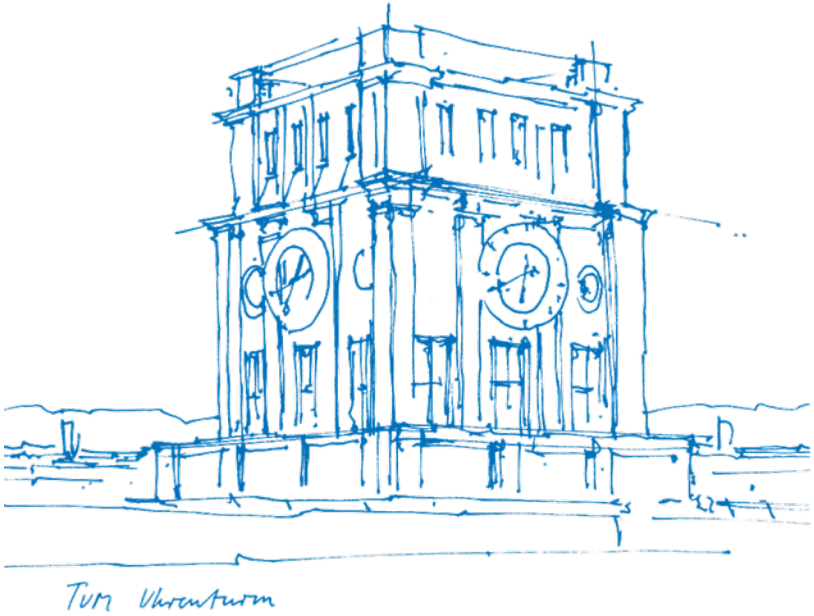
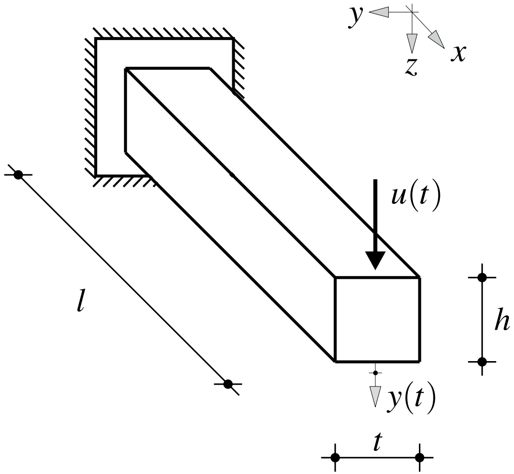
Removing Inconsistencies of Reduced Bases in Parametric Model Order Reduction by Matrix Interpolation

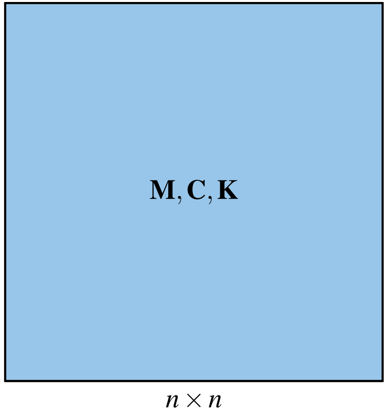
S. Schopper[1], and G. Müller[1]

[1]Technical University of Munich, School of Engineering and Design, Chair of Structural Mechanics



