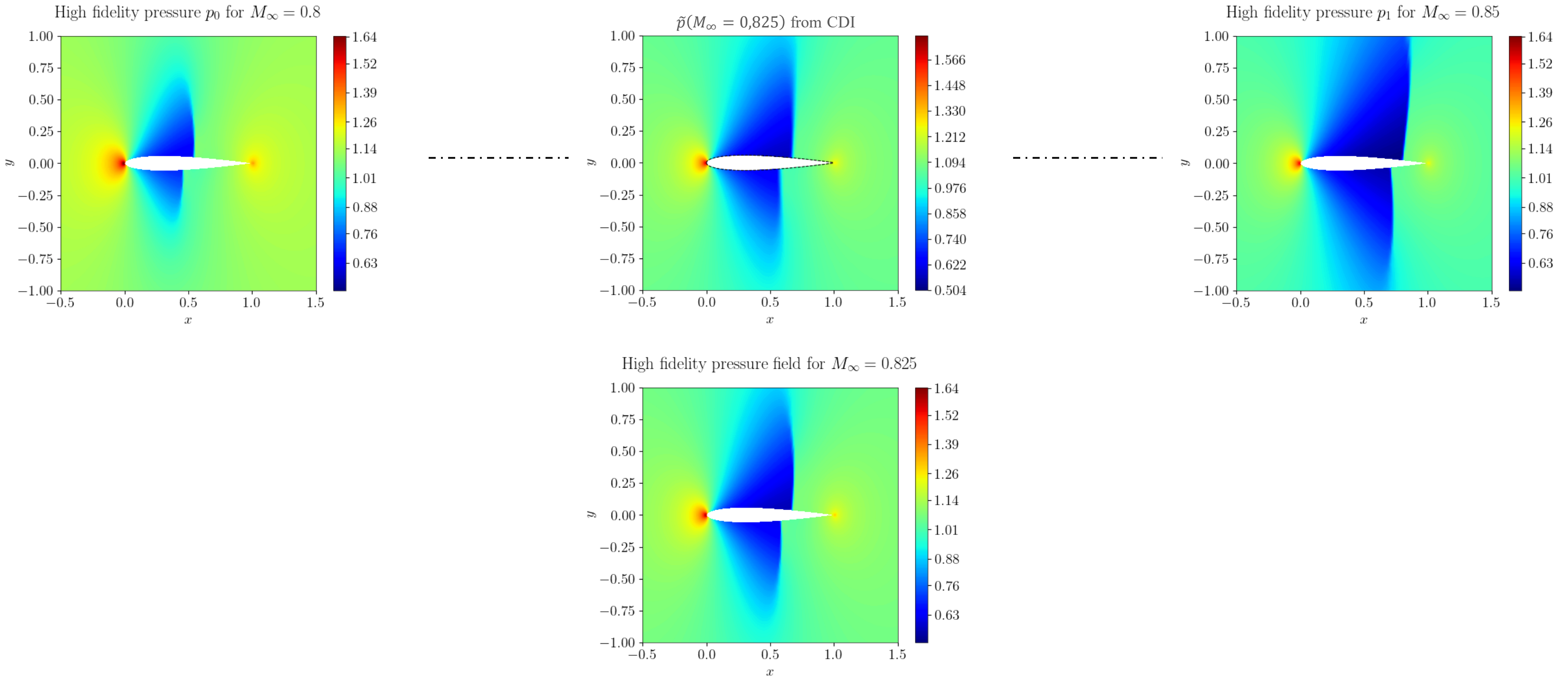
$$d(X(\partial\mathcal{T}_f, 1), \partial\Omega_f) = \max_{i\in\{1,\ldots,N_h\}} \min_{p\in\partial\Omega_f} \|X(\xi_i, 1) - p\|_2$$



