CSMA 2024
16ème Colloque National en Calcul des Structures
13-17 Mai 2024, Presqu'île de Giens (Var)

Morphing techniques for model order reduction with non parametric geometrical variabilities

A. Kabalan[1,2], F. Casenave[2], F. Bordeu[2], V. Ehrlacher[1], A. Ern[1]

[1] *Cermics, Ecole des Ponts ParisTech, Cité Descartes, 8 Av. Blaise Pascal, 77420 Champs-sur-Marne, FRANCE,*
{abbas.kabalan, virginie.ehrlacher, alexandre.ern}@enpc.fr
[2] *Safran Tech, Digital Sciences & Technologies, 78114 Magny-Les-Hameaux, FRANCE,*
{fabien.casenave, felipe.bordeu}@safrangroup.com

Résumé — In this work, we study the construction of reduced order basis in the presence of non parametric geometrical variabilities. We present a new morphing algorithm that is adapted to the task of model order reduction. This is done by finding the morphing optimally in the sense that the relative energy contained in an arbitrary number of eigenvalues of the correlation matrix is maximal. In addition, the morphing could be integrated in non intrusive model order reduction techniques.
Mots clés — model order reduction, mesh morphing, data compression.

1 Introduction

Very often in physical science and engineering, we are required to solve a parametric partial differential equation (PDE) for numerous value of the parameter μ, such as in the context of design, optimization and inverse problems. The parameter $\mu \in \mathcal{P}$ could represent initial and boundary values, physical properties of the domain Ω_0 where the solution is defined, or in a more interesting case, geometrical parameters of the domain. When the solution of the PDE is expensive to evaluate, we can resort to model order reduction techniques to accelerate the computation of the solutions while maintaining a good accuracy. The solution manifold is defined as $\mathcal{M} = \{u_\mu | \mu \in \mathcal{P}\}$. The reduced-order basis method [1] consists in the construction of a low dimensional approximation space $\mathcal{Z}_r = \text{span}\{\xi_1, \cdots, \xi_r\}$ of the solution manifold, where we can calculate the solution more rapidly. The use of these methods starts usually by the construction of the reduced-order basis $\mathcal{Z}_r$ by an approximation algorithm such as the singular value decomposition (SVD) or the proper orthogonal decomposition (POD). However, when using these construction methods, it is required that all the snapshots are defined on the same domain Ω_0.

Context

Let $n \in \mathbb{N}^*$ and $\Omega_1, \ldots, \Omega_n \subset \mathbb{R}^d$ to be n distinct domain with $d = 2, 3$. We suppose that no parametrization is known for the domains. Suppose also that $\forall i$, there exists a scalar field u_i defined on Ω_i. These fields are solutions to a PDE for a given parameter value μ. In the following study, we will focus only on the data compression step. Various techniques could then be used for the construction of the reduced model such as Galerkin projection methods (POD-Galerkin) or by using Gaussian process regression (GPR) where we construct a reduced order basis from the snapshots $(u_i)_{1 \leq i \leq n}$ using the snapshot-POD algorithm, then we train a algorithm by taking the parameter μ as input and the coefficient of the reduced basis as output.
Since each snapshot is defined on a different domain Ω_i, we can resort to finding a morphing ϕ_i from a reference domain Ω_0 to the physical domain Ω_i, where we can determine the reduced basis on the parameter independent domain Ω_0. Thus we seek to construct a family of morphings $\Phi = (\phi_i)_{1 \leq i \leq n}$ with :