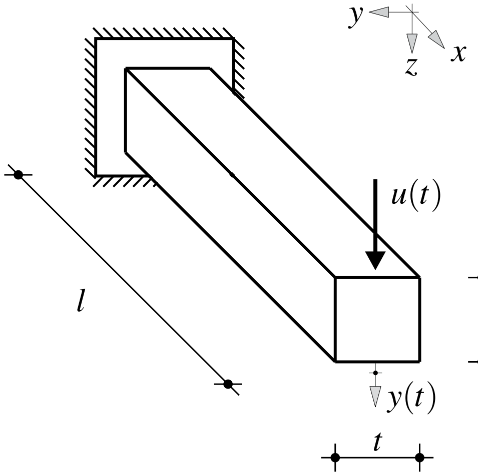
Motivation



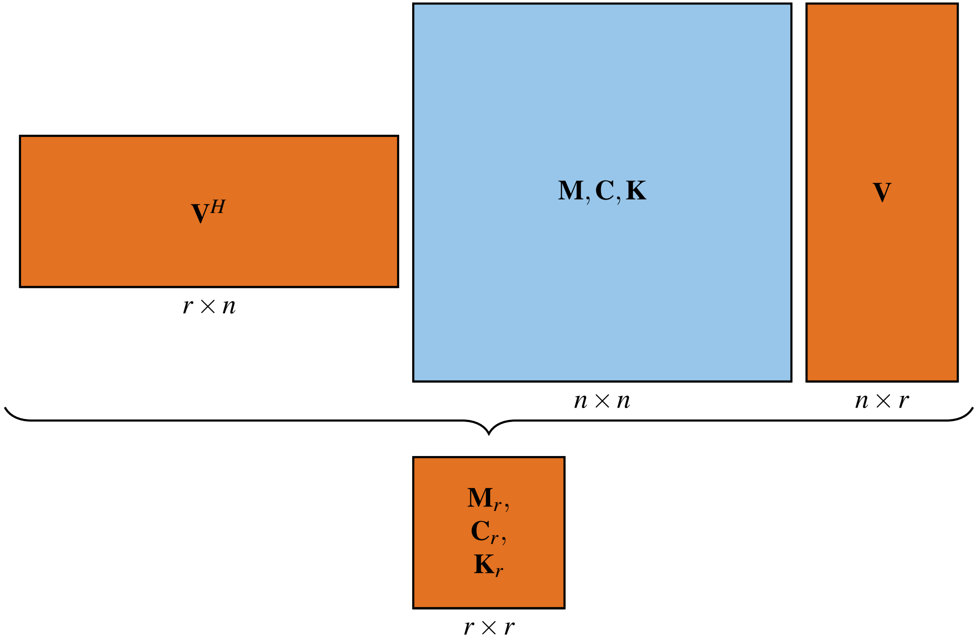


$$\Sigma : \begin{cases} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) &= \mathbf{f}u(t), \\ y(t) &= \mathbf{g}\mathbf{x}(t). \end{cases}$$

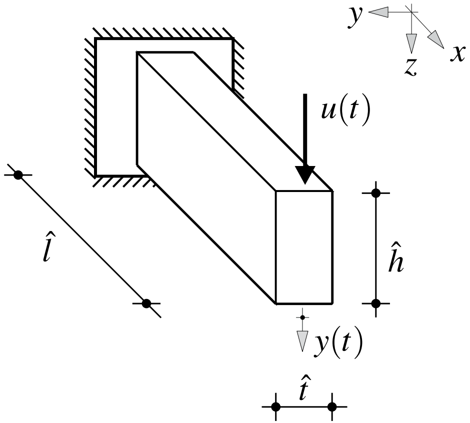
Motivation

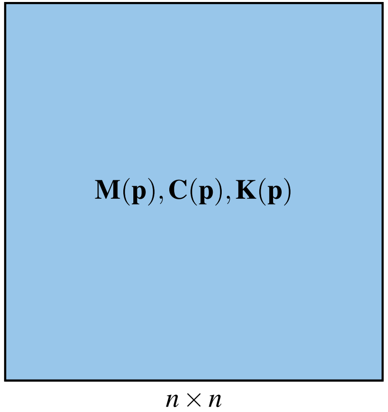


$$\Sigma : \begin{cases} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) &= \mathbf{f}u(t), \\ y(t) &= \mathbf{g}\mathbf{x}(t). \end{cases}$$