$\{\boldsymbol{\xi}_i\} \in \partial\mathcal{T}_f \subset \partial\Omega_f$

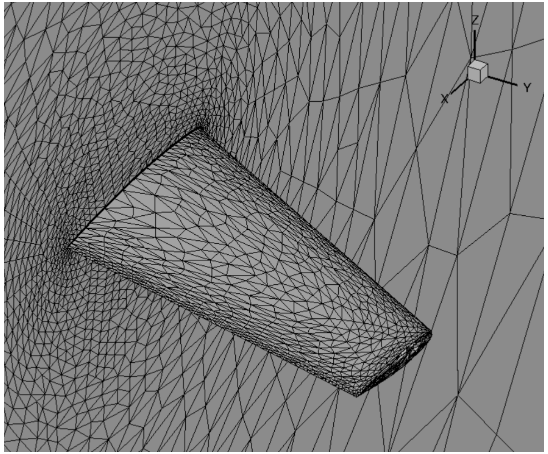


Interpolation of pressure field

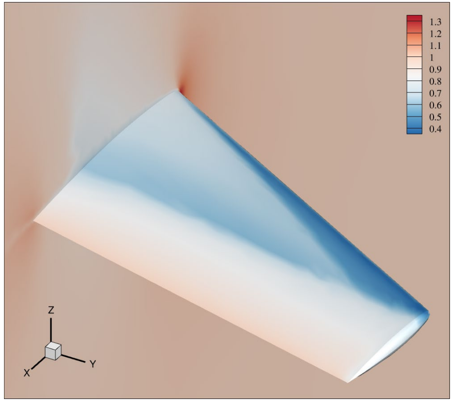


Test Case presentation: ONERA M6

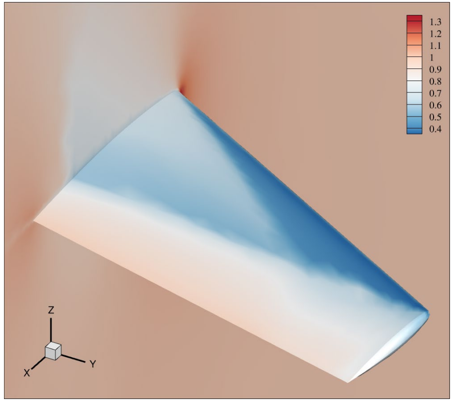
Numerical solver : CODA
DG p1: Mesh of $\approx 3 \times 10^5$ nodes
Fluid model RANS $Re = 14.6 \times 10^6$, $M_\infty = 0{,}84$
Convective flux: ROE
Linear solver: Restarted GMRES with left preconditioning
Time stepping : Netwon methods with finite differencing approximation of the jacobian
Turbulence model: SA with artifical viscosity



