A Data-Driven Method for Parametric Reduced Order Models

Mourad OULGHELOU, Amine Ammar

Arts et Métiers ParisTech - Angers Campus

GDR-GDM Bordeaux, 27 June - 1st July, 2022

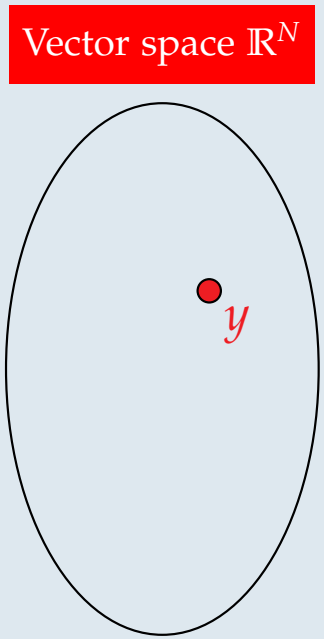
1st July 2022

Model reduction principle

- $y(t)$ solution at time t of the discrete PDE $\mathcal{N}_h(y, v) = 0$

Model reduction principle

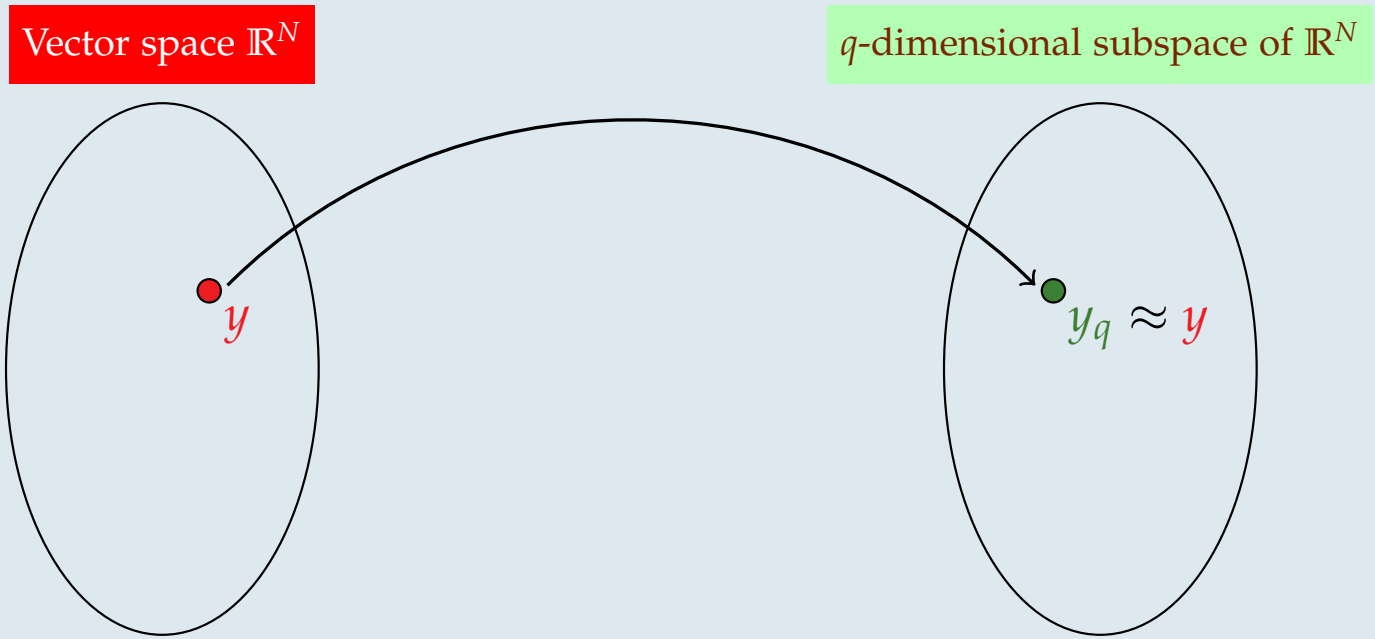
- $y(t)$ solution at time t of the discrete PDE $\mathcal{N}_h(y, v) = 0$



  - ▷ dimension : N
  - ▷ N : thousands, millions !!
  - ▷ basis $e_1, \ldots, e_N$
  - ▷ solution : $y = \sum_{i=1}^{N} y_i \, e_i$

Model reduction principle

- $y(t)$ solution at time t of the discrete PDE $\mathcal{N}_h(y, \nu) = 0$



▷ dimension : N
▷ N : thousands, millions !!
▷ basis $e_1, \ldots, e_N$
▷ solution : $y = \sum_{i=1}^{N} y_i \, e_i$

▷ dimension : $q \ll N$
▷ q : tens, hundreds
▷ reduced basis $\varphi^1, \ldots, \varphi^q$
▷ solution : $y_q = \sum_{i=1}^{q} \alpha^i \, \varphi^i$

Model reduction of Navier-Stokes equations

Solve NSE in a window $]0, T]$

$$\partial_t \mathbf{u} - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = f$$
$$\nabla \cdot \mathbf{u} = 0, \quad \mathbf{u}_{\partial\Omega} = 0, \quad \mathbf{u}(0) = \mathbf{u}_0$$

$$\longrightarrow \quad S = \{\mathbf{u}(t_1), \mathbf{u}(t_2), \ldots, \mathbf{u}(t_{N_s})\} \in \mathbb{R}^{dN \times N_s}$$

Correlation $C = \mathcal{A}_t^{\frac{1}{2}} S^T \mathcal{A}_x S \mathcal{A}_t^{\frac{1}{2}}$

Solve $CV = V\lambda \Rightarrow \varphi_i = S \mathcal{A}_t^{\frac{1}{2}} V \lambda_i^{-\frac{1}{2}}$

$\downarrow$

Galerkin proj. of NSE on $\varphi_1, \ldots, \varphi_q$

$$\left(\partial_t \mathbf{u}_q - \nu \Delta \mathbf{u}_q + \mathbf{u}_q \cdot \nabla \mathbf{u}_q - f, \varphi_i\right)_{L^2(\Omega)} + (\mathbf{r}, \varphi_i)_{L^2(\Omega)} = 0$$

$\longleftarrow$

Galerkin assumption : $\mathbf{r} \perp \phi_i$

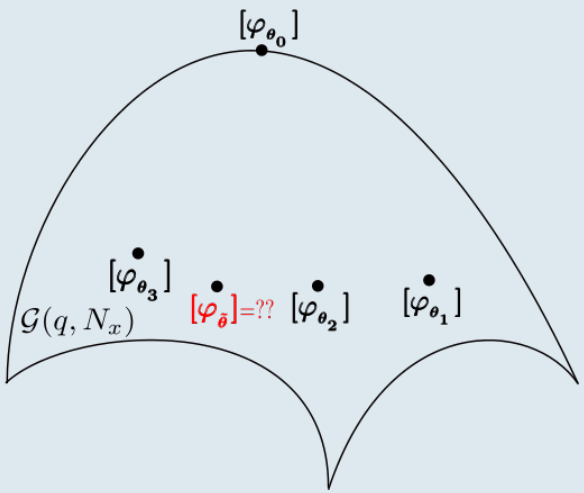
$$\Rightarrow \quad \dot{\alpha}_i + \nu R_{ij} \alpha_j + C_{ijk} \alpha_j \alpha_k = F_i$$
$$\alpha_i(0) = (u_0, \phi_i)_{L^2(\Omega)}$$

- d : dimension 2,3
- N : Number of space degrees of freedom (mesh vertices)
- N_s : Number of snapshots
- q : POD truncation order $q \ll N_s$
- $\mathcal{A}_x$ and $\mathcal{A}_t$: Space and time inner product operators

Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

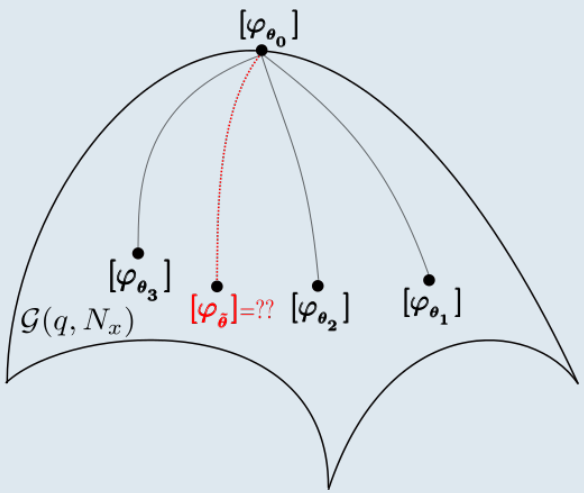
Handling parameterization



Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

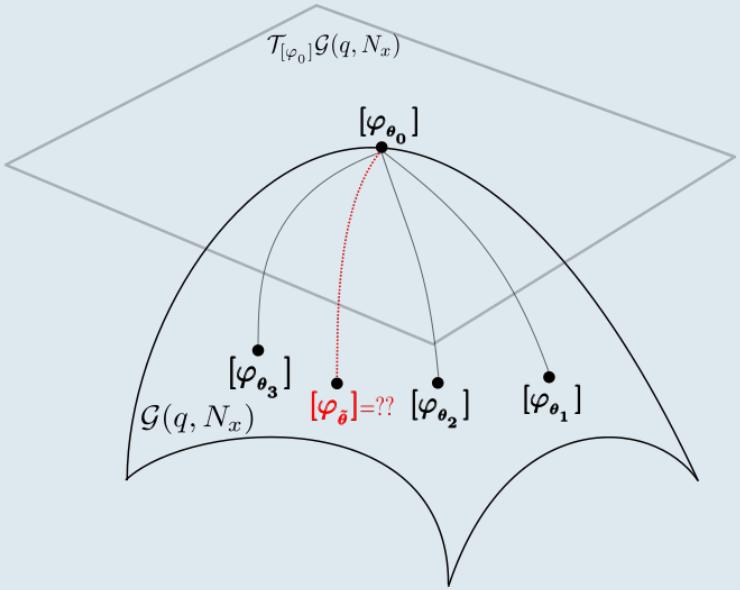
Handling parameterization



Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

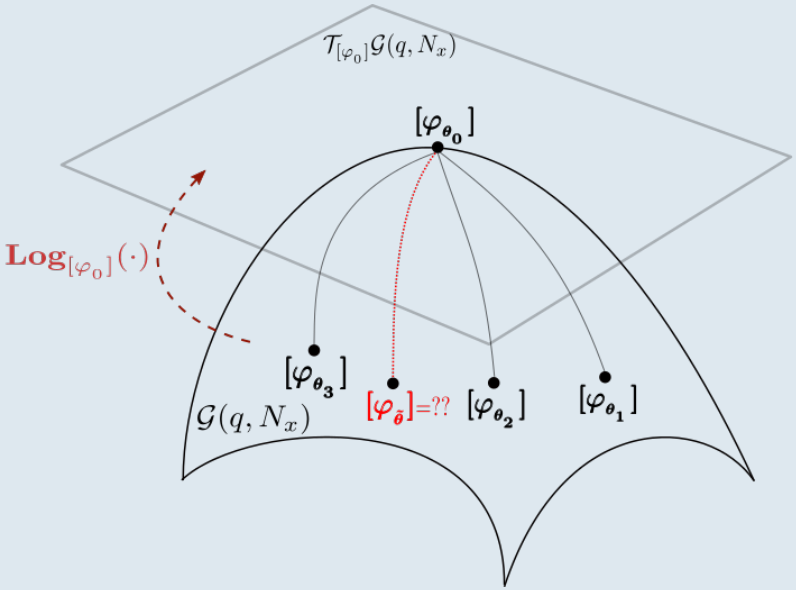
Handling parameterization



Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

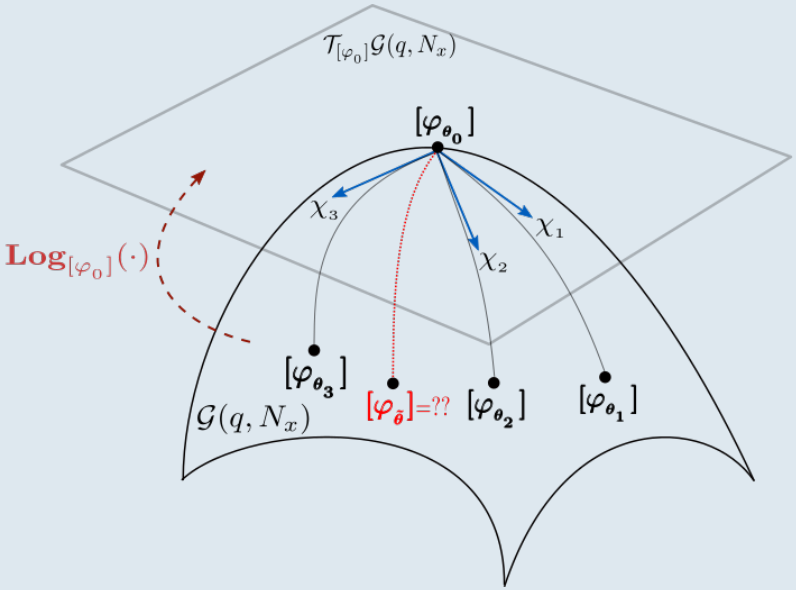
Handling parameterization



Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

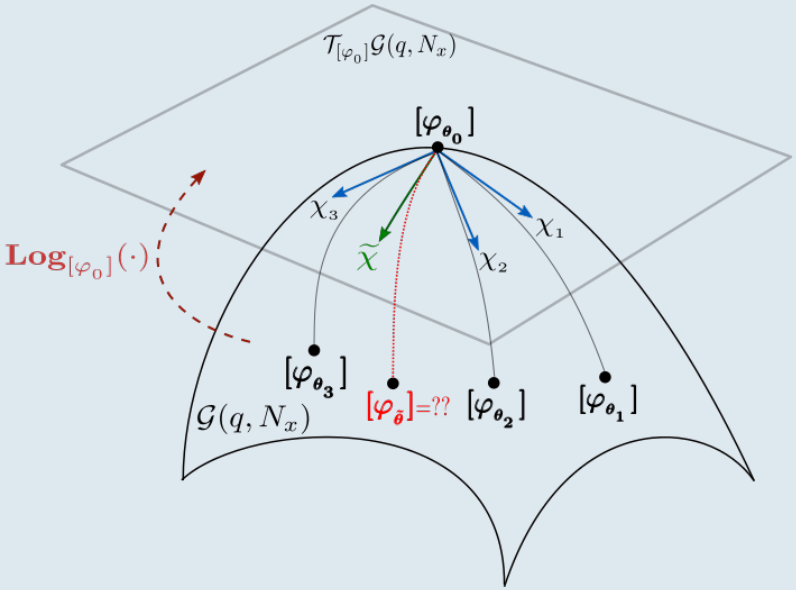
Handling parameterization



Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

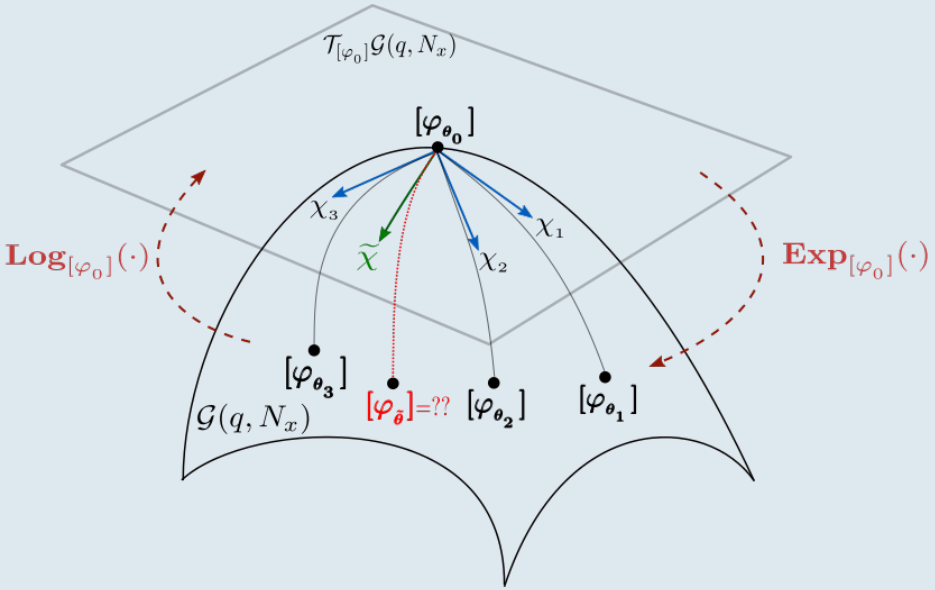
Handling parameterization



Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

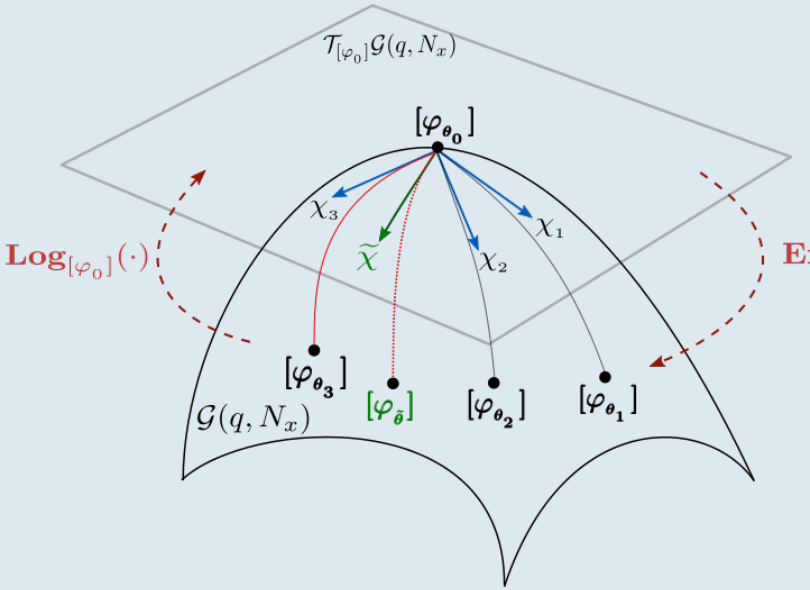
Handling parameterization



Problem setting

- Consider a set of parameterized snapshots mat. $S(\nu_1), S(\nu_2), \ldots, S(\nu_{N_p})$ and $[\varphi(\nu_1)], [\varphi(\nu_2)], \ldots, [\varphi(\nu_{N_p})]$ the corresponding subspaces belonging to a Riemannian manifold $\mathcal{M}$
  - ▷ ν_i are some discrete values of the parameter ν
- **Problem** : *" Find an approximation* $[\varphi(\widetilde{\nu})] \in \mathcal{M}$ *where* $\widetilde{\nu} \neq \nu_i$ *"*

Handling parameterization



Galerkin proj. on $\tilde{\varphi}_1, \ldots, \tilde{\varphi}_q$:

$$\left(\partial_t \tilde{\mathbf{u}}_q - \nu\, \Delta \tilde{\mathbf{u}}_q + \tilde{\mathbf{u}}_q \cdot \nabla \tilde{\mathbf{u}}_q - f, \tilde{\varphi}_i\right)_{L^2(\Omega)} + (\mathbf{r}, \tilde{\varphi}_i)_{L^2(\Omega)} = 0$$

Predicted reduced order model :

$$\dot{\tilde{\alpha}}_i + \nu\, R_{ij}\, \tilde{\alpha}_j + C_{ijk}\, \tilde{\alpha}_j\, \tilde{\alpha}_k = \tilde{F}_i$$

$$\tilde{\alpha}_i(0) = (u_0, \tilde{\varphi}_i)_{L^2(\Omega)}$$

Data pretreatment (1st POD stage)

Consider a set of parametrized snapshots matrices

$$S(\nu_k) = \left[u^1(\nu_k) \quad u^2(\nu_1) \quad \dots \quad u^N(\nu_k) \right], \quad \text{for } k = 1, \dots, p$$

for which we assume that the temporal derivatives $\dot{S}(\nu_k)$ are available

$$\dot{S}(\nu_k) = \left[\dot{u}^1(\nu_k) \quad \dot{u}^2(\nu_k) \quad \dots \quad \dot{u}^N(\nu_k) \right], \quad \text{for } k = 1, \dots, p$$

- N : number of time instants
- p : number of points in the parameter space

Data pretreatment (1st POD stage)

Consider a set of parametrized snapshots matrices

$$S(\nu_k) = \begin{bmatrix} u^1(\nu_k) & u^2(\nu_1) & \dots & u^N(\nu_k) \end{bmatrix}, \quad \text{for } k = 1, \dots, p$$

for which we assume that the temporal derivatives $\dot{S}(\nu_k)$ are available

$$\dot{S}(\nu_k) = \begin{bmatrix} \dot{u}^1(\nu_k) & \dot{u}^2(\nu_k) & \dots & \dot{u}^N(\nu_k) \end{bmatrix}, \quad \text{for } k = 1, \dots, p$$