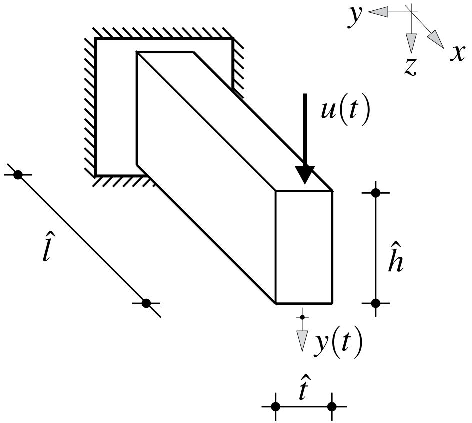
Motivation

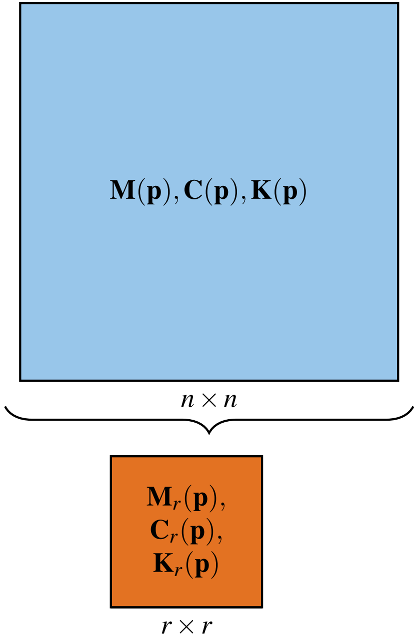




$$\Sigma(\mathbf{p}) : \begin{cases} \mathbf{M}(\mathbf{p})\ddot{\mathbf{x}}(t) + \mathbf{C}(\mathbf{p})\dot{\mathbf{x}}(t) + \mathbf{K}(\mathbf{p})\mathbf{x}(t) & = \mathbf{f}(\mathbf{p})u(t), \\ y(t) & = \mathbf{g}(\mathbf{p})\mathbf{x}(t). \end{cases}$$

Motivation





$$\Sigma(\mathbf{p}) : \begin{cases} \mathbf{M}(\mathbf{p})\ddot{\mathbf{x}}(t) + \mathbf{C}(\mathbf{p})\dot{\mathbf{x}}(t) + \mathbf{K}(\mathbf{p})\mathbf{x}(t) & = \mathbf{f}(\mathbf{p})u(t), \\ y(t) & = \mathbf{g}(\mathbf{p})\mathbf{x}(t). \end{cases}$$

Outline

- Mathematical System Description
- Parametric Model Order Reduction by Matrix Interpolation
- Removal of Inconsistencies via Adaptive Sampling and Clustering
- Results
- Conclusion and Future Work

Mathematical System Description

Parametric Dynamic Systems

Linear-time invariant, parametric dynamical systems with single input and single output (SISO) in second-order form are regarded:

$$\Sigma(\mathbf{p}) : \begin{cases} \mathbf{M}(\mathbf{p})\ddot{\mathbf{x}}(t) + \mathbf{C}(\mathbf{p})\dot{\mathbf{x}}(t) + \mathbf{K}(\mathbf{p})\mathbf{x}(t) & = \mathbf{f}(\mathbf{p})u(t), \\ y(t) & = \mathbf{g}(\mathbf{p})\mathbf{x}(t), \end{cases} \tag{1}$$

with mass, damping and stiffness matrix $\mathbf{M}(\mathbf{p}), \mathbf{C}(\mathbf{p}), \mathbf{K}(\mathbf{p}) \in \mathbb{R}^{n\times n}$, and input and output mapping $\mathbf{f}(\mathbf{p}) \in \mathbb{R}^{n\times 1}$ and $\mathbf{g}(\mathbf{p}) \in \mathbb{R}^{1\times n}$, which depend on d parameters $\mathbf{p} = [p_1, p_2, \dots, p_d]$. The vectors $\mathbf{x}(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}$ and $y(t) \in \mathbb{R}$ denote the state, input and output.

Parametric Dynamic Systems

Linear-time invariant, parametric dynamical systems with single input and single output (SISO) in second-order form are regarded:

$$\Sigma(\mathbf{p}) : \begin{cases} \mathbf{M}(\mathbf{p})\ddot{\mathbf{x}}(t) + \mathbf{C}(\mathbf{p})\dot{\mathbf{x}}(t) + \mathbf{K}(\mathbf{p})\mathbf{x}(t) & = \mathbf{f}(\mathbf{p})u(t), \\ y(t) & = \mathbf{g}(\mathbf{p})\mathbf{x}(t), \end{cases} \tag{1}$$

with mass, damping and stiffness matrix $\mathbf{M}(\mathbf{p}), \mathbf{C}(\mathbf{p}), \mathbf{K}(\mathbf{p}) \in \mathbb{R}^{n\times n}$, and input and output mapping $\mathbf{f}(\mathbf{p}) \in \mathbb{R}^{n\times 1}$ and $\mathbf{g}(\mathbf{p}) \in \mathbb{R}^{1\times n}$, which depend on d parameters $\mathbf{p} = [p_1, p_2, \ldots, p_d]$. The vectors $\mathbf{x}(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}$ and $y(t) \in \mathbb{R}$ denote the state, input and output.

After performing a Laplace transformation, the transfer function of the system can be computed as

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p}) \left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1} \mathbf{f}(\mathbf{p}), \tag{2}$$

with the complex frequency $s \in \mathbb{C}$.