$AoA_0 = 3{,}06°$

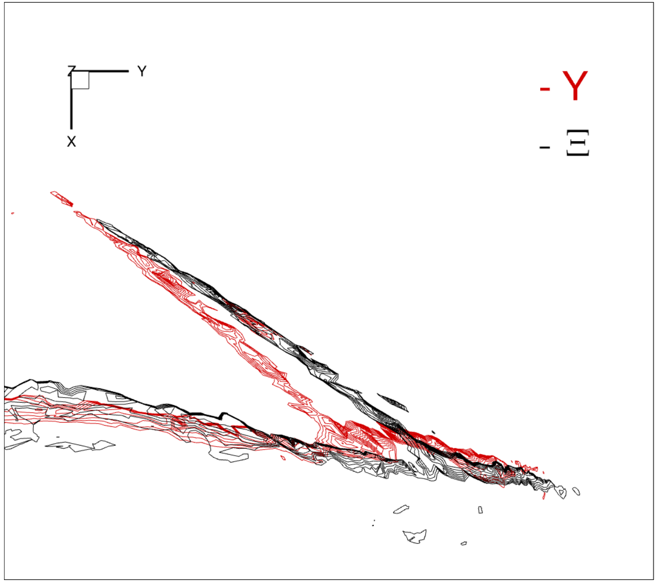


$AoA_1 = 5{,}1°$



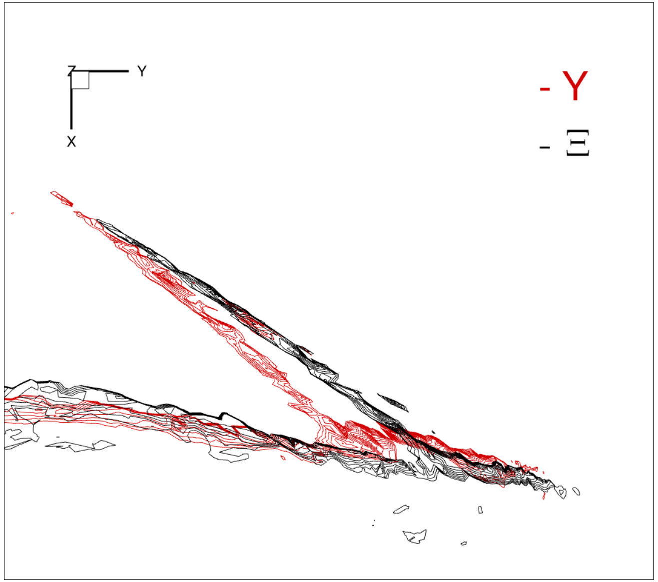
Point set construction

$$\Xi = \{(\xi_j)_{j=1}^{N_0} \in \mathcal{T}_{DG} \; : \; \varsigma_S(U_0^h(\xi_j)) > 0.6\}$$
$$Y = \{(y_i)_{i=1}^{N_1} \in \mathcal{T}_{DG} \; : \; \varsigma_S(U_1^h(y_i)) > 0.6\}$$