$$\phi_i : \Omega_0 \to \Omega_i, \quad 1 \leq i \leq n, \tag{1}$$

is a morphing from a reference domain Ω_0 to Ω_i. In this setting, the reduced order basis could now be constructed from the mapped snapshots $(u_i \circ \phi_i)_{1 \leq i \leq n}$ [1] defined now on Ω_0.
In the same fashion, when dealing with non reducible problems which are characterized by a slowly decaying Kolmogorov N-width of the solution manifold, the use of linear model order reduction techniques might not be sufficient. One approach in this case is to find morphings as in (1), such that the mapped snapshots $(u_i \circ \phi_i)_{1 \leq i \leq n}$ are more amenable to linear data compression. A special case is when $\Omega_i = \Omega_0$ $\forall i$.
The problem of constructing reduced order models with shape variabilities is the subject of numerous studies in the literature. Researchers relied on finding the morphings using free form deformation (FFD) [2]-[3], radial basis function interpolation [4]-[5] or linear elasticity mesh morphing approaches [6]. In these works, the geometries are assumed to be parameterized and the deformation of the boundary is given. The problem of finding the morphing to a reference geometry for non parameterized geometries proves to be more difficult, especially for problems in 3D. In [7], the authors construct a reduced order model on a reference geometry and then transport the reduced basis to each physical domain. The morphing is computed using the LDDMM method [8]. A similar approach is employed in [9] for inverse problems, where the authors construct a dictionary based reduced order model. In [10], the snapshot-POD is coupled with Gaussian process regression to solve the forward reduced order problem in the case of non parameterized geometries. Data compression for non reducible problem is also studied heavily in the literature (see [11] and references within). In this work, we aim to develop a new general morphing technique that could be adopted to a wide range of problems, including the morphing between non parameterized geometries. In addition, we aim to couple this morphing approach with an optimization algorithm to improve the data compression for non reducible problems.
The rest of this paper is organized as follows. In section 2, we present the elastic shape matching algorithm and our variation that is more adapted to our context. In section 3, we introduce the algorithm to align the different snapshots on an arbitrary number of modes. We present some concluding remarks and future work in section 4.

2 Elastic shape matching

The elastic shape matching is a morphing technique proposed in [12]. To map a domain Ω onto a target domain Ω_0, the authors solve a shape optimization problem to minimize the functional :

$$J(\Omega) = \int_{\Omega} d_{\Omega_0}(x) dx,$$

where d_{Ω_0} is the Euclidean signed distance function to the domain Ω_0 defined as :

$$d_{\Omega_0}(x) = \begin{cases} -d(x, \partial\Omega_0) & \text{if } x \in \Omega_0 \\ d(x, \partial\Omega_0) & \text{if } x \in \overline{{}^c\Omega_0} \\ 0 & \text{if } x \in \partial\Omega_0 \end{cases} \tag{2}$$

and $d(x, \partial\Omega_0)$ is the Euclidean distance from x to the boundary of Ω_0. The functional J has its unique global minimum for $\Omega = \Omega_0$. Taking the shape derivative of J (see [13]), we get :

$$J'(\Omega)(v) = \int_{\partial\Omega} d_{\Omega_0} v \cdot n ds.$$

To obtain a gradient descent direction, we chose v that satisfies the linear elasticity problem :

$$\int_{\Omega} \sigma(u_\Omega) : \varepsilon(v) dx = -J'(\Omega)(v) = -\int_{\partial\Omega} d_{\Omega_0} v \cdot n ds. \tag{3}$$

The above problem is solved iteratively, until we reach the global minimum of J where $\Omega = \Omega_0$. We thus note $\Omega^{(m)}$ the mapped domain after m iterations and we have $\Omega^{(M)} = \Omega_0$ with M being the final iteration.

1. This is called pullback transformation. The Piola transformation which conserves the flux of a tensor field across boundaries could also be used.

We start from $\Omega^{(0)} = \Omega$ and $u_\Omega^{(0)} = id$. Then at each iteration $1 \leq m \leq M$, we calculate $u_\Omega^{(m)}$, the solution of :

$$\forall v \in [H^1(\Omega^{(m)})]^d, \int_{\Omega^{(m)}} \sigma(u_\Omega^{(m)}) : \varepsilon(v) dx = -\int_{\partial\Omega^{(m)}} d_{\Omega_0} v \cdot n ds,$$

that maps $\Omega^{(m)}$ to $\Omega^{(m+1)}$. The mapping $\phi^{(m)}$ from Ω to $\Omega^{(m)}$ is given by the induction :

$$\forall x \in \Omega, \quad \phi^{(m)}(x) = \phi^{(m-1)}(x) + u_\Omega^{(m)} \circ \phi^{(m-1)}(x) = (Id + u_\Omega^{(m)}) \circ \phi^{(m-1)}(x).$$

A major advantage of this approach over other morphing techniques used in the same context, is that we do not have to calculate explicitly the morphing of the boundary of the domain Ω_0. This is extremely useful especially for non parameterized geometries in a 3D settings.
For problem (3) to be well posed, we should impose Dirichlet boundary conditions, which can restrict the displacement of some nodes. In addition, we would like to add the ability to map certain points, lines and surfaces of interest in the reference domain to their correspondence in the target domain.