Parametric Dynamic Systems

After performing a Laplace transformation, the transfer function of the system can be computed as

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{3}$$

with the complex frequency $s \in \mathbb{C}$.

We assume that **no** reasonable affine representation of the parametric dependency of the following form is available (exemplarily shown for the stiffness matrix):

$$\mathbf{K}(\mathbf{p}) = \mathbf{K}_0 + \sum_{i=1}^{M} f_i(\mathbf{p})\mathbf{K}_i, \qquad i = 1,\dots,M, \tag{4}$$

where $f_i(\mathbf{p})$ are scalar functions. [BGW15]

Objective

After performing a Laplace transformation, the transfer function of the system can be computed as

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{5}$$

with the complex frequency $s \in \mathbb{C}$.

We want to generate a parametric reduced-order model (pROM) that allows to solve the above transfer function efficiently and

Objective

After performing a Laplace transformation, the transfer function of the system can be computed as

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{5}$$

with the complex frequency $s \in \mathbb{C}$.

We want to generate a parametric reduced-order model (pROM) that allows to solve the above transfer function efficiently and

- does not require an affine representation of the parametric dependency,

Objective

After performing a Laplace transformation, the transfer function of the system can be computed as

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{5}$$

with the complex frequency $s \in \mathbb{C}$.

We want to generate a parametric reduced-order model (pROM) that allows to solve the above transfer function efficiently and

- does not require an affine representation of the parametric dependency,
- is valid for a large range of the parameters, and

Objective

After performing a Laplace transformation, the transfer function of the system can be computed as

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{5}$$

with the complex frequency $s \in \mathbb{C}$.

We want to generate a parametric reduced-order model (pROM) that allows to solve the above transfer function efficiently and

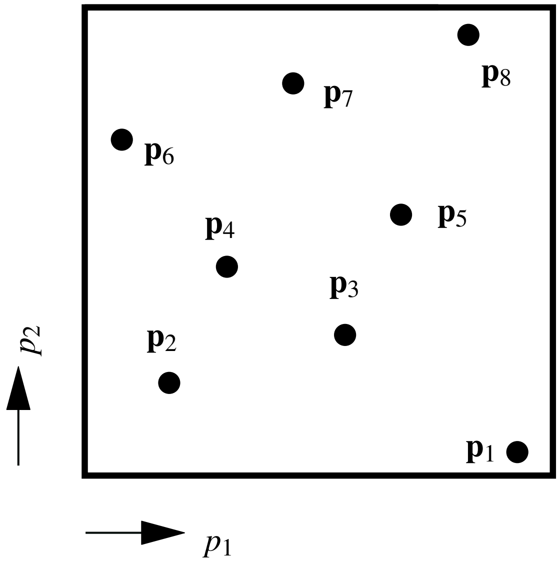
- does not require an affine representation of the parametric dependency,
- is valid for a large range of the parameters, and
- is generated via an adaptive algorithm.

Parametric Model Order Reduction by Matrix Interpolation

Parametric Model Order Reduction by Matrix Interpolation

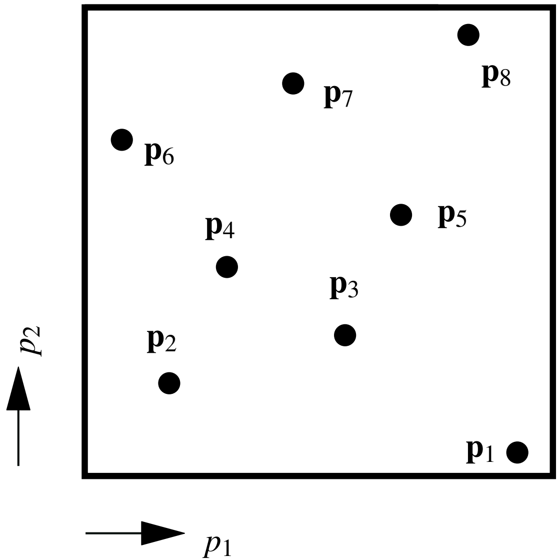
We follow the approach of pMOR by matrix interpolation by [PMEL10]:

$$\{\mathbf{M}(\mathbf{p}_k), \mathbf{C}(\mathbf{p}_k), \mathbf{K}(\mathbf{p}_k), \mathbf{f}(\mathbf{p}_k), \mathbf{g}(\mathbf{p}_k)\}$$