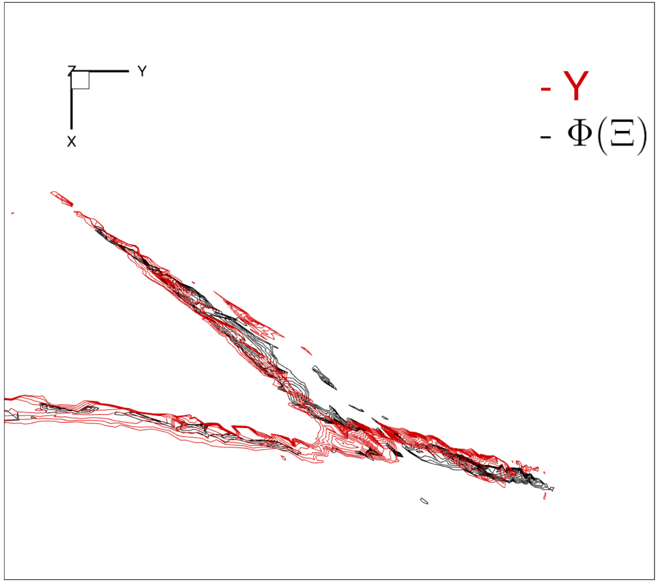


Mapping results after minimization

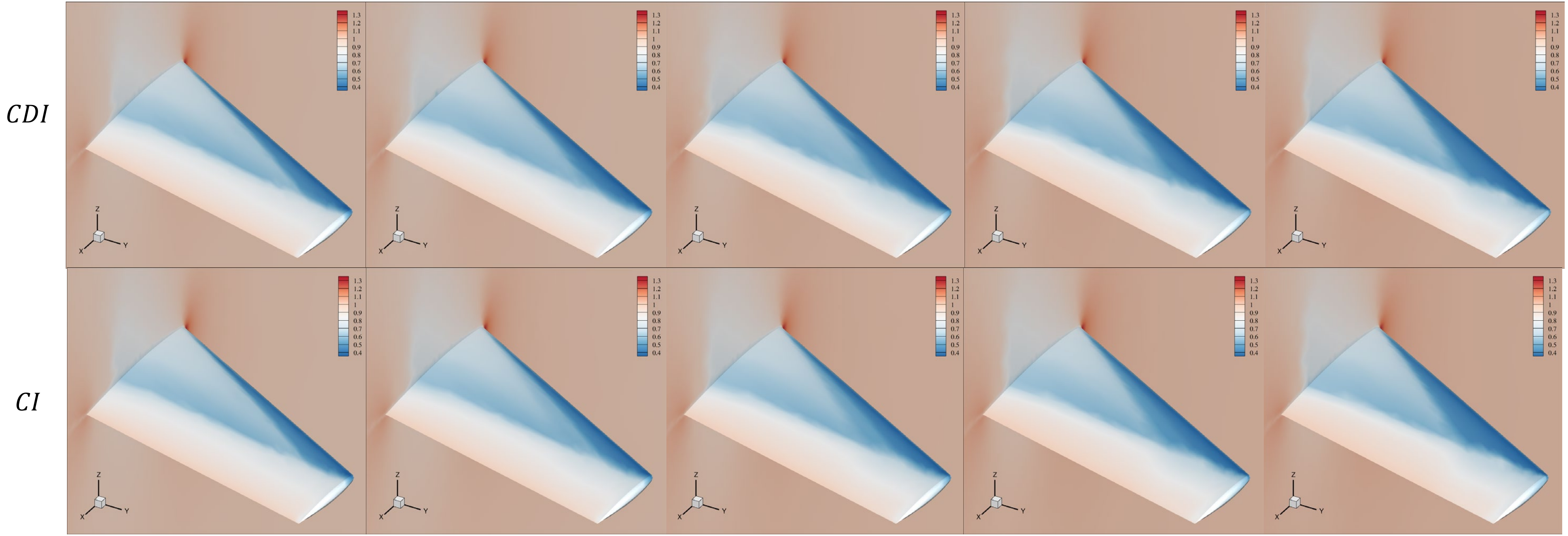
$$\Xi = \{(\xi_j)_{j=1}^{N_0} \in \mathcal{T}_{DG} \ : \ \varsigma_S(U_0^h(\xi_j)) > 0.6\}$$
$$Y = \{(y_i)_{i=1}^{N_1} \in \mathcal{T}_{DG} \ : \ \varsigma_S(U_1^h(y_i)) > 0.6\}$$