2.1 Vectorial distance formulation

We present here the variation of the elastic shape matching method we propose in this work in 2D. The extension to 3D domains is straightforward. We will refer to the left hand side of (3) as the *operator term* and the right hand side as the *matching term*.
Operator term variation : we add to the linear elasticity operator, a stiffness operator on $\partial\Omega$ defined as :

$$\alpha \int_{\partial\Omega} u_\Omega \cdot n \, v \cdot n dx,$$

with $\alpha \in \mathbb{R}^+$. This term has the effect of imposing, in a penalized sense, the Dirichlet boundary condition $u_\Omega \cdot n = 0$ on the boundary of Ω. In this case, all nodes are free to move in order to achieve the matching between the two domains.

Matching term variation : In our context, we suppose that the boundary of Ω is partitioned as $\partial\Omega = \Gamma_1 \cup \Gamma_2 \ldots \cup \Gamma_s$, with additional tagged points of interest $\mathrm{x}_1^*, \cdots, \mathrm{x}_{n_{points}}^*$ on its boundary. In addition, for each Γ_i (point x_j^*), there exist its corresponding $\tilde{\Gamma}_i$ (point $\tilde{\mathrm{x}}_j^*$) on $\partial\Omega_0$. Define $\mathbf{D}_{\Omega_0}$ as :

$$\begin{aligned} \mathbf{D}_{\Omega_0} : \quad & \partial\Omega \to \mathbb{R}^2 \\ & x \to \mathbf{D}_{\Omega_0}(x) = \sum_{i=1}^{s} (\Pi_{\tilde{\Gamma}_i}(x) - x) \mathbf{1}_{\Gamma_i}(x), \end{aligned} \tag{4}$$

with $\Pi_A(x)$ is the projection of point x on set A, $\Pi_A(x) = \arg\min_{y \in A} ||x-y||_2$, $\mathbf{1}_A$ is the indicator function of set A, and $\tilde{\Gamma}_i \subset \Omega_0$ is where we want Γ_i to be mapped.
We replace the matching term in (3) by :

$$\beta \sum_{i=1}^{n_{points}} \int_{\partial\Omega(\mathrm{x}_i^*)} (\tilde{\mathrm{x}}_i^* - \mathrm{x}_i^*) \cdot v ds + \gamma \int_{\partial\Omega} \mathbf{D}_{\Omega_0} \cdot n v \cdot n ds,$$

with $\beta, \gamma \in \mathbb{R}^+$, and $\partial\Omega(\mathrm{x}_i^*)$ is a small neighborhood of x_i^*. These terms will force points x_i^* to be mapped to $\tilde{\mathrm{x}}_i^*$ and each Γ_i to be mapped to $\tilde{\Gamma}_i$.

Thus formulation (3) is replaced by :

$$\int_\Omega \sigma(u_\Omega) : \varepsilon(v) dx + \alpha \int_{\partial\Omega} u_\Omega \cdot n \, v \cdot n dx = \beta \sum_{i=1}^{n_{points}} \int_{\partial\Omega(\mathrm{x}_i^*)} (\tilde{\mathrm{x}}_i^* - \mathrm{x}_i^*) \cdot v ds + \gamma \int_{\partial\Omega} \mathbf{D}_{\Omega_0} \cdot n v \cdot n ds. \tag{5}$$

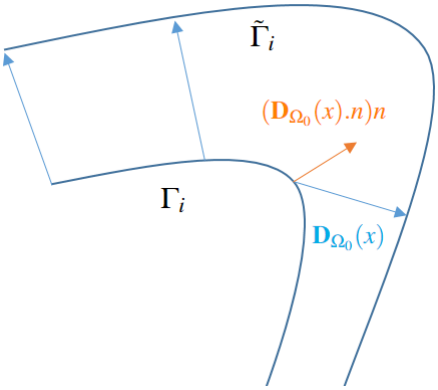


FIGURE 1 – Visualisation of (4). For each point $x \in \Gamma_i$, $\mathbf{D}_{\Omega_0}(x)$ is the vector that points from x to the closest point (the projection) of x on $\tilde{\Gamma}_i$