Parametric Model Order Reduction by Matrix Interpolation

We follow the approach of pMOR by matrix interpolation by [PMEL10]:



$$\{\mathbf{M}(\mathbf{p}_k), \mathbf{C}(\mathbf{p}_k), \mathbf{K}(\mathbf{p}_k), \mathbf{f}(\mathbf{p}_k), \mathbf{g}(\mathbf{p}_k)\}$$

$\Downarrow$ Project into $\mathbf{V}_k \in \mathbb{C}^{n\times r}$, $(\mathbf{x}(\mathbf{p}_k) \approx \mathbf{V}_k \mathbf{x}_r(\mathbf{p}_k))$

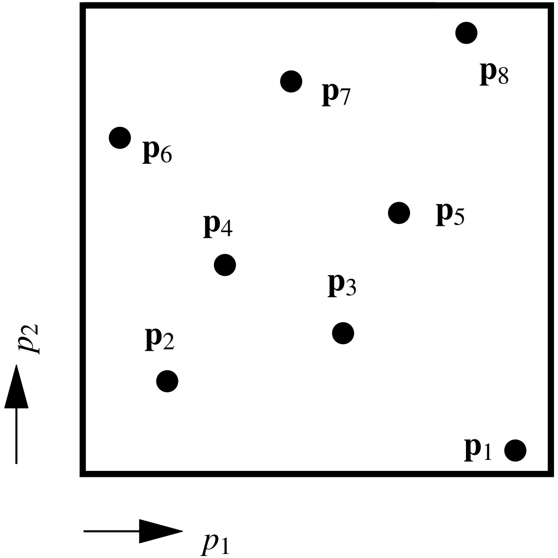
$$\{\mathbf{M}_r(\mathbf{p}_k), \mathbf{C}_r(\mathbf{p}_k), \mathbf{K}_r(\mathbf{p}_k), \mathbf{f}_r(\mathbf{p}_k), \mathbf{g}_r(\mathbf{p}_k)\}$$

with

$$\mathbf{K}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{K}(\mathbf{p}_k)\mathbf{V}_k, \quad \mathbf{M}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{M}(\mathbf{p}_k)\mathbf{V}_k, \quad \mathbf{C}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{C}(\mathbf{p}_k)\mathbf{V}_k,$$
$$\mathbf{f}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{f}(\mathbf{p}_k), \quad \mathbf{g}_r(\mathbf{p}_k) = \mathbf{g}(\mathbf{p}_k)\mathbf{V}_k,$$

Parametric Model Order Reduction by Matrix Interpolation

We follow the approach of pMOR by matrix interpolation by [PMEL10]:



$$\{\mathbf{M}(\mathbf{p}_k), \mathbf{C}(\mathbf{p}_k), \mathbf{K}(\mathbf{p}_k), \mathbf{f}(\mathbf{p}_k), \mathbf{g}(\mathbf{p}_k)\}$$

$\Downarrow$ Project into $\mathbf{V}_k \in \mathbb{C}^{n\times r}$, $(\mathbf{x}(\mathbf{p}_k) \approx \mathbf{V}_k \mathbf{x}_r(\mathbf{p}_k))$

$$\{\mathbf{M}_r(\mathbf{p}_k), \mathbf{C}_r(\mathbf{p}_k), \mathbf{K}_r(\mathbf{p}_k), \mathbf{f}_r(\mathbf{p}_k), \mathbf{g}_r(\mathbf{p}_k)\}$$