- N : number of time instants
- p : number of points in the parameter space

For each ν_k, we apply the POD to the matrices

$$[S(\nu_k) \;\; \dot{S}(\nu_k)] = \Phi_k \, \Sigma_k \, V_k^T$$

- Φ_k and V_k are respectively the spatial and temporal bases of dim m associated to the parameter value ν_k such that

$$(\Phi_k, \Phi_k)_{\mathcal{A}_x} = I \quad \text{and} \quad (V_k, V_k)_{\mathcal{A}_t} = I$$

- Σ_k diagonal matrix (square roots of POD eigenvelues)

Data pretreatment (2nd POD stage)

We perform a second stage POD to the matrices Φ_k, we obtain

$$[\Phi_1 \Sigma_1 \quad \Phi_2 \Sigma_2 \quad \dots \quad \Phi_p \Sigma_p] = \Psi \Theta [\varphi_1 \quad \varphi_2 \quad \dots \quad \varphi_p]$$

By writing

$$V_k^T = [v_k^1 \quad v_k^2 \quad \dots \quad v_k^N \quad \dot{v}_k^1 \quad \dot{v}_k^2 \quad \dots \quad \dot{v}_k^N]$$

$u(\nu_k)$ and $\dot{u}(\nu_k)$ can be written as

$$u^n(\nu_k) = \Psi \Theta \varphi_k v_k^n \qquad \text{and} \qquad \dot{u}^n(\nu_k) = \Psi \Theta \varphi_k \dot{v}_k^n$$

Data pretreatment (2nd POD stage)

We perform a second stage POD to the matrices Φ_k, we obtain

$$[\Phi_1 \Sigma_1 \quad \Phi_2 \Sigma_2 \quad \dots \quad \Phi_p \Sigma_p] = \Psi \Theta [\varphi_1 \quad \varphi_2 \quad \dots \quad \varphi_p]$$

By writing

$$V_k^T = [v_k^1 \quad v_k^2 \quad \dots \quad v_k^N \quad \dot{v}_k^1 \quad \dot{v}_k^2 \quad \dots \quad \dot{v}_k^N]$$

$u(\nu_k)$ and $\dot{u}(\nu_k)$ can be written as

$$u^n(\nu_k) = \Psi \Theta \varphi_k v_k^n \qquad \text{and} \qquad \dot{u}^n(\nu_k) = \Psi \Theta \varphi_k \dot{v}_k^n$$

Since u and $\dot{u}$ satisfy the Navier-Stokes equation, the data-driven reduced order model for ν_k writes

$$\begin{cases} \dot{v}_{k_i} + \nu_k \varphi_{k_{si}} \varphi_{k_{rj}} L_{rs} v_{k_i} + \varphi_{k_{je}} \varphi_{k_{si}} \varphi_{k_{rj}} C_{ers} v_{k_e} v_{k_j} = 0, \\ v_{k_i}(0) = v_{k_i}^0, \end{cases}$$

$i, j, e \in \{1, 2, \dots, q\}$ the 1st stage POD modes indices, $r, s \in \{1, 2, \dots, pq\}$ the 2nd stage POD modes indices, and $k \in \{1, 2, \dots, p\}$ the parameter index.

Optimization problem

For $t = t_n$, The residual of the NSE reduced order model for ν_k writes

$$\mathcal{R}_k^n(L,C) = \sum_{i=1}^{q} \left(\dot{v}_i^n + \nu_k \, \varphi_{k_{si}} \, \varphi_{k_{rj}} \, L_{rs} \, v_j^n + \varphi_{k_{je}} \, \varphi_{k_{si}} \, \varphi_{k_{rj}} C_{ers} \, v_e^n \, v_j^n \right)^2$$

Let $\mathcal{J}$ be the residual functional

$$\mathcal{J}(L,C) = \frac{1}{2} \sum_{n=1}^{N} \sum_{k=1}^{p} \mathcal{R}_k^n(L,C) + \gamma \, L_{rs}^2 + \gamma \, C_{ers}^2$$

where γ is the Tikhonov regularization. The Data-Driven model reduction approach suggests to seek L and C such that

$$\{L, C\} = \underset{\delta L, \delta C}{\textbf{argmin}} \, \mathcal{J}(\delta L, \delta C)$$

implementation

To determine L and C, we use an optimal gradient descent approach with alternating directions

$$\begin{cases} L \text{ known} \\ \text{Solve } \min\limits_{C} \mathcal{J}(L, C) \longrightarrow C \\ \text{Solve } \min\limits_{L} \mathcal{J}(L, C) \longrightarrow L \end{cases}$$

For a new parameter value $\tilde{\nu}$, one needs to solve

$$\begin{cases} \dot{\tilde{v}}_{k_i} + \nu_k \, \tilde{\varphi}_{k_{si}} \, \tilde{\varphi}_{k_{rj}} \, L_{rs} \, \tilde{v}_{k_i} + \tilde{\varphi}_{k_{je}} \, \tilde{\varphi}_{k_{si}} \, \tilde{\varphi}_{k_{rj}} C_{ers} \, \tilde{v}_{k_e} \, \tilde{v}_{k_j} = 0, \\ \tilde{v}_{k_i}(0) = v^0_{k_i}, \end{cases}$$

The procedure is

- ▷ interpolate the training POD subspaces and obtain $\tilde{\varphi}$
- ▷ solve the resulting system and obtain $\tilde{v}$
- ▷ reconstruct the solution $\tilde{u}(t) = \Psi \, \Theta \, \tilde{\varphi} \, \tilde{v}(t)$

Numerical test : flow past a cylinder

Consider the two sets of parametrized snapshots matrices

$$\begin{bmatrix} u^1(\nu) & u^2(\nu) & \dots & u^N(\nu) \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \dot{u}^1(\nu) & \dot{u}^2(\nu) & \dots & \dot{u}^N(\nu) \end{bmatrix}$$

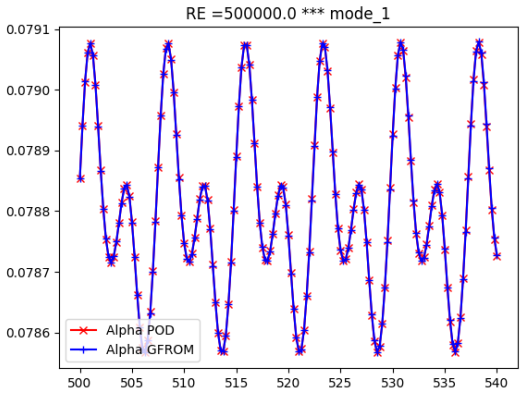
where $\nu = \boldsymbol{U}_\infty D/Re$ the parameter of interest such that

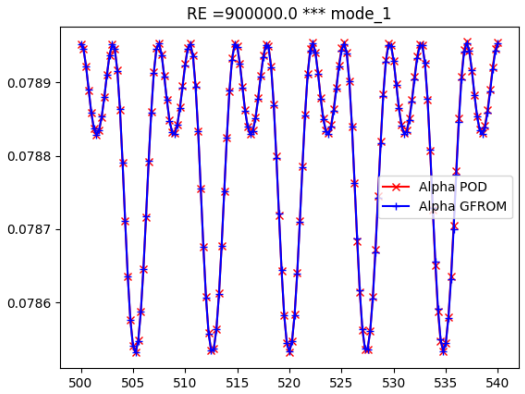
- Training points : $Re_{\text{train}} \in \{500000, 900000, 1300000\}$
- Test points : $Re_{\text{test}} \in \{600000, 700000, 800000, 1100000, 1200000\}$

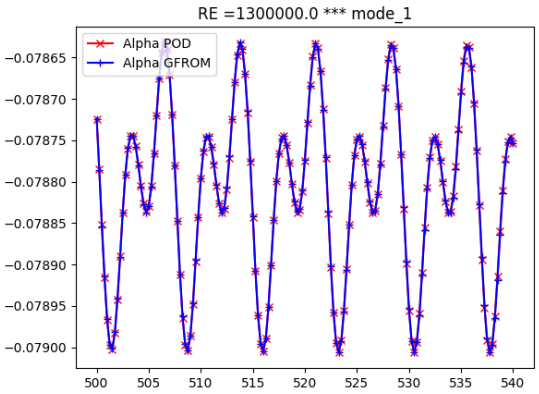
The steps followed for the Data-Driven approach are

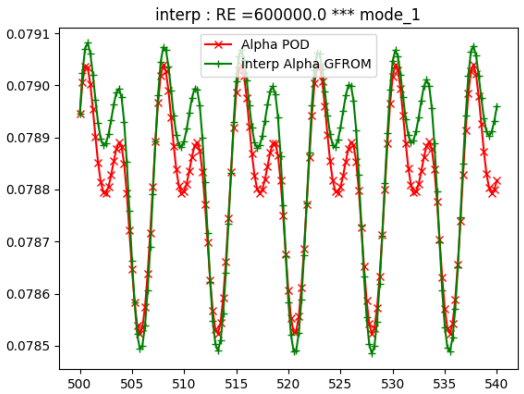
- High fidelity simulations (OpenFoam) : Turbulence model pimpleFoam SpalartAllmaras with $dt = 0.005$
- POD truncation order $m = 9$
- Tikhonov constant $\gamma = 1e-8$