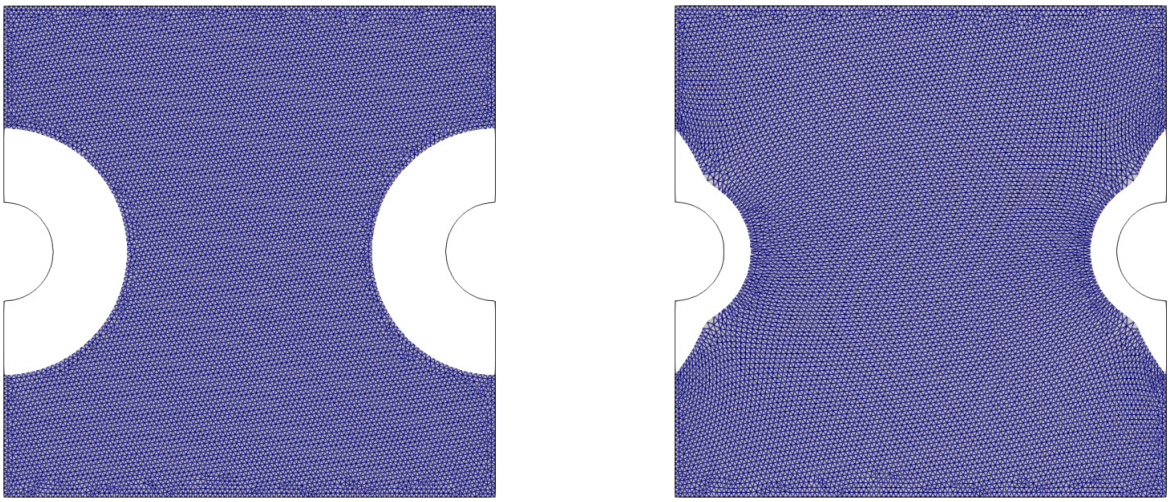


FIGURE 2 – The original domain and the morphed domain after a few iterations using (3).

2.2 Numerical results

We apply the original elastic shape matching to the geometry shown in Figure 2. On the left, we have a mesh of the reference geometry and the boundary of the target geometry in black. On the right, we see the results of the morphing using (3). By using the original elastic shape matching algorithm, we do not specify any constraint on the mapping of the boundary. In this particular example, the algorithm fails to converge .
In Figure 3, we specify that the corners and the semi circles in the reference domain should be mapped to their correspondence in the target domain using our method with the vectorial distance formulation as in (5).

3 Optimal Morphing

Define $\mathbf{T}_i$ as the space of bijective morphings from Ω_0 to Ω_i and $\mathbf{T} = \mathbf{T}_1 \times \mathbf{T}_2 \times \cdots \times \mathbf{T}_n$. The correlation matrix will be defined as :

$$\begin{aligned} C: \quad & \mathbf{T} \to \mathcal{M}_n(\mathbb{R}) \\ & \Phi \mapsto C[\Phi] := (C_{ij}[\Phi])_{1\le i,j\le n} \end{aligned} \tag{6}$$

with $C_{ij}[\Phi] = \langle u_i \circ \phi_i, u_j \circ \phi_j \rangle_{L^2(\Omega_0)} = \int_{\Omega_0} u_i \circ \phi_i(x) u_j \circ \phi_j(x) dx$, for $\Phi = (\phi_i)_{1\le i\le n}$.