with

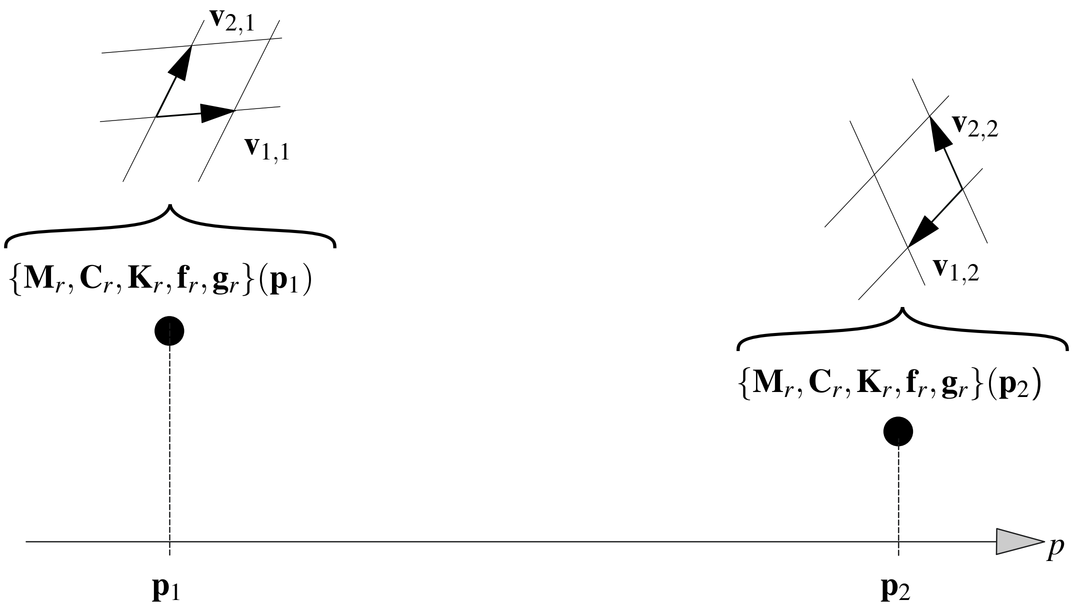
$$\mathbf{K}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{K}(\mathbf{p}_k)\mathbf{V}_k, \qquad \mathbf{M}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{M}(\mathbf{p}_k)\mathbf{V}_k, \qquad \mathbf{C}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{C}(\mathbf{p}_k)\mathbf{V}_k,$$
$$\mathbf{f}_r(\mathbf{p}_k) = \mathbf{V}_k^{\mathsf{H}}\mathbf{f}(\mathbf{p}_k), \qquad \mathbf{g}_r(\mathbf{p}_k) = \mathbf{g}(\mathbf{p}_k)\mathbf{V}_k,$$

where

$$(\mathbf{K}(\mathbf{p}_k) - \omega^2\mathbf{M}(\mathbf{p}_k))\boldsymbol{\phi} = \mathbf{0}$$
$$\mathbf{V}_k = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \dots, \boldsymbol{\phi}_r].$$

Parametric Model Order Reduction by Matrix Interpolation

1. Sampling of local reduced systems

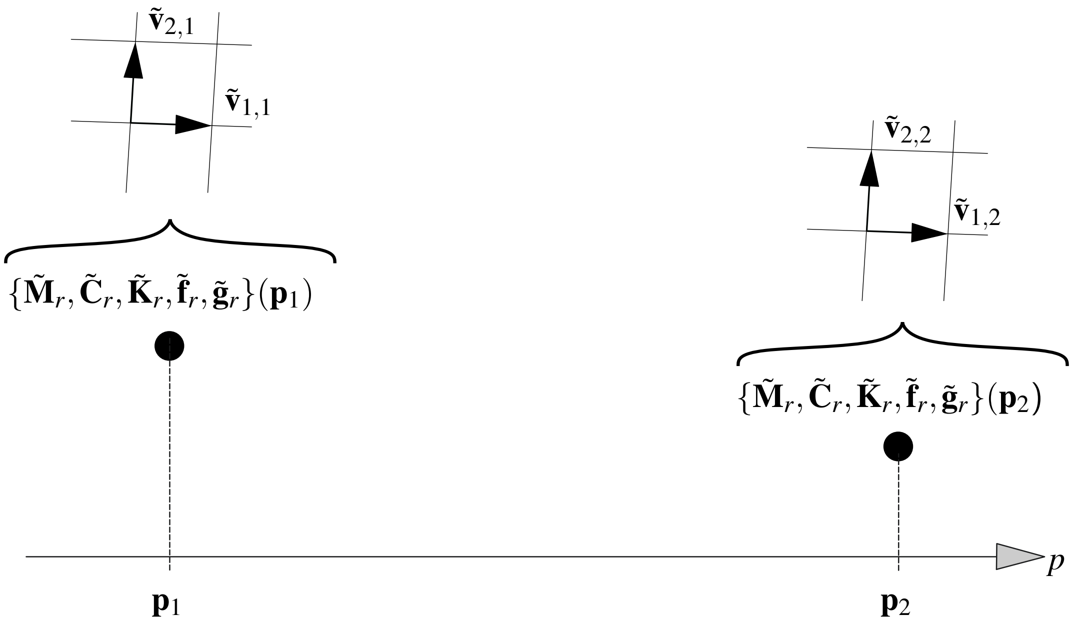


Parametric Model Order Reduction by Matrix Interpolation

1. Sampling of local reduced systems

↓

2. Transformation to generalized coordinate system



Parametric Model Order Reduction by Matrix Interpolation

For a meaningful interpolation, the reduced operators should be in the same coordinate system. To achieve this, the following approach was suggested in [PMEL10]:

