Interpolation of parameterized reduced-order models

David Amsallem

Stanford University

February 06, 2014

Outline

- Parameterized unsteady PDE
- Database approach
- Interpolation of reduced-order bases
- Interpolation of reduced-order matrices
- Applications

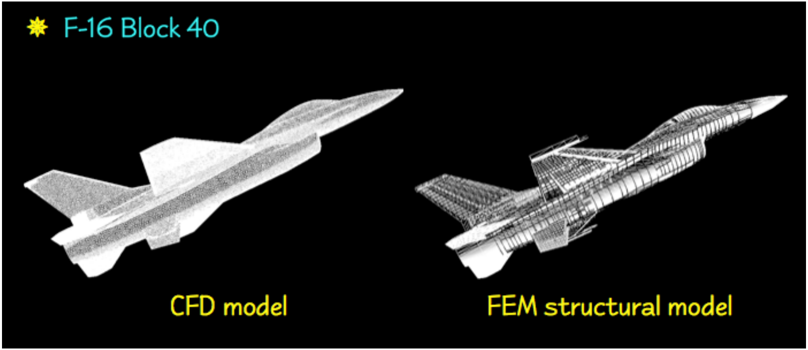
Parametrized unsteady PDE

- Linear (linearized) PDE

$$\frac{d\mathbf{w}}{dt}(t) = \mathbf{A}(\boldsymbol{\mu})\mathbf{w}(t) + \mathbf{B}(\boldsymbol{\mu})\mathbf{u}(t)$$

Example: aeroelasticity

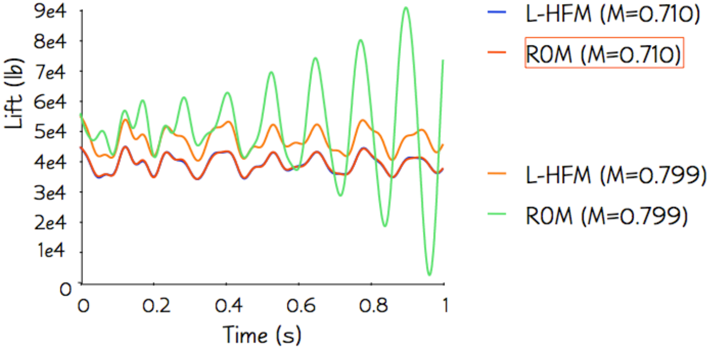
- Example: aeroelastic analysis of full aircraft configuration



- Hundreds of flight conditions $\boldsymbol{\mu} = (M_\infty, \alpha)$ to clear for flutter

Example: aeroelasticity

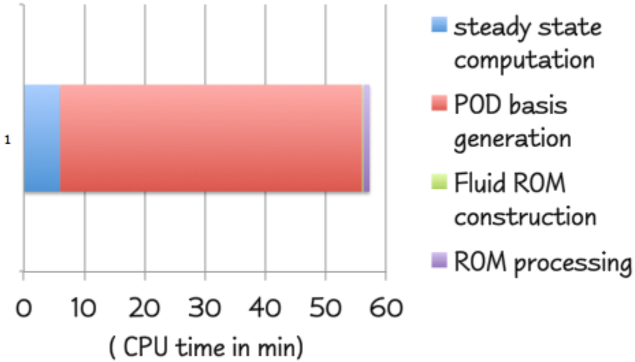
- Example



- Non-robustness with respect to the operating condition

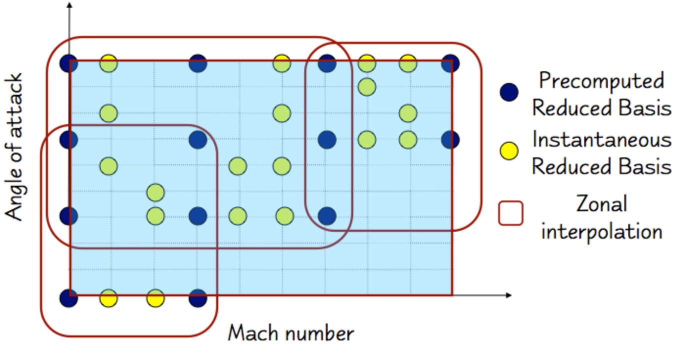
Example: aeroelasticity

- Can we afford rebuilding the POD basis?