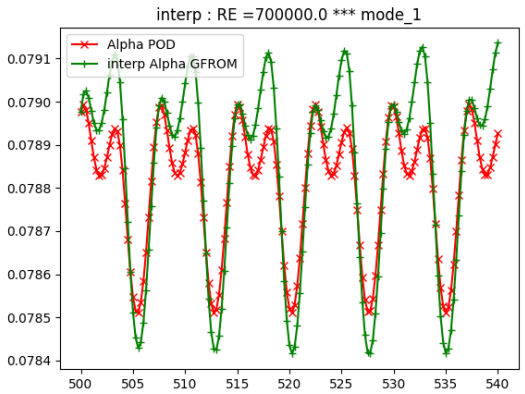


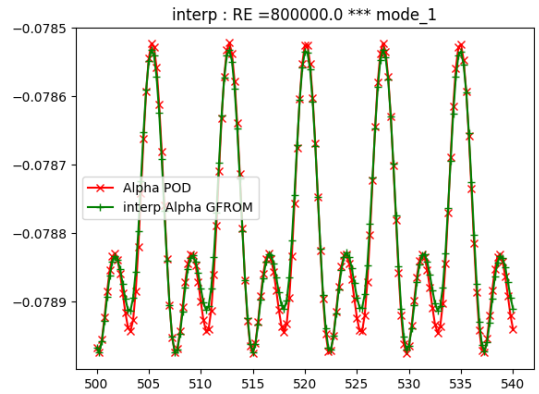

RE =500000.0 *** mode_1
Alpha POD
Alpha GFROM

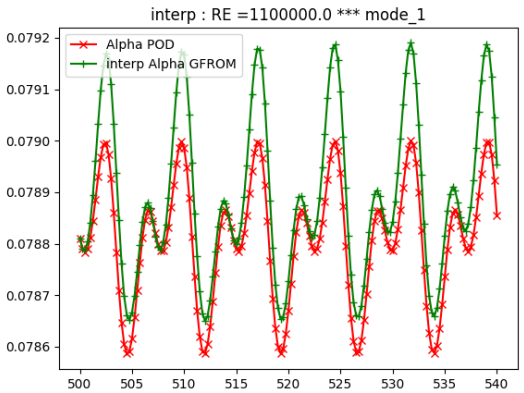

RE =900000.0 *** mode_1
Alpha POD
Alpha GFROM

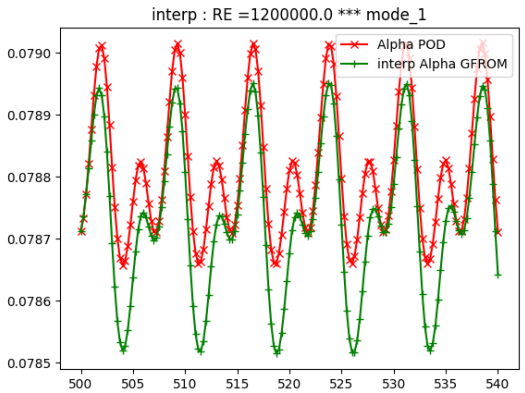

RE =1300000.0 *** mode_1
Alpha POD
Alpha GFROM

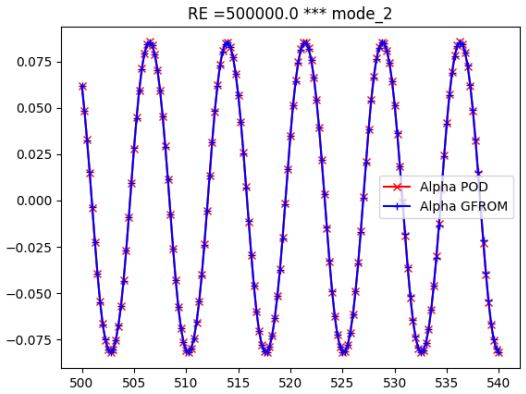

interp : RE =600000.0 *** mode_1
Alpha POD
interp Alpha GFROM

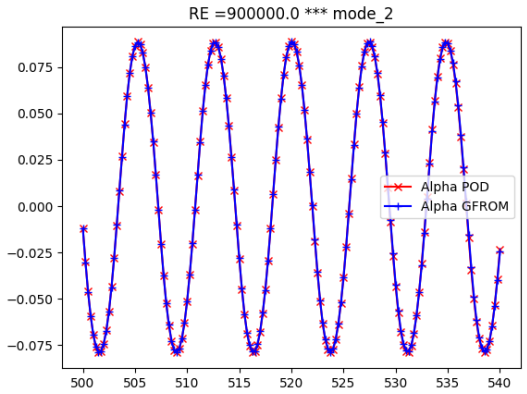

interp : RE =700000.0 *** mode_1
Alpha POD
interp Alpha GFROM

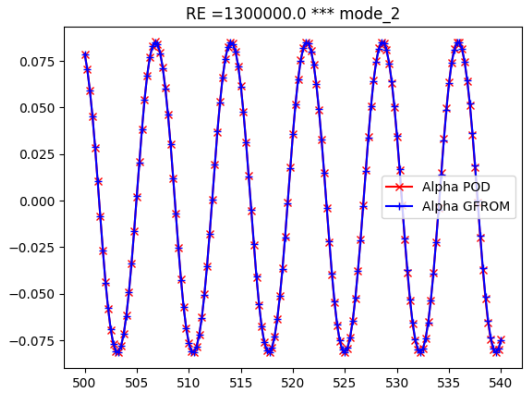

interp : RE =800000.0 *** mode_1
Alpha POD
interp Alpha GFROM

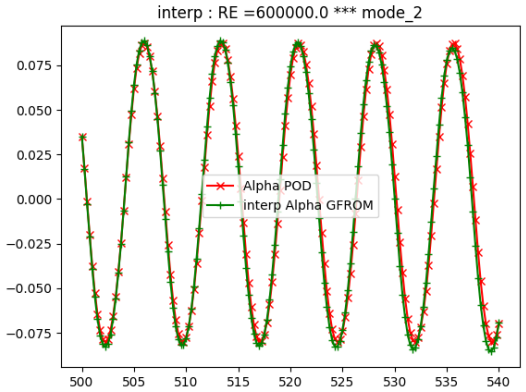

interp : RE =1100000.0 *** mode_1
Alpha POD
interp Alpha GFROM

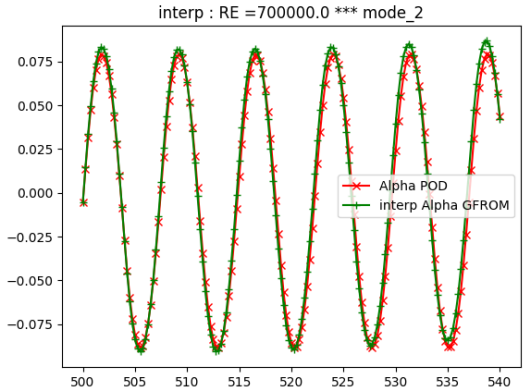

interp : RE =1200000.0 *** mode_1
Alpha POD
interp Alpha GFROM

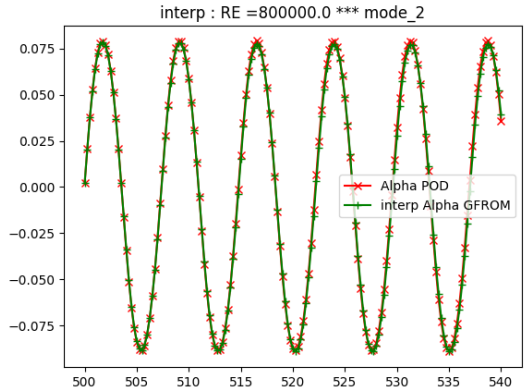

RE =500000.0 *** mode_2
Alpha POD
Alpha GFROM

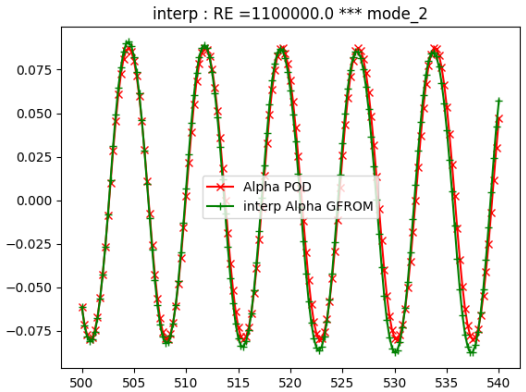

RE =900000.0 *** mode_2
Alpha POD
Alpha GFROM

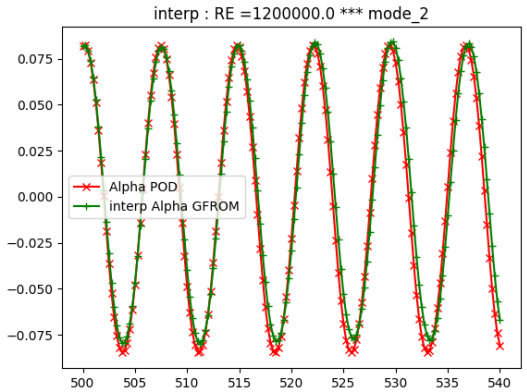

RE =1300000.0 *** mode_2
Alpha POD
Alpha GFROM

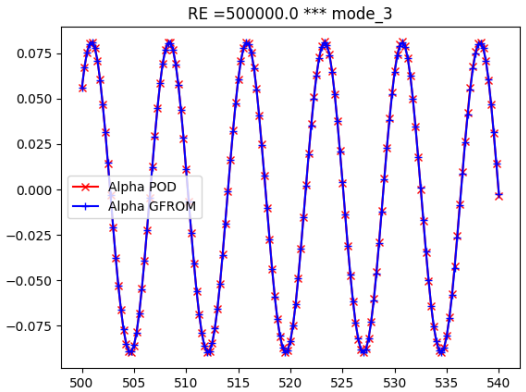

interp : RE =600000.0 *** mode_2
Alpha POD
interp Alpha GFROM

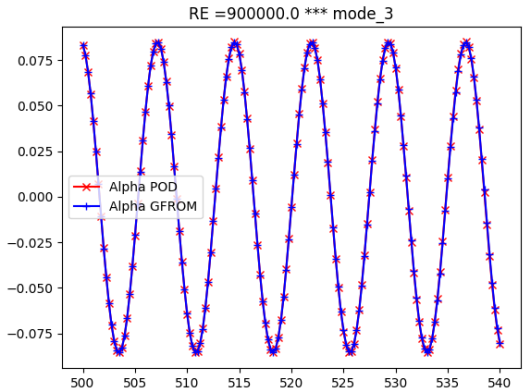

interp : RE =700000.0 *** mode_2
Alpha POD
interp Alpha GFROM

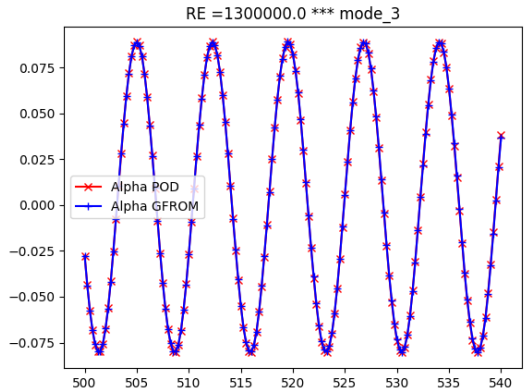

interp : RE =800000.0 *** mode_2
Alpha POD
interp Alpha GFROM

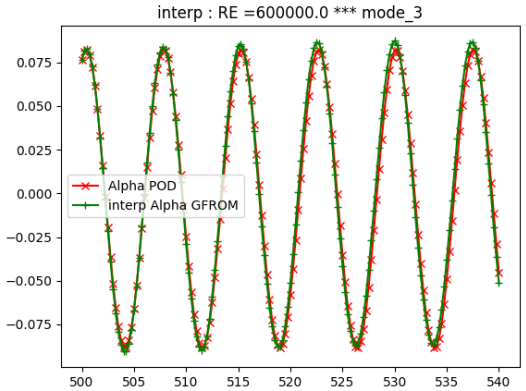

interp : RE =1100000.0 *** mode_2
Alpha POD
interp Alpha GFROM

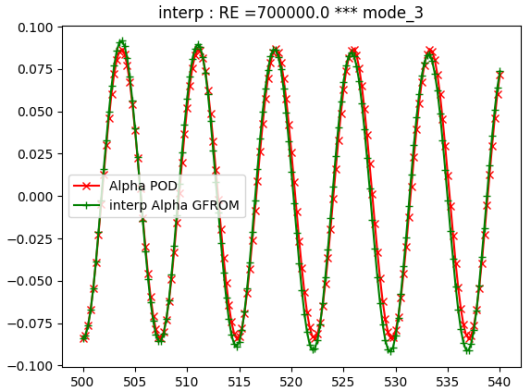

interp : RE =1200000.0 *** mode_2
Alpha POD
interp Alpha GFROM