$$\Phi = X(\cdot, 1; \boldsymbol{\alpha}^\star)$$



CDI and CI Interpolations



$t = 0{,}0$ $AoA = 3{,}06$

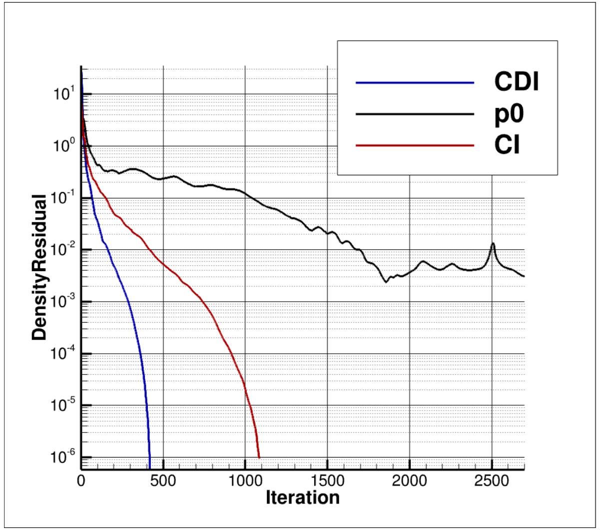
$t = 0{,}25$ $AoA = 3{,}57$

$t = 0{,}5$ $AoA = 4{,}08$

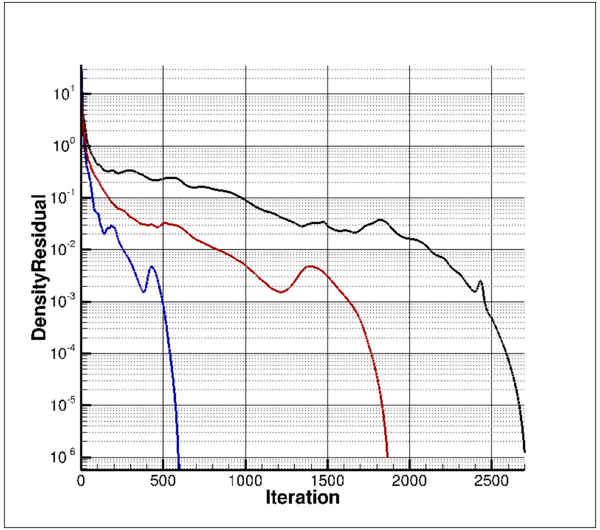
$t = 0{,}75$ $AoA = 4{,}59$

$t = 1$ $AoA = 5{,}10$

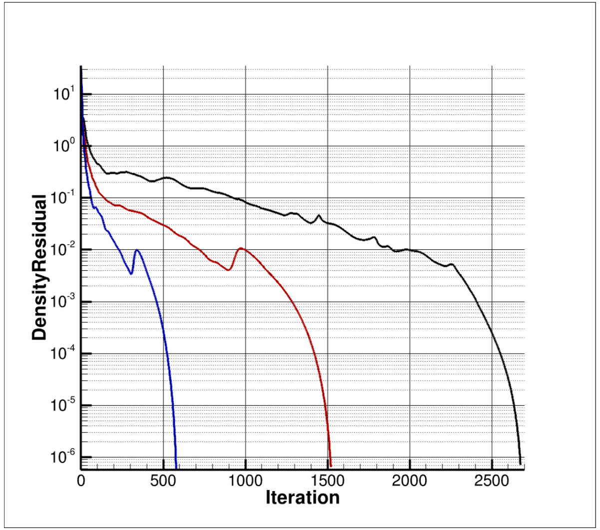
CODA p1 initialization from predictions



$t = 0{,}25$
$AoA = 3{,}57$



$t = 0{,}5$
$AoA = 4{,}08$



$t = 0{,}75$
$AoA = 4{,}59$

Conclusions

- *How to define a method to compute diffeomorphic mappings capable of aligning coherent structures of 3D compressible viscous flows in the context of industrial reduced-order models for large-query simulations ?*