- Not suitable for real-time analysis

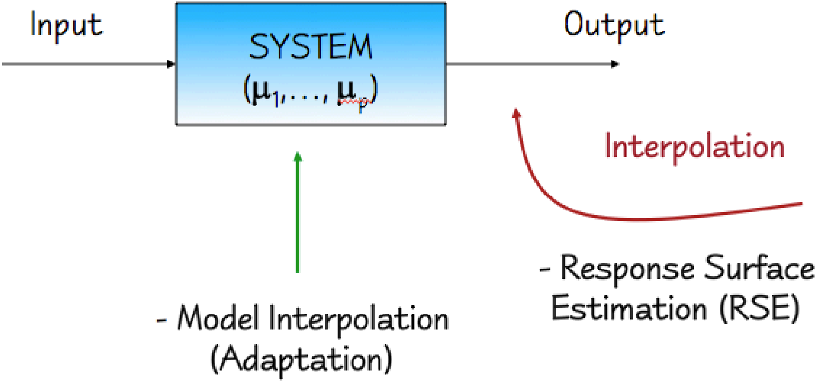
Database approach



- What should we interpolate?

Database approach

- What should we interpolate?



Model interpolation

- Reduced set of equations (after Galerkin projection)

$$\frac{d\mathbf{q}}{dt}(t) = \mathbf{V}(\boldsymbol{\mu})^T\mathbf{A}(\boldsymbol{\mu})\mathbf{V}(\boldsymbol{\mu})\mathbf{q}(t) + \mathbf{V}(\boldsymbol{\mu})^T\mathbf{B}\mathbf{u}(t)$$

- Full state reconstruction

$$\mathbf{w}(t) \approx \mathbf{V}(\boldsymbol{\mu})\mathbf{q}(t)$$

- Model: $\left(\mathbf{V}(\boldsymbol{\mu})^T\mathbf{A}(\boldsymbol{\mu})\mathbf{V}(\boldsymbol{\mu}), \mathbf{V}(\boldsymbol{\mu})^T\mathbf{B}, \mathbf{V}(\boldsymbol{\mu})\right)$
- Approach #1:
  1. interpolate $\mathbf{V}(\boldsymbol{\mu})$
  2. evaluate $(\mathbf{A}(\boldsymbol{\mu}), \mathbf{B}(\boldsymbol{\mu}))$
  3. form $\left(\mathbf{V}(\boldsymbol{\mu})^T\mathbf{A}(\boldsymbol{\mu})\mathbf{V}(\boldsymbol{\mu}), \mathbf{V}(\boldsymbol{\mu})^T\mathbf{B}\right)$

Direct interpolation

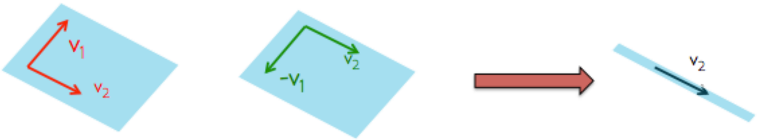
- Natural idea: interpolate $\mathbf{V}(\boldsymbol{\mu}) \in \mathbb{R}^{N_{\mathbf{w}} \times k}$ entry-by-entry
- Input:
  - target $\boldsymbol{\mu}$
  - precomputed reduced bases $\{\mathbf{V}(\boldsymbol{\mu}_l)\}_{l=1}^{m}$
  - multi-variate interpolation operator $a(\boldsymbol{\mu}) = \mathcal{I}(\boldsymbol{\mu}; \{a(\boldsymbol{\mu}_l), \boldsymbol{\mu}_l\}_{l=1}^{m})$
- Algorithm

1: **for** $i = 1 : N_{\mathbf{w}}$ **do**
2:     **for** $j = 1 : k$ **do**
3:         Compute $v_{ij}(\boldsymbol{\mu}) = \mathcal{I}(\boldsymbol{\mu}; \{v_{ij}(\boldsymbol{\mu}_l), \boldsymbol{\mu}_l\}_{l=1}^{m})$
4:     **end for**
5: **end for**
6: Form $\mathbf{V}(\boldsymbol{\mu}) = [v_{ij}(\boldsymbol{\mu})]$

Direct interpolation doesn't work

- Example
  - $N_{\mathbf{w}} = 3$, $k = 2$, $p = 1$
  - for $\mu_1 = 0$, $\mathbf{V}(0) = [\mathbf{v}_1, \mathbf{v}_2]$
  - for $\mu_2 = 1$, $\mathbf{V}(1) = [-\mathbf{v}_1, \mathbf{v}_2]$
  - target parameter $\mu = 0.5$
  - use linear interpolation
- Interpolation result:

$$\mathbf{V}(0.5) = 0.5(\mathbf{V}(0) + \mathbf{V}(1)) = [0.5(\mathbf{v}_1 - \mathbf{v}_1), 0.5(\mathbf{v}_2 + \mathbf{v}_2)] = [0, \mathbf{v}_2]$$



- What went wrong?
- We haven't interpolated the correct object

Subspace interpolation

- Projection-based model reduction

$$\frac{d\mathbf{q}}{dt}(t) = \mathbf{V}(\boldsymbol{\mu})^T\mathbf{A}(\boldsymbol{\mu})\mathbf{V}(\boldsymbol{\mu})\mathbf{q}(t) + \mathbf{V}(\boldsymbol{\mu})^T\mathbf{B}\mathbf{u}(t)$$