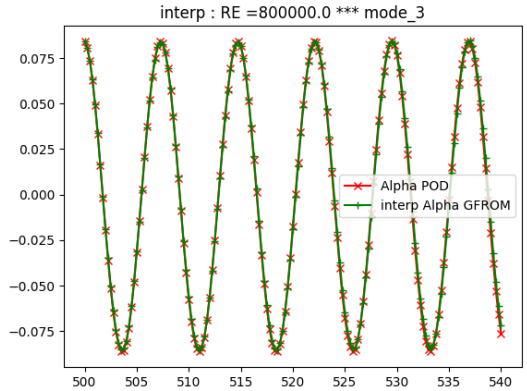

RE =500000.0 *** mode_3
Alpha POD
Alpha GFROM

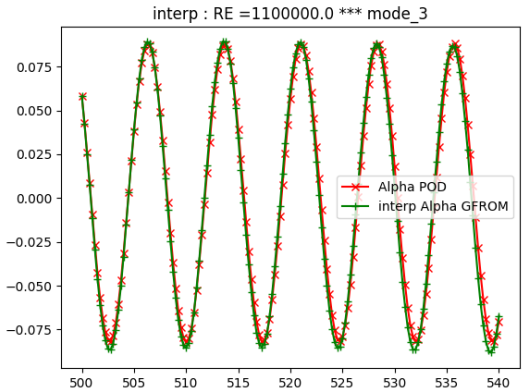

RE =900000.0 *** mode_3
Alpha POD
Alpha GFROM

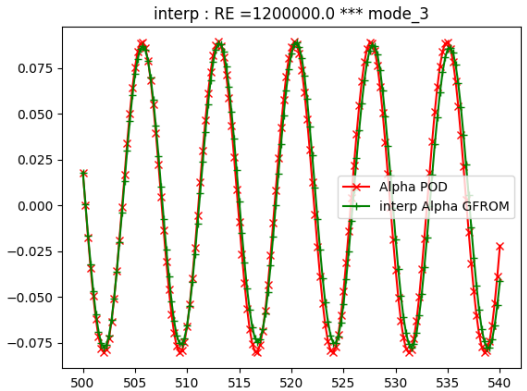

RE =1300000.0 *** mode_3
Alpha POD
Alpha GFROM

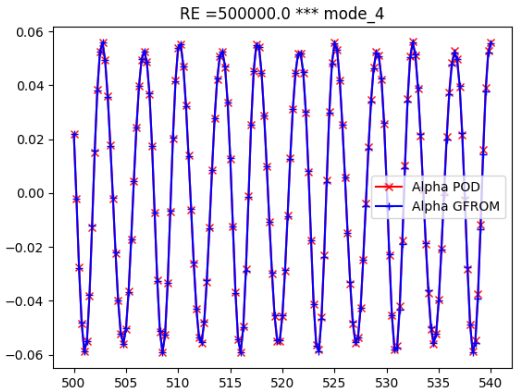

interp : RE =600000.0 *** mode_3
Alpha POD
interp Alpha GFROM

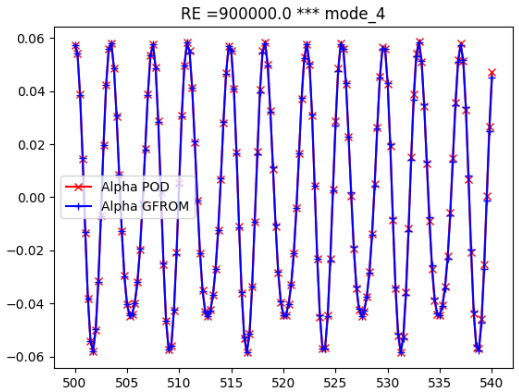

interp : RE =700000.0 *** mode_3
Alpha POD
interp Alpha GFROM

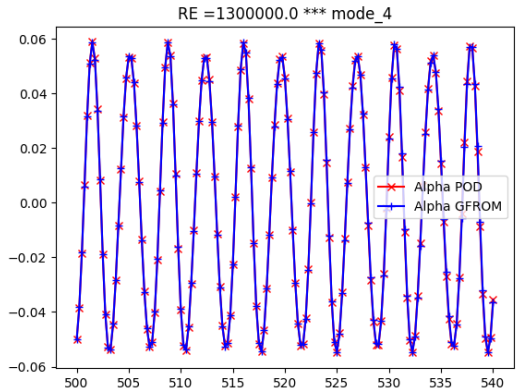

interp : RE =800000.0 *** mode_3
Alpha POD
interp Alpha GFROM

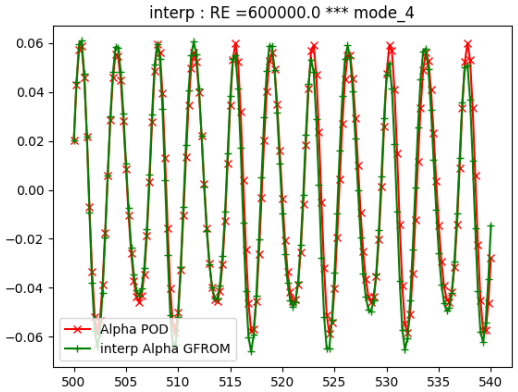

interp : RE =1100000.0 *** mode_3
Alpha POD
interp Alpha GFROM

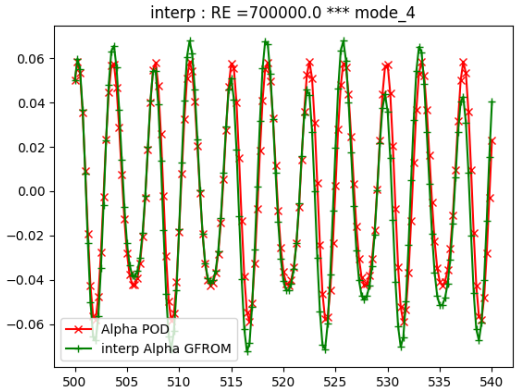

interp : RE =1200000.0 *** mode_3
Alpha POD
interp Alpha GFROM

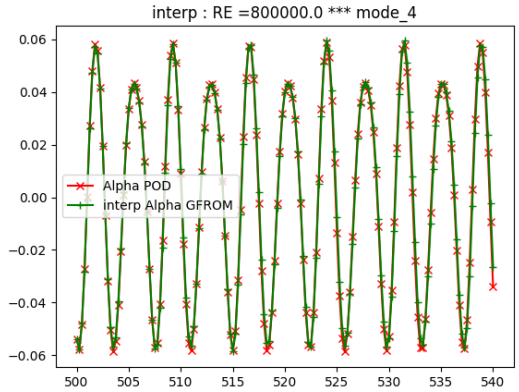

RE =500000.0 *** mode_4
Alpha POD
Alpha GFROM

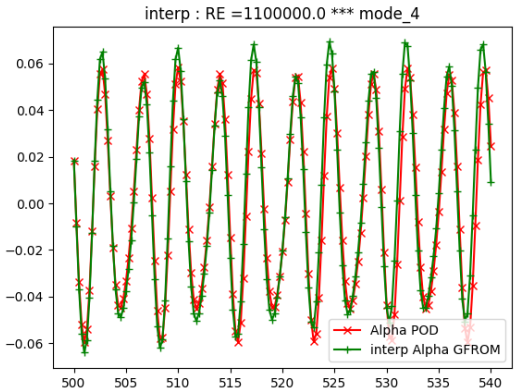

RE =900000.0 *** mode_4
Alpha POD
Alpha GFROM

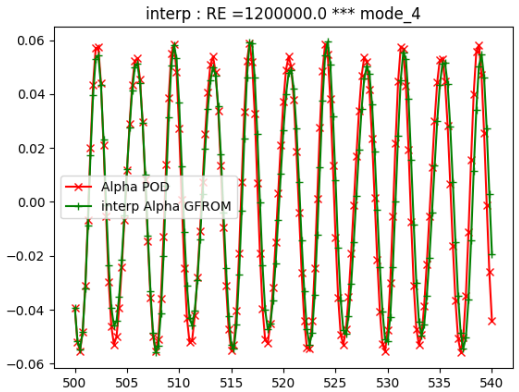

RE =1300000.0 *** mode_4
Alpha POD
Alpha GFROM

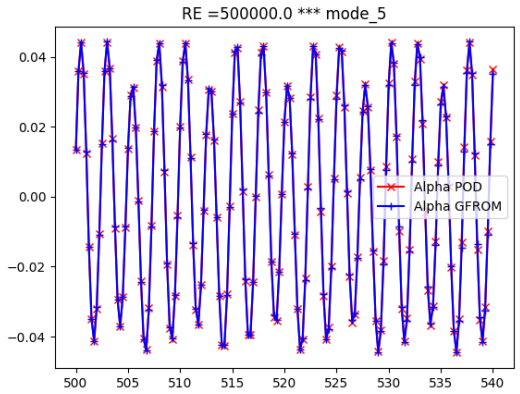

interp : RE =600000.0 *** mode_4
Alpha POD
interp Alpha GFROM

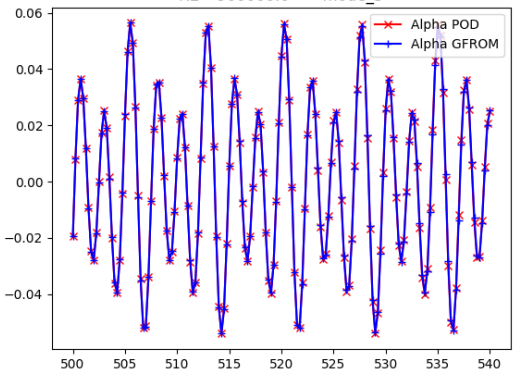

interp : RE =700000.0 *** mode_4
Alpha POD
interp Alpha GFROM

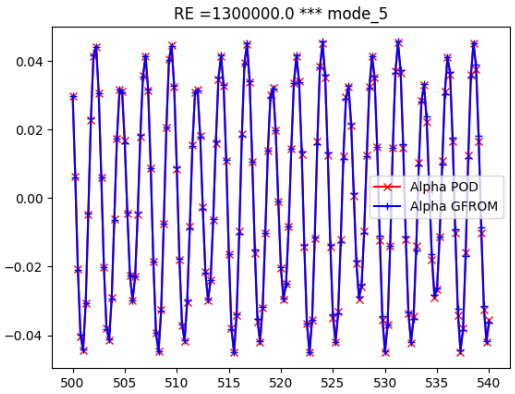

interp : RE =800000.0 *** mode_4
Alpha POD
interp Alpha GFROM

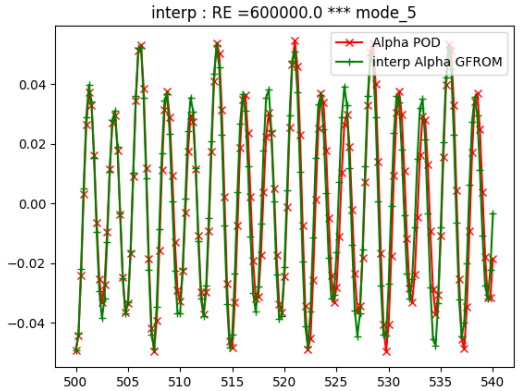

interp : RE =1100000.0 *** mode_4
Alpha POD
interp Alpha GFROM

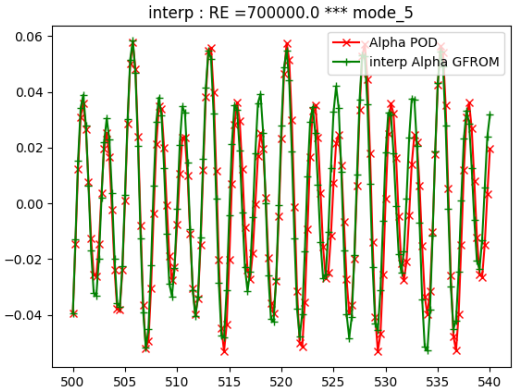

interp : RE =1200000.0 *** mode_4
Alpha POD
interp Alpha GFROM

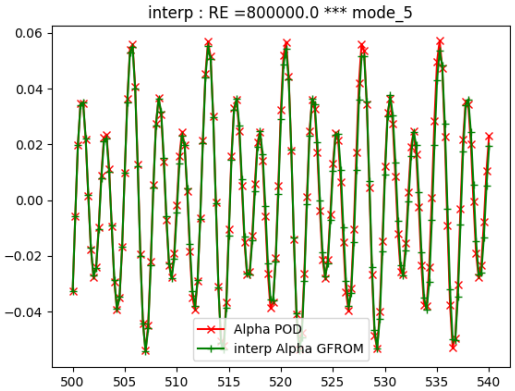

RE =500000.0 *** mode_5