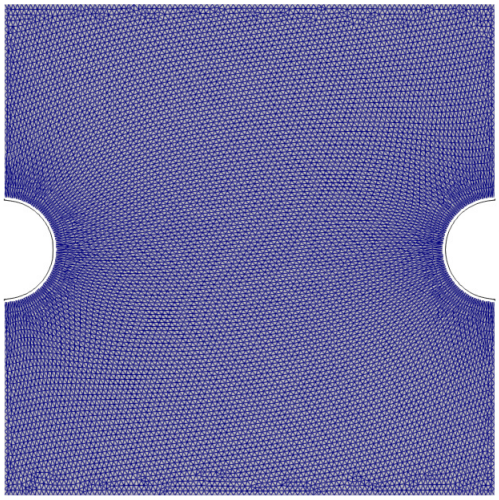


FIGURE 3 – The result of the morphing using the vectorial distance formulation (5)

For a given $\Phi \in \mathbf{T}$, we designate $\lambda_1^\Phi \geq \lambda_2^\Phi \geq \cdots \geq \lambda_n^\Phi$ as the eigenvalues of $C[\Phi]$, and $\zeta_1^\Phi, \zeta_2^\Phi, \cdots, \zeta_n^\Phi$ as the corresponding eigenvectors. Finally define the functional :

$$\begin{aligned} J: \quad & \mathbf{T} \to \mathbb{R} \\ & \Phi \mapsto J[\Phi] := \frac{\sum_{j=1}^{r} \lambda_j^\Phi}{Tr(C[\Phi])}, \end{aligned} \tag{7}$$

which represents the cumulative energy in the first r modes. We seek to solve the following maximization problem :

$$\text{find } \Phi^* \in \arg\max_{\Phi \in \mathbf{T}} J[\Phi]$$

Differential of J

Let $\Phi = (\phi_i)_{1\leq i\leq n} \in \mathbf{T}$, $\Psi = (\psi_i)_{1\leq i\leq n}$ a small variation around Φ and $\varepsilon \in \mathbb{R}, \varepsilon << 1$. To evaluate the differential of J at a point Φ, we evaluate J at $\bar{\Phi} := \Phi + \varepsilon\Psi$ and calculate :

$$\delta J[\Phi][\Psi] = \lim_{\varepsilon \to 0} \frac{J(\Phi + \varepsilon\Psi) - J(\Phi)}{\varepsilon}. \tag{8}$$

There holds

$$\begin{aligned} \forall 1 \leq i \leq n, \;\; u_i \circ (\phi_i + \varepsilon\psi_i)(x) &= u_i(\phi_i(x) + \varepsilon\psi_i(x)) \\ &\simeq u_i(\phi_i(x)) + \varepsilon\vec{\nabla}u_i(\phi_i(x)) \cdot \psi_i(x), \end{aligned}$$

where we neglect higher order terms. Now we evaluate :

$$\begin{aligned} C_{ij}[\Phi + \varepsilon\Psi] &= \langle u_i \circ (\phi_i + \varepsilon\psi_i), u_j \circ (\phi_j + \varepsilon\psi_j) \rangle_{L^2(\Omega_0)} \\ &= \int_{\Omega_0} u_i \circ (\phi_i + \varepsilon\psi_i)(x) u_j \circ (\phi_j + \varepsilon\psi_j)(x) dx \\ &= \int_{\Omega_0} u_i \circ \phi_i(x) u_j \circ \phi_j(x) dx + \varepsilon \int_{\Omega_0} u_i \circ \phi_i(x) \vec{\nabla} u_j(\phi_j(x)) \cdot \psi_j(x) dx \\ &+ \varepsilon \int_{\Omega_0} \vec{\nabla} u_i(\phi_i(x)) \cdot \psi_i(x) u_j \circ \phi_j(x) dx + \varepsilon^2 \int_{\Omega_0} \vec{\nabla} u_i(\phi_i(x)) \psi_i(x) \vec{\nabla} u_j(\phi_j(x)) \psi_j(x) dx \\ &\simeq C_{ij}[\Phi] + \varepsilon\delta C_{ij}[\Phi][\Psi] + o(\varepsilon), \end{aligned} \tag{9}$$

and

$$\begin{aligned} Tr(C[\Phi+\varepsilon\Psi]) &= \sum_{i=1}^{n} C_{ii}[\Phi+\varepsilon\Psi] \\ &= Tr(C[\Phi]) + 2\varepsilon\sum_{i=1}^{n}\int_{\Omega_0} u_i\circ\phi_i(x)\vec{\nabla}u_i(\phi_i(x))\cdot\psi_i(x)dx + \varepsilon^2\sum_{i=1}^{n}\int_{\Omega_0}[\vec{\nabla}u_i(\phi_i(x))\cdot\psi_i(x)]^2dx \\ &\simeq Tr(C[\Phi]) + \varepsilon Tr(\delta C[\Phi][\Psi]), \end{aligned}$$