- Equivalent full state equation (multiply by $\mathbf{V}(\boldsymbol{\mu})$)

$$\frac{d\mathbf{w}}{dt}(t) = \boldsymbol{\Pi}_{\mathbf{V}(\boldsymbol{\mu}),\mathbf{V}(\boldsymbol{\mu})}\mathbf{A}(\boldsymbol{\mu})\mathbf{w}(t) + \boldsymbol{\Pi}_{\mathbf{V}(\boldsymbol{\mu}),\mathbf{V}(\boldsymbol{\mu})}\mathbf{B}\mathbf{u}(t)$$

- An orthogonal projection is independent of the choice of reduced basis associated to the projection subspace
- Important quantity to interpolate: subspace

The Grassmann manifold

- A subspace $\mathcal{S}$ is typically represented by a reduced basis
- The choice of reduced basis is not unique

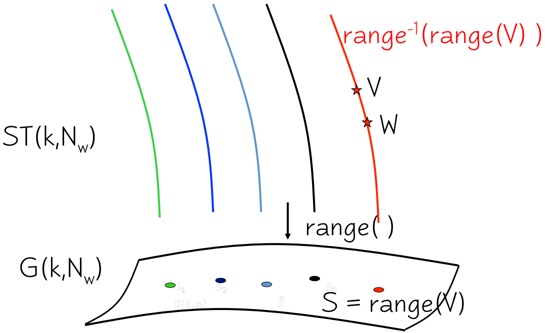
$$\mathcal{S} = \text{range}(\mathbf{V}) = \text{range}(\mathbf{VQ}),\ \forall \mathbf{Q} \in \text{GL}(k)$$

- Matrix manifolds of interest
  - $\mathcal{G}(k, N_{\mathbf{w}})$ (Grassmann manifold): set of subspaces of dimension k in $\mathbb{R}^{N_{\mathbf{w}}}$
  - $\mathcal{ST}(k, N_{\mathbf{w}})$ (orthogonal Stiefel manifold): set of orthogonal reduced bases of dimension k in $\mathbb{R}^{N_{\mathbf{w}}}$
- Case of model reduction
  - $\mathbf{V}(\boldsymbol{\mu}) \in \mathcal{ST}(k, N_{\mathbf{w}})$
  - range$(\mathbf{V}(\boldsymbol{\mu})) \in \mathcal{G}(k, N_{\mathbf{w}})$
- Interpolation on the Grasmann manifold (quotient manifold) using quantities belonging to the Stiefel manifold

$$\mathcal{G}(k, N_{\mathbf{w}}) = \mathcal{ST}(k, N_{\mathbf{w}})/\mathcal{O}(k)$$

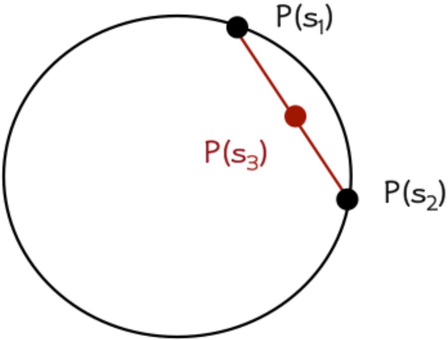
The Grassmann manifold

- Matrix manifolds of interest
  - $\mathcal{G}(k, N_{\mathbf{w}})$ (Grassmann manifold): set of subspaces of dimension k in $\mathbb{R}^{N_{\mathbf{w}}}$
  - $\mathcal{ST}(k, N_{\mathbf{w}})$ (orthogonal Stiefel manifold): set of orthogonal reduced bases of dimension k in $\mathbb{R}^{N_{\mathbf{w}}}$



Interpolation on matrix manifolds

- First example: the circle



- Standard interpolation fails
- Idea: interpolate on a linear space $\Rightarrow$ a tangent space to the manifold

Interpolation on matrix manifolds

- Input:
  - precomputed matrices $\{\mathbf{A}(\boldsymbol{\mu}_l)\}_{l=1}^{m}$
  - multi-variate interpolation operator $a(\boldsymbol{\mu}) = \mathcal{I}(\boldsymbol{\mu}; \{a(\boldsymbol{\mu}_l), \boldsymbol{\mu}_l\}_{l=1}^{m})$
  - map $m_{\mathbf{A}}$ from the manifold $\mathcal{M}$ to the tangent space of $\mathcal{M}$ at $m_{\mathbf{A}}$
  - inverse map $m_{\mathbf{A}}^{-1}$ from the tangent space to $\mathcal{M}$ at $m_{\mathbf{A}}$ to the manifold $\mathcal{M}$