Alpha POD
Alpha GFROM

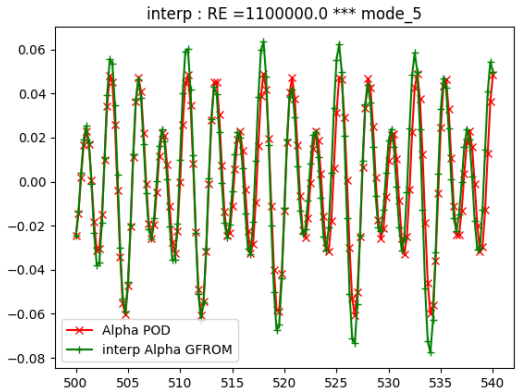

RE =900000.0 *** mode_5
Alpha POD
Alpha GFROM

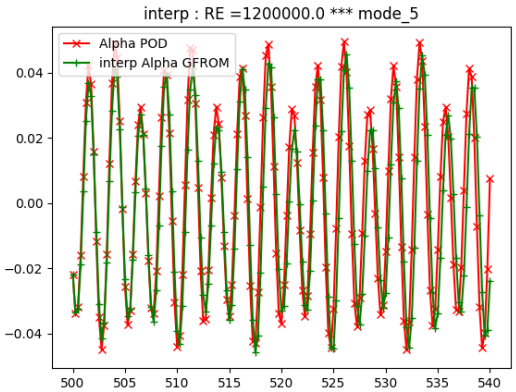

RE =1300000.0 *** mode_5
Alpha POD
Alpha GFROM

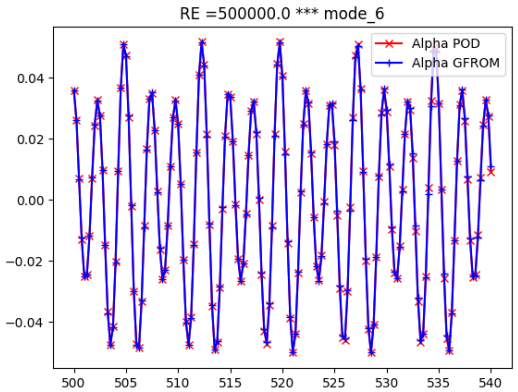

interp : RE =600000.0 *** mode_5
Alpha POD
interp Alpha GFROM

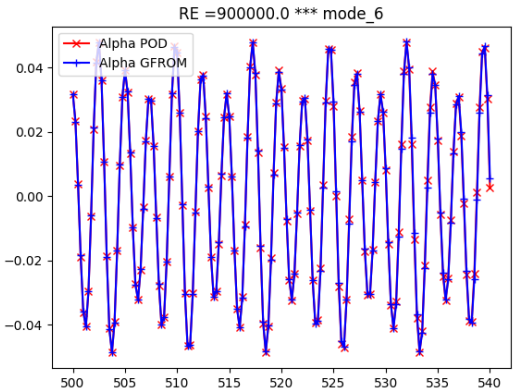

interp : RE =700000.0 *** mode_5
Alpha POD
interp Alpha GFROM

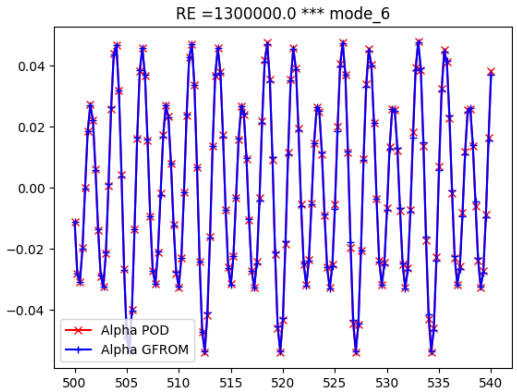

interp : RE =800000.0 *** mode_5
Alpha POD
interp Alpha GFROM

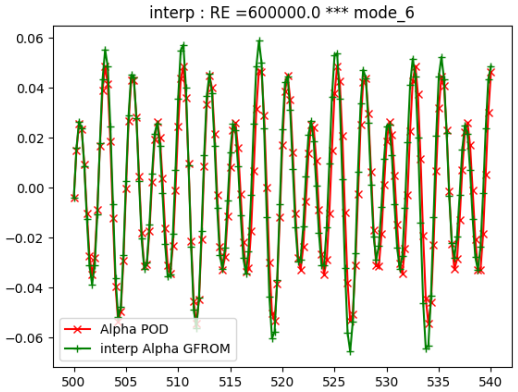

interp : RE =1100000.0 *** mode_5
Alpha POD
interp Alpha GFROM

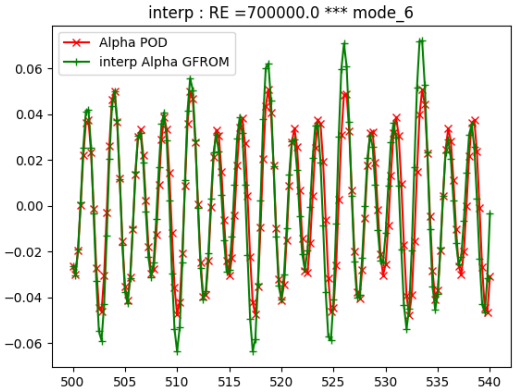

interp : RE =1200000.0 *** mode_5
Alpha POD
interp Alpha GFROM

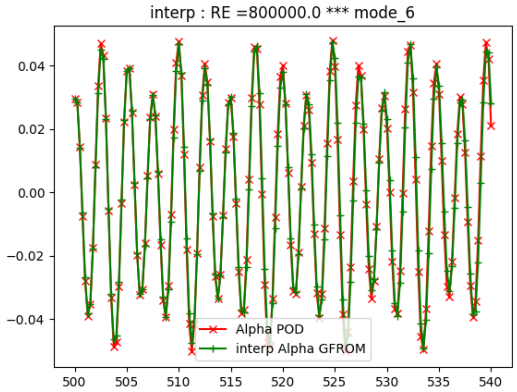

RE =500000.0 *** mode_6
Alpha POD
Alpha GFROM

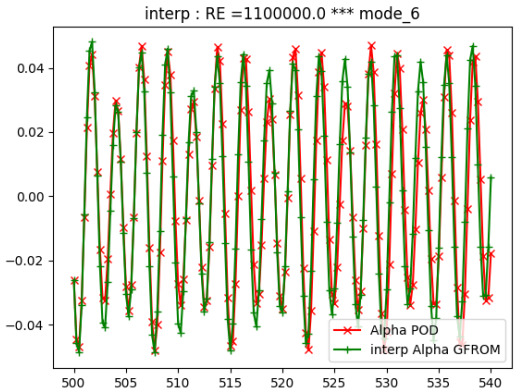

RE =900000.0 *** mode_6
Alpha POD
Alpha GFROM

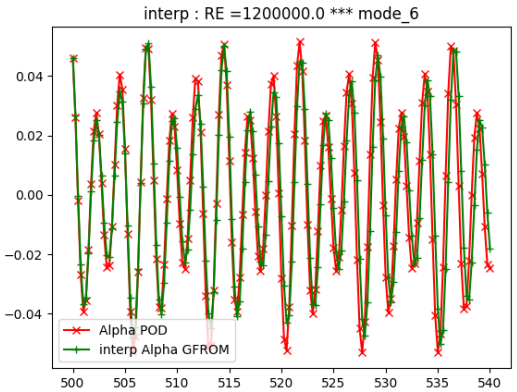

RE =1300000.0 *** mode_6
Alpha POD
Alpha GFROM

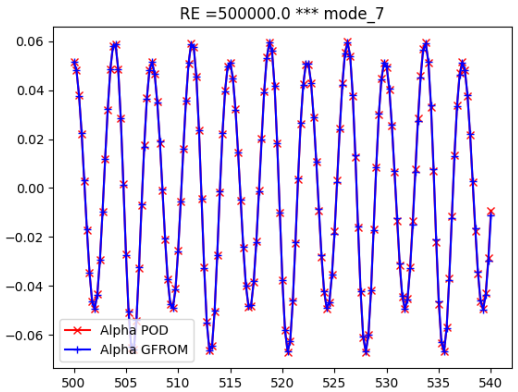

interp : RE =600000.0 *** mode_6
Alpha POD
interp Alpha GFROM

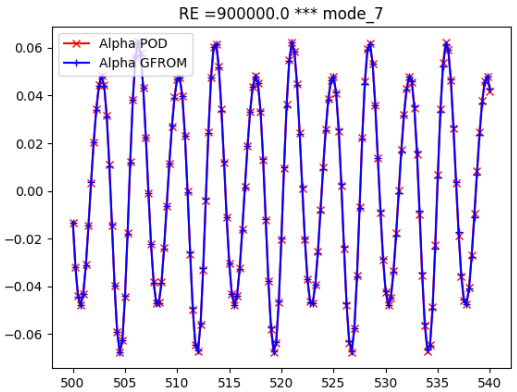

interp : RE =700000.0 *** mode_6
Alpha POD
interp Alpha GFROM

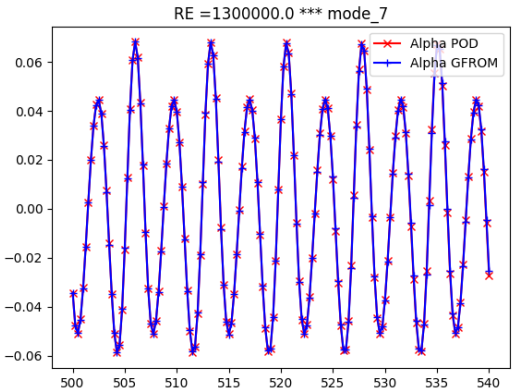

interp : RE =800000.0 *** mode_6
Alpha POD
interp Alpha GFROM

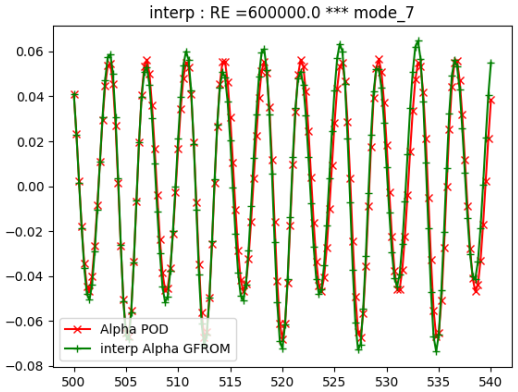

interp : RE =1100000.0 *** mode_6
Alpha POD
interp Alpha GFROM

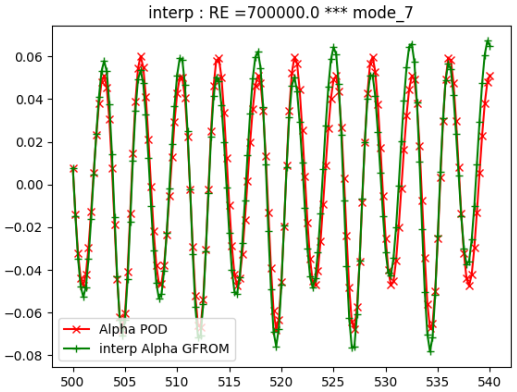

interp : RE =1200000.0 *** mode_6
Alpha POD
interp Alpha GFROM

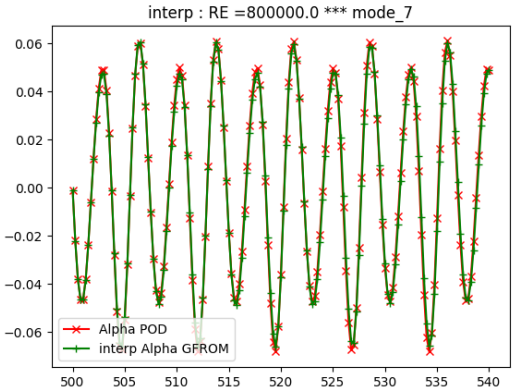

RE =500000.0 *** mode_7
Alpha POD
Alpha GFROM

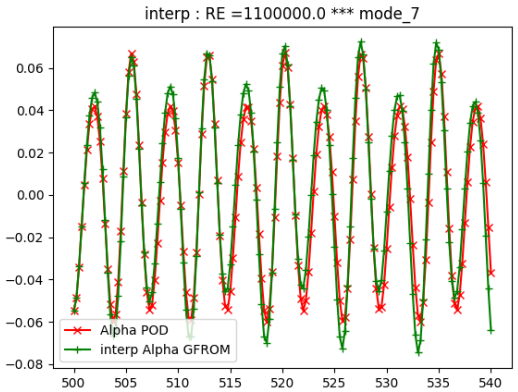