with $\delta C_{ij}[\Phi][\Psi] = \int_{\Omega_0} u_i\circ\phi_i(x)\vec{\nabla}u_j(\phi_j(x))\cdot\psi_j(x)dx + \int_{\Omega_0}\vec{\nabla}u_i(\phi_i(x))\cdot\psi_i(x)u_j\circ\phi_j(x)dx$ and $\delta C[\Phi][\Psi] = (\delta C_{ij}[\Phi][\Psi])_{1\le i,j\le n}$.

Next we evaluate $\delta\lambda_i^{\Phi}[\Psi]$. There holds

$$\forall 1\le i\le n,\ \ ||\zeta_i^{\Phi}||^2 = 1,$$

By computing the differential :

$$\langle\delta\zeta_i^{\Phi}[\Psi],\zeta_i^{\Phi}\rangle = 0, \tag{10}$$

using the fact that $C[\Phi]\zeta_i^{\Phi} = \lambda_i^{\Phi}\zeta_i^{\Phi}$, and computing again the differential :

$$\delta C[\Phi][\Psi]\zeta_i^{\Phi} + C[\Phi]\delta\zeta_i^{\Phi}[\Psi] = \delta\lambda_i^{\Phi}[\Psi]\zeta_i^{\Phi} + \lambda_i^{\Phi}\delta\zeta_i^{\Phi}[\Psi].$$

We multiply the last equation by ζ_i^{Φ} and we use (10) to get :

$$(\zeta_i^{\Phi})^T\delta C[\Phi][\Psi]\zeta_i^{\Phi} + 0 = \delta\lambda_i^{\Phi}[\Psi] + 0.$$

Thus we have finally :

$$\lambda_i^{\Phi+\varepsilon\Psi} \simeq \lambda_i^{\Phi} + (\zeta_i^{\Phi})^T\delta C[\Phi][\Psi]\zeta_i^{\Phi} \quad ,\forall 1\le i\le n. \tag{11}$$

Taking the sum over the first r eigenvalues, we obtain :

$$\sum_{j=1}^{r}\lambda_j^{\Phi+\varepsilon\Psi} \simeq \sum_{j=1}^{r}\lambda_j^{\Phi} + \varepsilon Tr((Z_r^{\Phi})^T\delta C[\Phi][\Psi]Z_r^{\Phi}), \tag{12}$$

with $Z_r^{\Phi} = (\zeta_1^{\Phi},\zeta_2^{\Phi},\cdots,\zeta_r^{\Phi})^T$. Now we can evaluate (8) to obtain :

$$\delta J[\Phi][\Psi] = \frac{Tr((Z_r^{\Phi})^T\delta C[\Phi][\Psi]Z_r^{\Phi})}{Tr(C[\Phi])} - \frac{\sum_{k=1}^{r}\lambda_k^{\Phi}}{Tr(C[\Phi])^2}\times Tr(\delta C[\Phi][\Psi]), \tag{13}$$

which can be written explicitly as :

$$\begin{aligned} \delta J[\Phi][\Psi] &= \frac{2}{Tr(C[\Phi])}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{r}\zeta_{k,i}^{\Phi}\zeta_{k,j}^{\Phi}\int_{\Omega_0} u_j\circ\phi_j(x)\vec{\nabla}u_i(\phi_i(x))\cdot\psi_i(x)dx \\ &\quad - \frac{2\sum_{k=1}^{r}\lambda_k^{\Phi}}{Tr(C[\Phi])^2}\sum_{i=1}^{n}\int_{\Omega_0} u_j\circ\phi_j(x)\vec{\nabla}u_i(\phi_i(x))\cdot\psi_i(x)dx \\ &= \sum_{i=1}^{n}\delta J_i[\Phi][\psi_i], \end{aligned}$$

with :