for $l = 1 : m$ **do**
    Compute $\boldsymbol{\Gamma}(\boldsymbol{\mu}_l) = m_{\mathbf{A}}(\mathbf{A}(\boldsymbol{\mu}_l))$
end for
for $i = 1 : N_{\mathbf{w}}$ **do**
    for $j = 1 : k$ **do**
        Compute $\Gamma_{ij}(\boldsymbol{\mu}_l) = \mathcal{I}(\boldsymbol{\mu}; \{\Gamma_{ij}(\boldsymbol{\mu}_l), \boldsymbol{\mu}_l\}_{l=1}^{m})$
    end for
end for
Form $\boldsymbol{\Gamma}(\boldsymbol{\mu}) = [\Gamma_{ij}(\boldsymbol{\mu})]$ and compute $\mathbf{A}(\boldsymbol{\mu}) = m_{\mathbf{A}}^{-1}(\boldsymbol{\Gamma}(\boldsymbol{\mu}))$

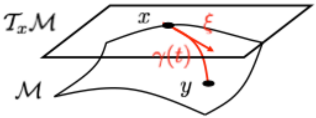
- Requirement: the interpolation operator $\mathcal{I}$ preserves the tangent space.

$$\text{for instance:} \quad a(\boldsymbol{\mu}) = \mathcal{I}(\boldsymbol{\mu}; \{a(\boldsymbol{\mu}_l), \boldsymbol{\mu}_l\}_{l=1}^{m}) = \sum_{l=1}^{m} \theta_l(\boldsymbol{\mu}) a(\boldsymbol{\mu}_l)$$

Interpolation on matrix manifolds

- How do we find $m_{\mathbf{A}}$ and its inverse $m_{\mathbf{A}}^{-1}$
- Idea: use concepts from differential geometry
- Geodesics
  - generalize straight lines on manifolds
  - uniquely defined given a point x of the manifold and a tangent vector ξ at this point

$$\gamma(t; x, \xi) : [0, 1] \rightarrow \mathcal{M}$$
$$\gamma(0; x, \xi) = x, \quad \dot{\gamma}(0, x, \xi) = \xi$$



- Exponential map

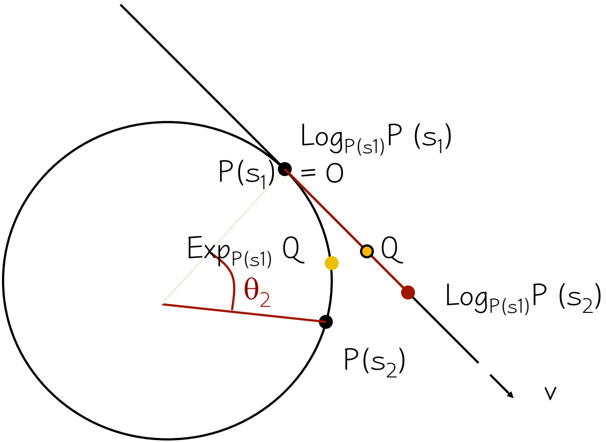
$$\text{Exp}_x : \mathcal{T}_x\mathcal{M} \rightarrow \mathcal{M} \quad \xi \longmapsto \gamma(1; x, \xi)$$

- Logarithm map (defined in a neighborhood $\mathcal{U}_x$ of x)

$$\text{Log}_x : \mathcal{U}_x \subset \mathcal{M} \rightarrow \mathcal{T}_x\mathcal{M} \quad y \longmapsto \text{Exp}_x^{-1}(y)$$