RE =900000.0 *** mode_7
Alpha POD
Alpha GFROM

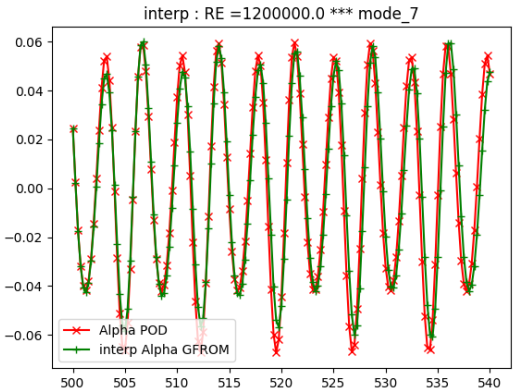

RE =1300000.0 *** mode_7
Alpha POD
Alpha GFROM

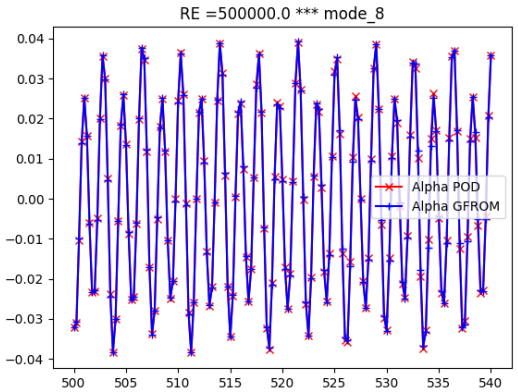

interp : RE =600000.0 *** mode_7
Alpha POD
interp Alpha GFROM

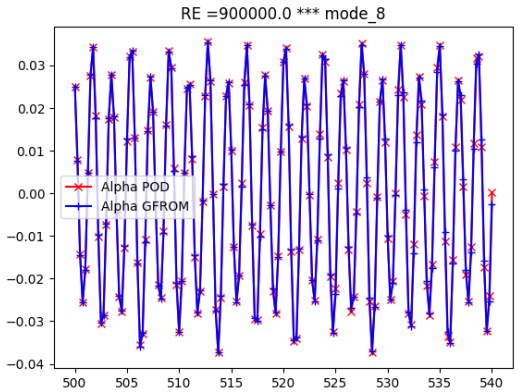

interp : RE =700000.0 *** mode_7
Alpha POD
interp Alpha GFROM

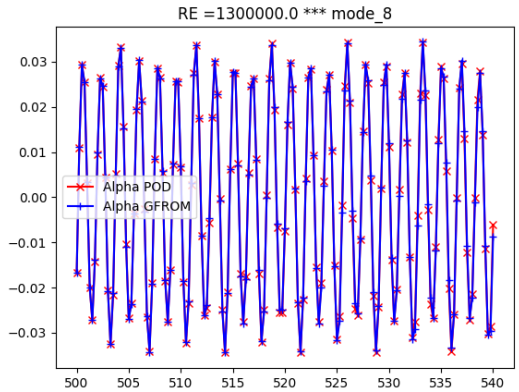

interp : RE =800000.0 *** mode_7
Alpha POD
interp Alpha GFROM

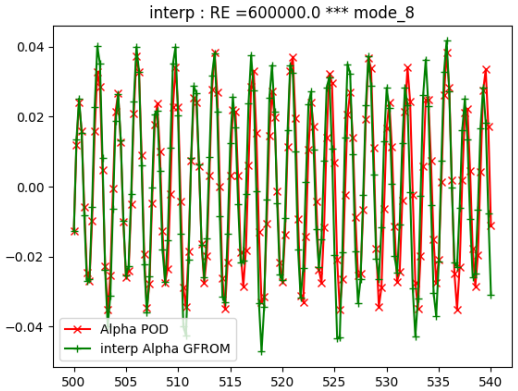

interp : RE =1100000.0 *** mode_7
Alpha POD
interp Alpha GFROM

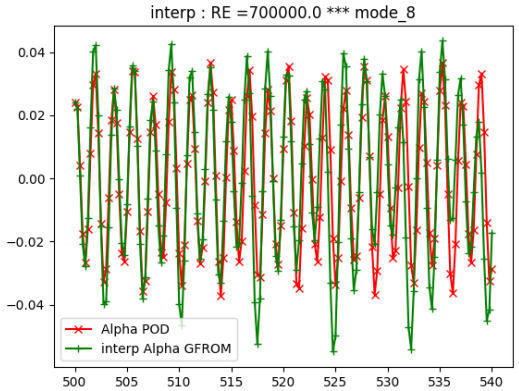

interp : RE =1200000.0 *** mode_7
Alpha POD
interp Alpha GFROM

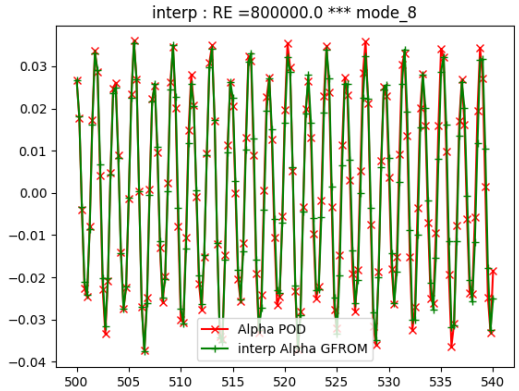

RE =500000.0 *** mode_8
Alpha POD
Alpha GFROM

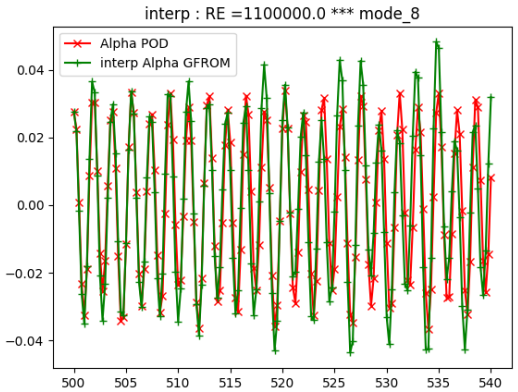

RE =900000.0 *** mode_8
Alpha POD
Alpha GFROM

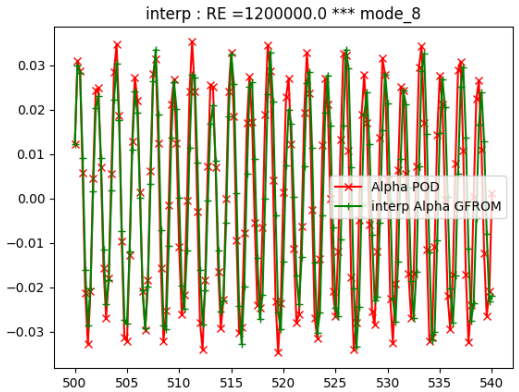

RE =1300000.0 *** mode_8
Alpha POD
Alpha GFROM

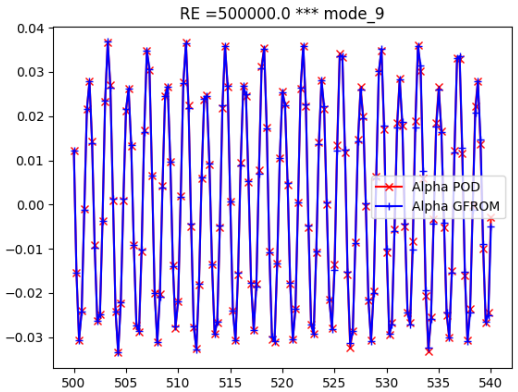

interp : RE =600000.0 *** mode_8
Alpha POD
interp Alpha GFROM

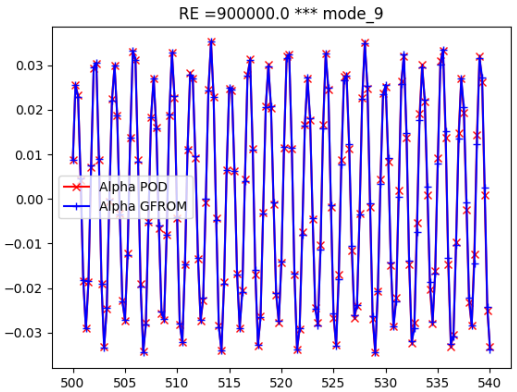

interp : RE =700000.0 *** mode_8
Alpha POD
interp Alpha GFROM

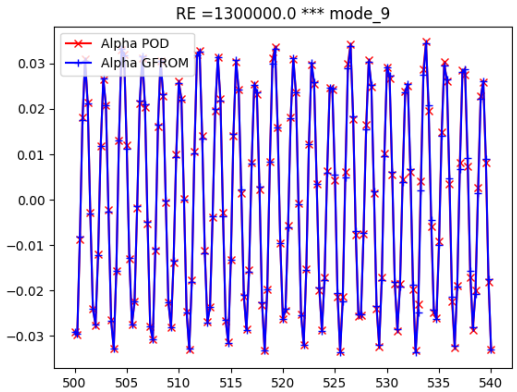

interp : RE =800000.0 *** mode_8
Alpha POD
interp Alpha GFROM

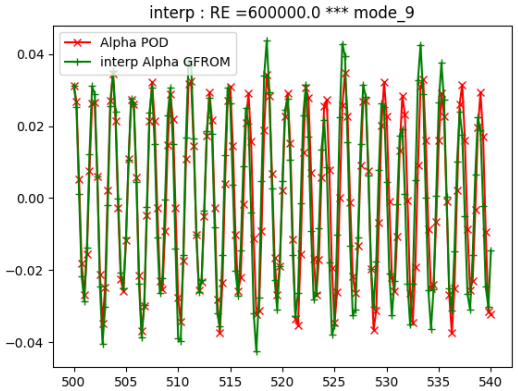

interp : RE =1100000.0 *** mode_8
Alpha POD
interp Alpha GFROM

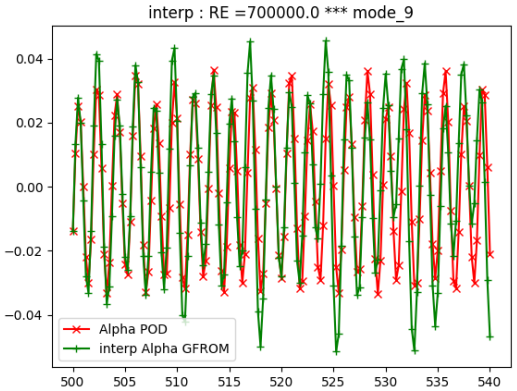

interp : RE =1200000.0 *** mode_8
Alpha POD
interp Alpha GFROM

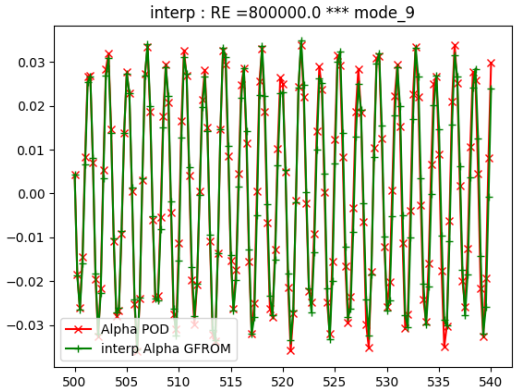

RE =500000.0 *** mode_9
Alpha POD
Alpha GFROM

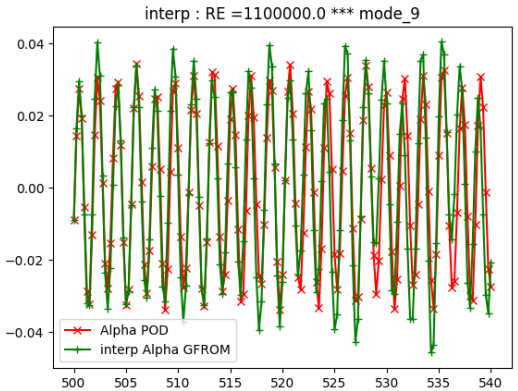

RE =900000.0 *** mode_9
Alpha POD
Alpha GFROM