$$\begin{aligned} \delta J_i[\Phi][\psi_i] &= \sum_{k=1}^{r}\Big(\frac{2{\zeta_{k,i}^{\Phi}}^2}{Tr(C[\Phi])} - \frac{2\lambda_k^{\Phi}}{Tr(C[\Phi])^2}\Big)\int_{\Omega_0} u_i\circ\phi_i(x)\vec{\nabla}u_i(\phi_i(x))\cdot\psi_i(x)dx \\ &\quad + \sum_{\substack{j=1\\ j\neq i}}^{n}\sum_{k=1}^{r}\frac{2\zeta_{k,i}^{\Phi}\zeta_{k,j}^{\Phi}}{Tr(C[\Phi])}\int_{\Omega_0} u_j\circ\phi_j(x)\vec{\nabla}u_i(\phi_i(x))\cdot\psi_i(x)dx. \end{aligned} \tag{14}$$

Gradient direction

To get a gradient direction of equation (7), we will determine Ψ by solving the n linear elasticity problems :

$$\int_{\Omega_0} \sigma(\psi_i) : \varepsilon(v)dx + \alpha \int_{\partial\Omega_0} \psi_i \cdot n \, v \cdot n dx = \delta J_i[\Phi][v] \ , \ 1 \leq i \leq n. \tag{15}$$

The above set of equations is solved iteratively, where at each step m we determine $\Psi^{(m)} = (\psi_i^{(m)})_{1\leq i\leq n}$ by solving :

$$\int_{\Omega_0} \sigma(\psi_i^{(m)}) : \varepsilon(v)dx + \alpha \int_{\partial\Omega_0} \psi_i^{(m)} \cdot n \, v \cdot n dx = \delta J_i[\Phi^{(m)}][v] \ , \ 1 \leq i \leq n. \tag{16}$$

and then we set $\Phi^{(m+1)} = \Phi^{(m)} + \varepsilon\Psi^{(m)}$.

Notes

1. For $1 \leq i \leq n$, we start by determining the initial morphing $\phi_i^{(0)}$ that will map Ω_0 to Ω_i. This could be done using elastic shape matching, RBF morphing or other morphing techniques.
2. Additionally, if $\Omega_i = \Omega_0$. We simply take $\phi_i^{(0)} = id$.
3. The fields $(u_i)_{1\leq i\leq n}$ could also be the coordinates fields used for the shape embedding as in [10]. This is useful to reduce the number of inputs in a POD-GPR model order reduction framework.

Numerical results

We test the above method on a dataset with $n = 30$ and $u_i(x,y) = exp - \left(\frac{(x_0(x+1)-y)^2}{0.05}\right)$ while varying x_0. In Figure 4, we see on the left two snapshot before optimization. We chose $r = 1$, that is we seek to construct a reduced order basis of dimension 1. On the right, we see the two snapshots aligned after the morphing.

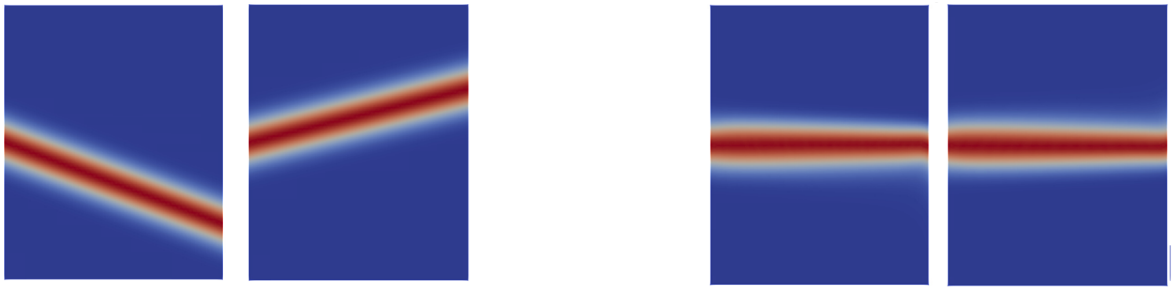
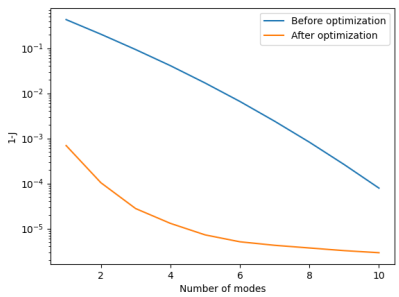
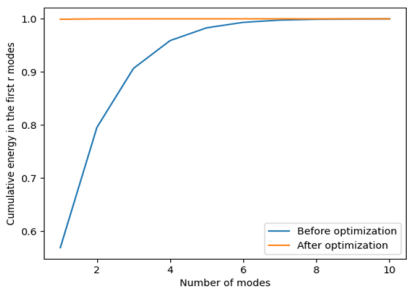