Interpolation on matrix manifolds

- Application to interpolation of points on a circle



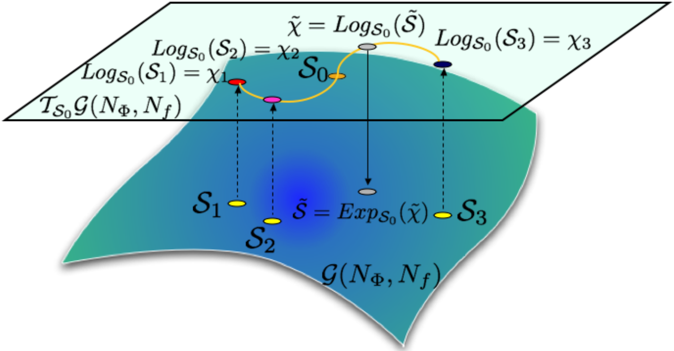
Interpolation on matrix manifolds

Case of the Grassmann manifold

- Logarithmic map
  1. Compute the thin SVD
  $$(\mathbf{I} - \mathbf{V}_0\mathbf{V}_0^T)\mathbf{V}_1(\mathbf{V}_0^T\mathbf{V}_1)^{-1} = \mathbf{U}\mathbf{\Sigma}\mathbf{Z}^T$$
  2. Compute
  $$\mathbf{\Gamma} = \mathbf{U}\tan^{-1}(\mathbf{\Sigma})\mathbf{Z}^T$$
  3. $\text{Log}_{\mathcal{S}_0}(\mathcal{S}_1) \leftrightarrow \mathbf{\Gamma}$
- Exponential map of $\boldsymbol{\xi} \in \mathcal{T}_{\mathcal{S}_0}\mathcal{G} \leftrightarrow \mathbf{\Gamma}$
  1. Compute the thin SVD
  $$\mathbf{\Gamma} = \mathbf{U}\mathbf{\Sigma}\mathbf{Z}^T$$
  2. Compute
  $$\mathbf{V} = \mathbf{V}_0\mathbf{Z}\cos\mathbf{\Sigma} + \mathbf{U}\sin\mathbf{\Sigma}$$
  3. $\text{Exp}_{\mathcal{S}_0}(\boldsymbol{\xi}) = \text{range}(\mathbf{V})$
- Note: the trigonometric operators only apply to the diagonal elements of the matrices