1. Find a generalized coordinate system. For this purpose, find the most significant basis vectors by concatenating all N sampled bases and then performing an SVD:

   $$[\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_N] = \mathbf{U\Sigma Y}, \qquad \mathbf{V}_k \in \mathbb{C}^{n\times r}, \quad k = 1, \ldots, N \tag{6}$$

   The most significant basis vectors are the first r columns in $\mathbf{U}$ and denoted with $\mathbf{R}$:

   $$\mathbf{R} = \mathbf{U}(:, 1:r). \tag{7}$$

Parametric Model Order Reduction by Matrix Interpolation

For a meaningful interpolation, the reduced operators should be in the same coordinate system. To achieve this, the following approach was suggested in [PMEL10]:

1. Find a generalized coordinate system. For this purpose, find the most significant basis vectors by concatenating all N sampled bases and then performing an SVD:

$$[\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_N] = \mathbf{U}\boldsymbol{\Sigma}\mathbf{Y}, \qquad \mathbf{V}_k \in \mathbb{C}^{n \times r}, \quad k = 1, \ldots, N \tag{6}$$

   The most significant basis vectors are the first r columns in $\mathbf{U}$ and denoted with $\mathbf{R}$:

$$\mathbf{R} = \mathbf{U}(:, 1:r). \tag{7}$$

2. Transform the individual reduced operators from their individual bases $\mathbf{V}_k$ to the generalized coordinate system $\mathbf{R}$:

$$\tilde{\mathbf{K}}_r(\mathbf{p}_k) = \mathbf{T}_k^\mathsf{T}\mathbf{K}_r(\mathbf{p}_k)\mathbf{T}_k, \quad \tilde{\mathbf{C}}_r(\mathbf{p}_k) = \mathbf{T}_k^\mathsf{T}\mathbf{C}_r(\mathbf{p}_k)\mathbf{T}_k, \quad \tilde{\mathbf{M}}_r(\mathbf{p}_k) = \mathbf{T}_k^\mathsf{T}\mathbf{M}_r(\mathbf{p}_k)\mathbf{T}_k, \quad \tilde{\mathbf{f}}_r(\mathbf{p}_k) = \mathbf{T}_k^\mathsf{T}\mathbf{f}_r(\mathbf{p}_k), \quad \tilde{\mathbf{g}}_r(\mathbf{p}_k) = \mathbf{g}_r(\mathbf{p}_k)\mathbf{T}_k, \tag{8}$$

   with

$$\mathbf{T}_k = (\mathbf{R}^T\mathbf{V}_k)^{-1}, \quad \tilde{\mathbf{V}}_k = \mathbf{V}_k\mathbf{T}_k. \tag{9}$$