FIGURE 4 – Two snapshots before and after optimization



(a) $1 - J[\Phi]$ for different values of r. Logarithmic scale



(b) The energy (7) before and after optimization for different values of r.

FIGURE 5 – Comparison of the energy (7) before and after optimization. Notice how after the optimization, all the energy is now placed in the first mode. This means that after morphing, a reduced-order basis of cardinal 1 is now accurate.

4 Conclusion

In this work, we present a morphing technique based on the elastic shape matching that is suitable to model order reduction with non parameterized geometries. In section 3, we present an algorithm that maximizes data compression. Preliminary results show interesting gains. The algorithm is presented only in the offline phase. Calculating this morphing in the online phase is yet to be determined. Future works consist of extending the two methods to 3D cases, test the optimal morphing algorithm for the variable geometries case, and integrating these morphing algorithms into online-efficient model-order reduction framework.

Références

[1] Alfio Quarteroni, Andrea Manzoni, and Federico Negri. *Reduced basis methods for partial differential equations : an introduction*, volume 92. Springer, 2015.

[2] Thomas W Sederberg and Scott R Parry. Free-form deformation of solid geometric models. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 151–160, 1986.

[3] Filippo Salmoiraghi, Angela Scardigli, Haysam Telib, and Gianluigi Rozza. Free-form deformation, mesh morphing and reduced-order methods : enablers for efficient aerodynamic shape optimisation. *International Journal of Computational Fluid Dynamics*, 32(4-5) :233–247, 2018.

[4] Aukje De Boer, Martijn S Van der Schoot, and Hester Bijl. Mesh deformation based on radial basis function interpolation. *Computers & structures*, 85(11-14) :784–795, 2007.

[5] Nicola Demo, Marco Tezzele, Andrea Mola, and Gianluigi Rozza. Hull shape design optimization with parameter space and model reductions, and self-learning mesh morphing. *Journal of Marine Science and Engineering*, 9(2) :185, 2021.

[6] Andrea Manzoni and Federico Negri. Efficient reduction of pdes defined on domains with variable shape. *Model Reduction of Parametrized Systems*, pages 183–199, 2017.

[7] Romain Guibert, Kristin Mcleod, Alfonso Caiazzo, Tommaso Mansi, Miguel A Fernández, Maxime Sermesant, Xavier Pennec, Irene E Vignon-Clementel, Younes Boudjemline, and Jean-Frédéric Gerbeau. Group-wise construction of reduced models for understanding and characterization of pulmonary blood flows from medical images. *Medical image analysis*, 18(1) :63–82, 2014.

[8] M Faisal Beg, Michael I Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision*, 61 :139–157, 2005.

[9] Felipe Galarce, Damiano Lombardi, and Olga Mula. State estimation with model reduction and shape variability. application to biomedical problems. *SIAM Journal on Scientific Computing*, 44(3) :B805–B833, 2022.

[10] Fabien Casenave, Brian Staber, and Xavier Roynard. MMGP : a Mesh Morphing Gaussian Process-based machine learning method for regression of physical problems under non-parameterized geometrical variability. *arXiv preprint arXiv :2305.12871*, 2023.

[11] Tommaso Taddei. A registration method for model order reduction : data compression and geometry reduction. *SIAM Journal on Scientific Computing*, 42(2) :A997–A1027, 2020.

[12] Maya De Buhan, Charles Dapogny, Pascal Frey, and Chiara Nardoni. An optimization method for elastic shape matching. *Comptes Rendus Mathematique*, 354(8) :783–787, 2016.

[13] Grégoire Allaire, François Jouve, and Anca-Maria Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of computational physics*, 194(1) :363–393, 2004.