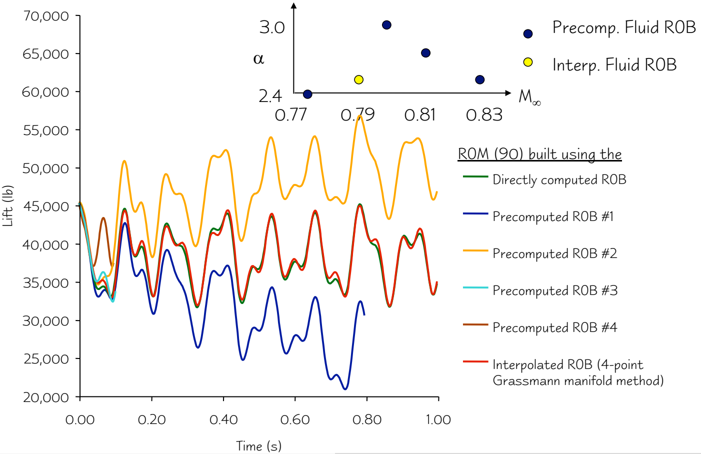
Interpolation on matrix manifolds

- Case of the Grassmann manifold



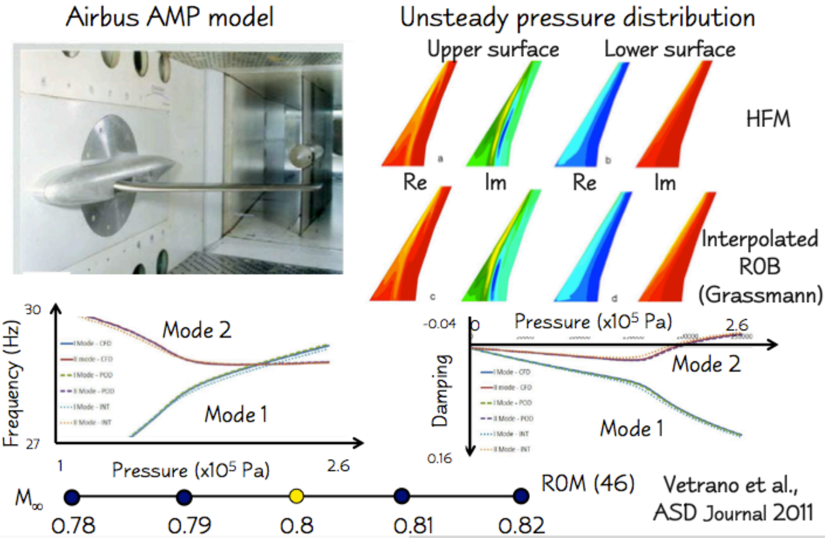
Application to aeroelasticity

- Aeroelastic behavior of the F-16



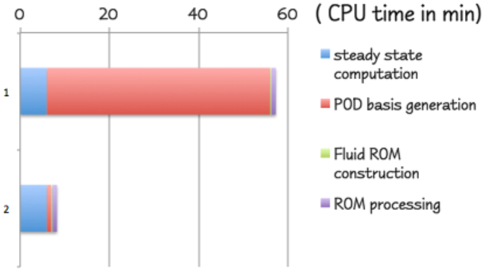
Application to aeroelasticity

- Aeroelastic behavior a commercial aircraft



Application to aeroelasticity

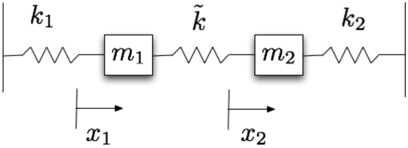
- Aeroelastic behavior of the F-16



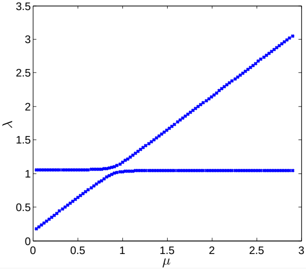
- Dominant cost: computation of $\mathbf{A}(\boldsymbol{\mu})$ and $\mathbf{B}(\boldsymbol{\mu})$ for a new value of $\boldsymbol{\mu}$
- Approach #2: interpolate $(\mathbf{V}(\boldsymbol{\mu})^T\mathbf{A}(\boldsymbol{\mu})\mathbf{V}(\boldsymbol{\mu}), \mathbf{V}(\boldsymbol{\mu})^T\mathbf{B}(\boldsymbol{\mu}))$

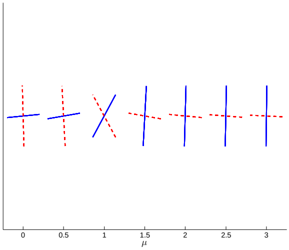
Example

- Simple example: mass-spring system with two degrees of freedom
- $\mu = k_1 - 0.1$



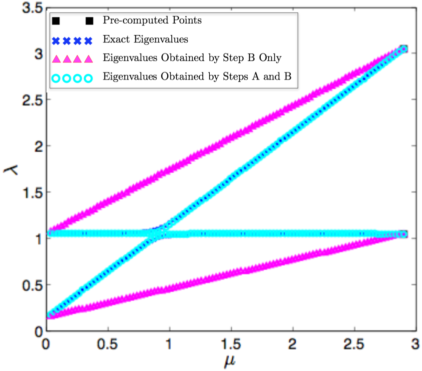
- $\mathbf{V}(\boldsymbol{\mu})^T \mathbf{A}(\boldsymbol{\mu}) \mathbf{V}(\boldsymbol{\mu}) = \mathbf{\Lambda}(\boldsymbol{\mu})$





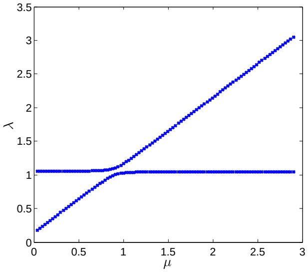
Interpolation on a matrix manifold

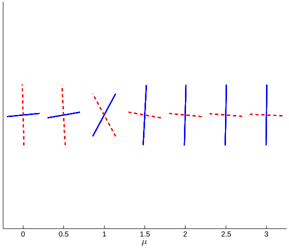
- $\mathbf{\Lambda}(\boldsymbol{\mu})$ belongs to the manifold of symmetric positive definite matrices (diagonal)
- Interpolate on the manifold using $(\mathbf{\Lambda}(0), \mathbf{\Lambda}(2.9))$



Example: Mode veering and mode crossing

- The issue is the mode veering: the coordinates of the reduced matrices are not consistent





- There would be an issue also with mode crossing (the eigenfrequencies are ordered increasingly in $\mathbf{\Lambda}$)

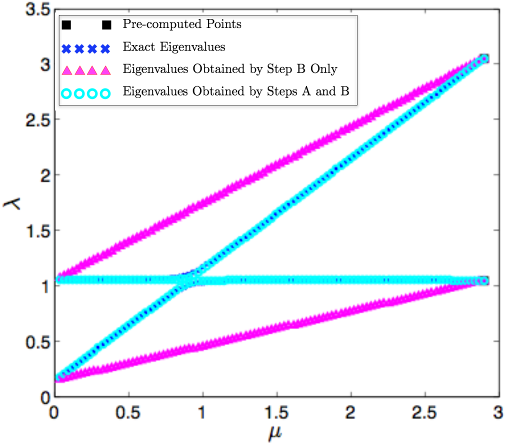
Consistent interpolation on matrix manifolds

- Solution: pre-process the reduced matrices (Step A)
- Consistency enforced by the solution of an orthogonal Procrustes problem

$$\min_{\mathbf{Q}_i \;\; \mathbf{Q}_i^T\mathbf{Q}_i=\mathbf{I}_k} \|\mathbf{V}_i\mathbf{Q}_i - \mathbf{V}_{i_0}\|_F, \; \forall i = 1, \cdots, m$$

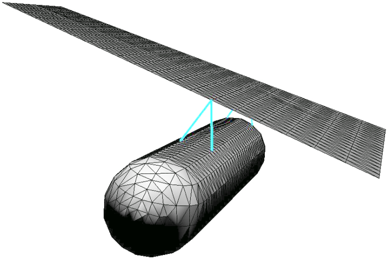
- Analytical solution using the SVD
  1. Compute $\mathbf{P}_{i,i_0} = \mathbf{V}_i^T\mathbf{V}_{i_0}$
  2. Compute the SVD $\mathbf{P}_{i,i_0} = \mathbf{U}_{i,i_0}\mathbf{\Sigma}_{i,i_0}\mathbf{Z}_{i,i_0}^T$
  3. Let $\mathbf{Q}_i = \mathbf{U}_{i,i_0}\mathbf{Z}_{i,i_0}^T$
- Can be processed online or offline
- Step B: interpolation on a matrix manifold