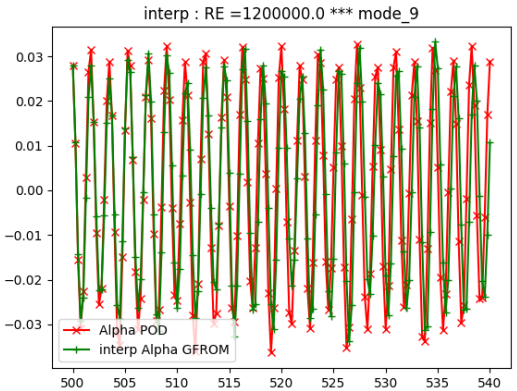

RE =1300000.0 *** mode_9
Alpha POD
Alpha GFROM


interp : RE =600000.0 *** mode_9
Alpha POD
interp Alpha GFROM


interp : RE =700000.0 *** mode_9
Alpha POD
interp Alpha GFROM


interp : RE =800000.0 *** mode_9
Alpha POD
interp Alpha GFROM


interp : RE =1100000.0 *** mode_9
Alpha POD
interp Alpha GFROM


interp : RE =1200000.0 *** mode_9
Alpha POD
interp Alpha GFROM

Stability !

Question : Knowing L and C, what can we say about the stability of the predicted reduced system ?

$$(*)\begin{cases} \dot{\tilde{v}}_{k_i} + \nu_k\, \tilde{\varphi}_{k_{si}}\, \tilde{\varphi}_{k_{rj}}\, L_{rs}\, \tilde{v}_{k_i} + \tilde{\varphi}_{k_{je}}\, \tilde{\varphi}_{k_{si}}\, \tilde{\varphi}_{k_{rj}} C_{ers}\, \tilde{v}_{k_e}\, \tilde{v}_{k_j} = 0 \\ \tilde{v}_{k_i}(0) = v^0_{k_i}, \end{cases} ,$$

Stability !

Question : Knowing L and C, what can we say about the stability of the predicted reduced system ?

$$(*)\begin{cases}\dot{\tilde{v}}_{k_i} + \nu_k\,\tilde{\varphi}_{k_{si}}\,\tilde{\varphi}_{k_{rj}}\,L_{rs}\,\tilde{v}_{k_i} + \tilde{\varphi}_{k_{je}}\,\tilde{\varphi}_{k_{si}}\,\tilde{\varphi}_{k_{rj}}C_{ers}\,\tilde{v}_{k_e}\,\tilde{v}_{k_j} & +\tilde{\varphi}_{k_{si}}\,G_s = 0,\\ \tilde{v}_{k_i}(0) = v^0_{k_i},\end{cases}$$

- in the training points

$$\min_{G}\sum_{k=1}^{p}\sum_{i=1}^{m}||v_i(\nu_k) - v_i^{\text{POD}}(\nu_k)||^2_{L^2(0,T)} \quad \text{s.t } (*)$$

- in the test points !

$$\min_{G}\sum_{k\in\mathcal{K}=i,j=1}^{m}\left(||\tilde{v}(\nu_k)||^2_{L^2(0,T)} + ||\dot{\tilde{v}}(\nu_k)||^2_{L^2(0,T)} - \delta_{ij}\right)^2 \quad \text{s.t } (*)$$

Thank you for your attention