Modelization

Energy Functional

Research Space

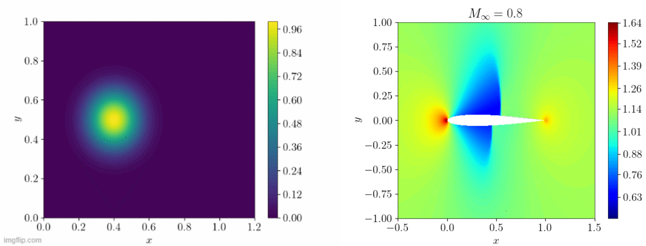
Minimization strategy

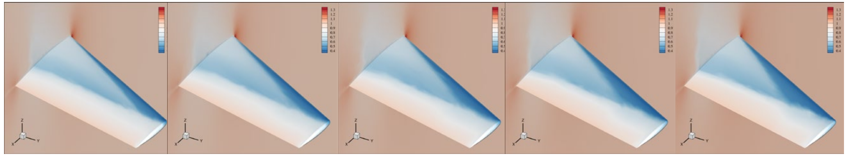
$$\Phi(\Omega) = \Omega$$

$$\Phi \in C^1(\Omega)$$

$$\Phi^{-1} \circ \Phi = \mathrm{Id}$$

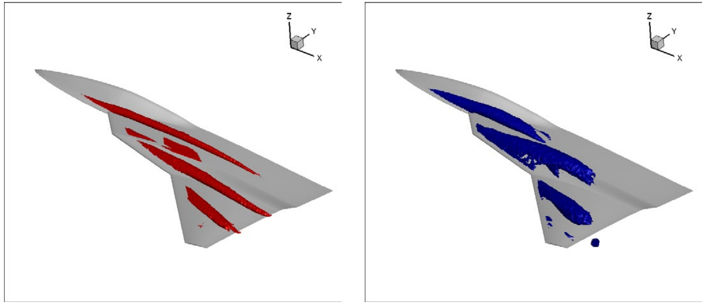
Data: $\Xi, Y, \varepsilon, \sigma_0, \boldsymbol{P}^{\varsigma}$
Result: velocity FE dof $\boldsymbol{\alpha}$
Initialization;
$\boldsymbol{\alpha} \leftarrow 0$;
$\sigma \leftarrow \sigma_0$;
$k \leftarrow 0$;
Optimization;
while $|\Delta\sigma_k^2| > \varepsilon$ **do**
E-step;
$P_{ij} \leftarrow (2.24)$;
M-step ;
$v^h = \sum_{p=1}^{N_{\mathrm{dof}}} \alpha_p \ell_p$;
$\hat{X}(\xi_i, 1) \leftarrow \xi_i + \sum_{m=1}^{N_{RK}} \omega_m v^h(\hat{X}(y_m, t_m), t_m)$;
$\widehat{\nabla X}(\xi_i, 1) \leftarrow \mathrm{Id} + \sum_{m=1}^{N_{RK}} \omega_m \nabla v^h(\hat{X}(y_m, t_m), t_m) \widehat{\nabla X}(\hat{X}(y_m, t_m), t_m)$;
$\hat{f}_n \leftarrow \int_0^1 \sum_{i,j=1}^{N_0, N_1} P_{ij} \left\langle X(\xi_i, 1) - y_j, \nabla X(\xi_i, 1)(\nabla X(\xi_i, \tau))^{-1} \ell_n(X(\xi_i, \tau)) \right\rangle_2 d\tau$;
$\boldsymbol{\alpha}_{k+1} \leftarrow \boldsymbol{\alpha}_k - \eta_k \left(\boldsymbol{K}_h^{-1} \boldsymbol{f}(\boldsymbol{\alpha}_k) + \lambda \boldsymbol{\alpha}_k\right)$; /* η from line search */
$\sigma_{k+1}^2 \leftarrow (2.28)$;
$\Delta\sigma_{k+1}^2 \leftarrow \sigma_{k+1}^2 - \sigma_k^2$;
$k \leftarrow k + 1$;
end





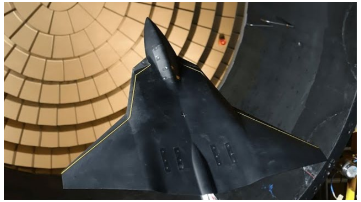
Future Application

Test Case: SUPERMAN

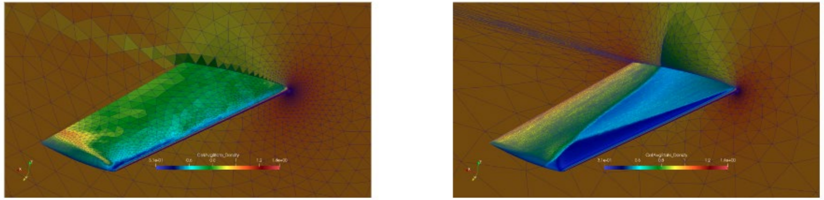


$AoA = 10°$ $AoA = 16°$

Application on experimental data



Mesh adaptation

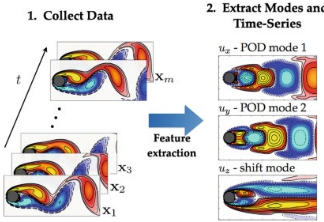


(a) initial computation, 2732120 DOFS (b) C=150000, iteration=20, 2517236 DOFS

Figure 20: ONERA-M6: visualisation of initial and final adapted meshes colored by density contours for DG-$p = 1$.

Perspectives

Lagrangian-based ROM



Data augmentation

POD Modes