Application 1

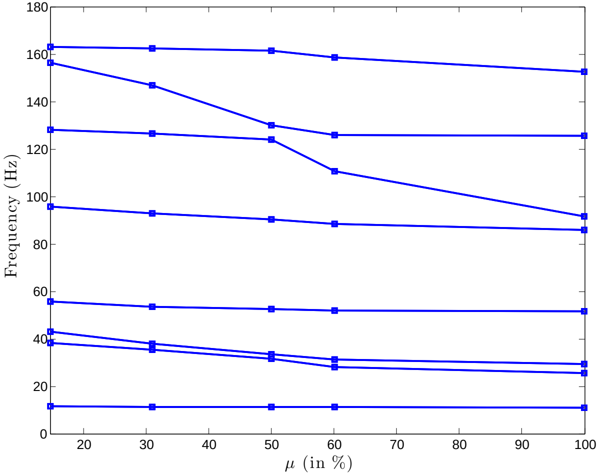


Application 2

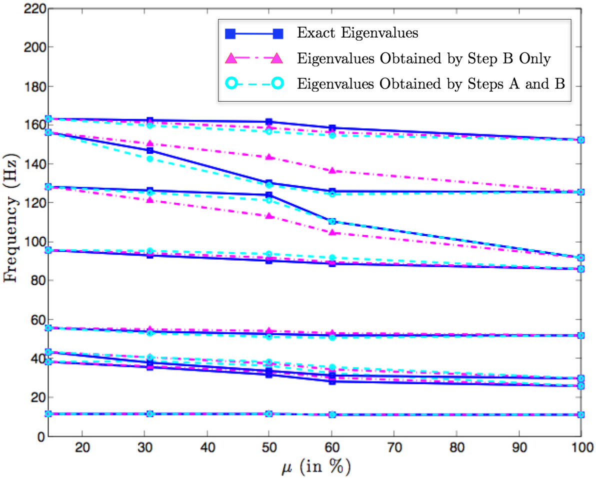
- More challenging example: wing with tank and sloshing effect
- The hydro-elastic effects affect the eigen-frequencies and eigen-modes of the structure
- The parameter μ defines the level of fuel in the tank $0 \leq \mu \leq 100\%$



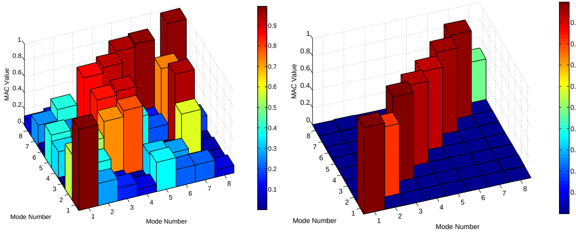
Application 2



Application 2



Link with Modal Assurance Criterion



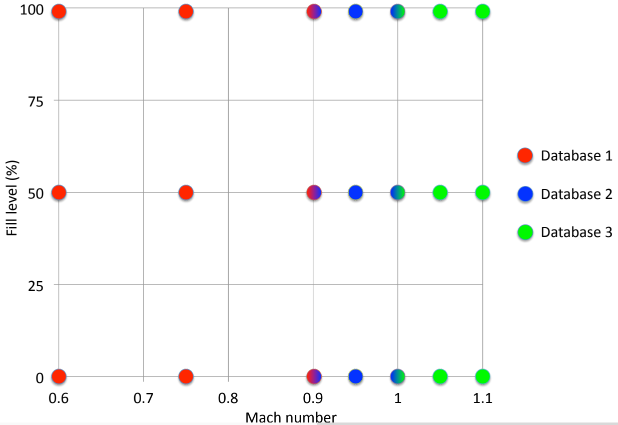
- The MAC between two eigenmodes ϕ and ψ is

$$\text{MAC}(\boldsymbol{\phi}, \boldsymbol{\psi}) = \frac{|\boldsymbol{\phi}^T \boldsymbol{\psi}|^2}{(\boldsymbol{\phi}^T \boldsymbol{\phi})(\boldsymbol{\psi}^T \boldsymbol{\psi})}$$

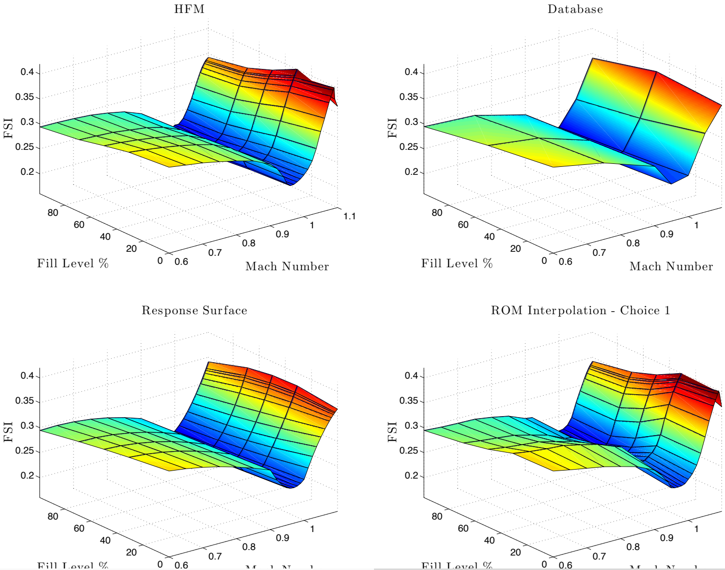
- When ϕ and ψ are normalized $\text{MAC}(\boldsymbol{\phi}, \boldsymbol{\psi}) = |\boldsymbol{\phi}^T \boldsymbol{\psi}|^2$
- $\mathbf{P}_{i,i_0}$ is the matrix of square roots of the MACs between the modes contained in $\mathbf{V}(\boldsymbol{\mu}_i)$ and those contained in $\mathbf{V}(\boldsymbol{\mu}_{i_0})$.
- This is the Modal Assurance Criterion Square Root (MACSR)

Application 3

- Aeroelastic study of the wing-tank system
- 2 parameters: fill level and free-stream Mach number M_∞
- Database approach

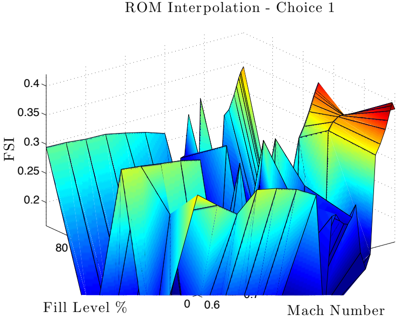


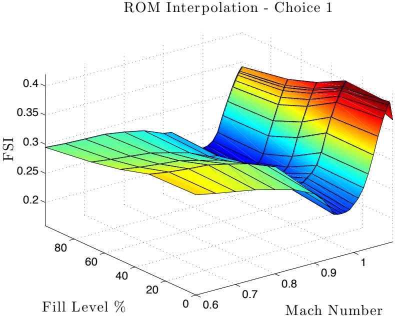
Application 3



Application 3

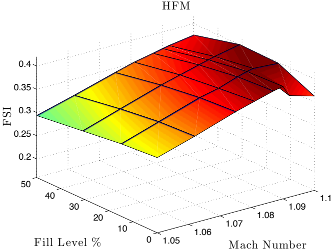
- Effect of Step A

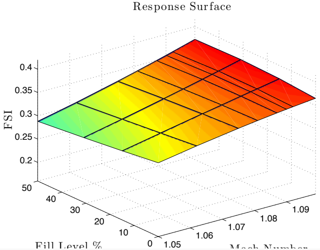


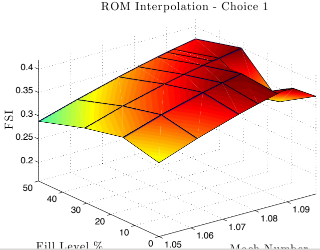


Application 3

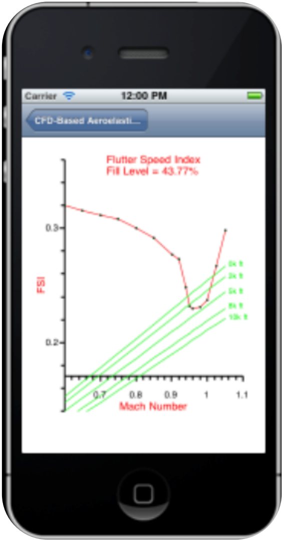
- Bifurcation detection







Mobile computing using a database of ROMs



References

- D. Amsallem, C. Farhat: Interpolation method for adapting reduced-order models and application to aeroelasticity. AIAA Journal 46(7), 1803–1813 (2008)
- D. Amsallem, C. Farhat: An online method for interpolating linear parametric reduced- order models. SIAM Journal on Scientific Computing 33(5), 2169–2198 (2011)
- D. Amsallem, Interpolation on Manifolds of CFD-based Fluid and Finite Element-based Structural Reduced-order Models for On-line Aeroelastic Prediction, PhD thesis, Stanford University (2010)