Parametric Model Order Reduction by Matrix Interpolation

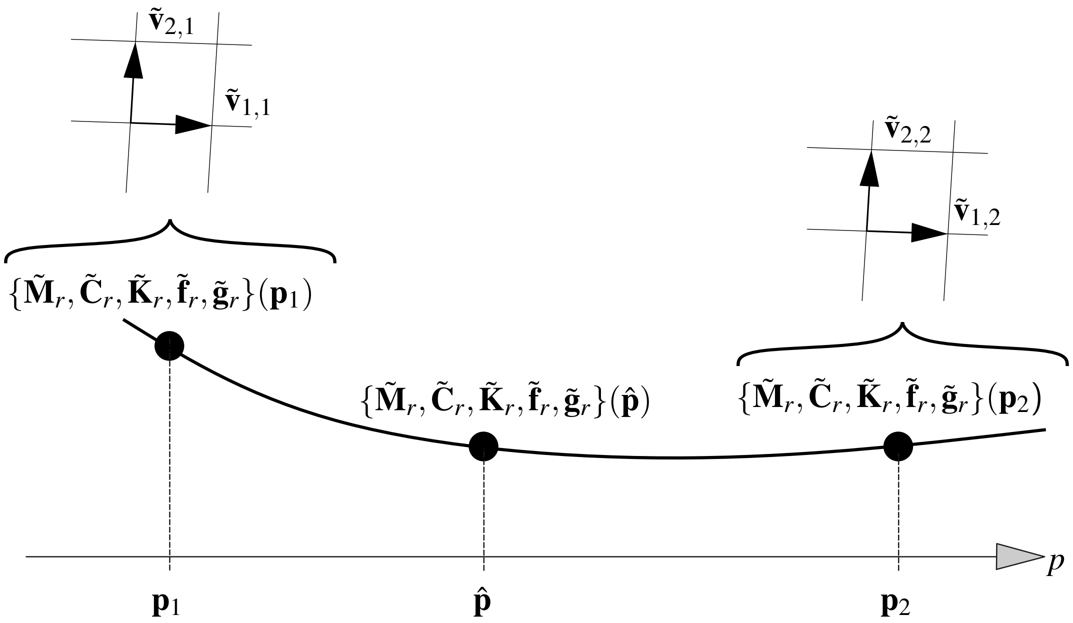
1. Sampling of local reduced systems

↓

2. Transformation to generalized coordinate system

↓

3. Interpolation/ regression of reduced operators



Inconsistencies in Reduced Bases

In the transformation, the vectors of the reduced basis are only reordered, but the subspace they span stays the same:

$$\tilde{\mathbf{V}}_k = \mathbf{V}_k \mathbf{T}_k. \tag{10}$$

Inconsistencies in Reduced Bases

In the transformation, the vectors of the reduced basis are only reordered, but the subspace they span stays the same:

$$\tilde{\mathbf{V}}_k = \mathbf{V}_k \mathbf{T}_k. \tag{10}$$

Therefore, the reduced dependency that shall be learned does not only depend on the change of the full operators, but also on the change of the reduced basis:

$$\tilde{\mathbf{K}}_r(\mathbf{p}) = \tilde{\mathbf{V}}(\mathbf{p})^{\mathsf{H}} \mathbf{K}(\mathbf{p}) \tilde{\mathbf{V}}(\mathbf{p}) \tag{11}$$

Inconsistencies in Reduced Bases

In the transformation, the vectors of the reduced basis are only reordered, but the subspace they span stays the same:

$$\tilde{\mathbf{V}}_k = \mathbf{V}_k \mathbf{T}_k. \tag{10}$$

Therefore, the reduced dependency that shall be learned does not only depend on the change of the full operators, but also on the change of the reduced basis:

$$\tilde{\mathbf{K}}_r(\mathbf{p}) = \tilde{\mathbf{V}}(\mathbf{p})^{\mathsf{H}} \mathbf{K}(\mathbf{p}) \tilde{\mathbf{V}}(\mathbf{p}) \tag{11}$$

Strong changes in the reduced bases introduce inconsistencies in the training data for the matrix interpolation. They can occur due to several reasons:

- Model Order Reduction method used [FE15]

Inconsistencies in Reduced Bases

In the transformation, the vectors of the reduced basis are only reordered, but the subspace they span stays the same:

$$\tilde{\mathbf{V}}_k = \mathbf{V}_k \mathbf{T}_k. \tag{10}$$

Therefore, the reduced dependency that shall be learned does not only depend on the change of the full operators, but also on the change of the reduced basis:

$$\tilde{\mathbf{K}}_r(\mathbf{p}) = \tilde{\mathbf{V}}(\mathbf{p})^{\mathsf{H}} \mathbf{K}(\mathbf{p}) \tilde{\mathbf{V}}(\mathbf{p}) \tag{11}$$

Strong changes in the reduced bases introduce inconsistencies in the training data for the matrix interpolation. They can occur due to several reasons:

- Model Order Reduction method used [FE15]
- Change of the system dynamics [BNN$^+$15]

Inconsistencies in Reduced Bases

In the transformation, the vectors of the reduced basis are only reordered, but the subspace they span stays the same:

$$\tilde{\mathbf{V}}_k = \mathbf{V}_k \mathbf{T}_k. \tag{10}$$

Therefore, the reduced dependency that shall be learned does not only depend on the change of the full operators, but also on the change of the reduced basis:

$$\tilde{\mathbf{K}}_r(\mathbf{p}) = \tilde{\mathbf{V}}(\mathbf{p})^{\mathsf{H}} \mathbf{K}(\mathbf{p}) \tilde{\mathbf{V}}(\mathbf{p}) \tag{11}$$

Strong changes in the reduced bases introduce inconsistencies in the training data for the matrix interpolation. They can occur due to several reasons:

- Model Order Reduction method used [FE15]
- Change of the system dynamics [BNN$^+$15]
- Mode switching and truncation [ATF15]

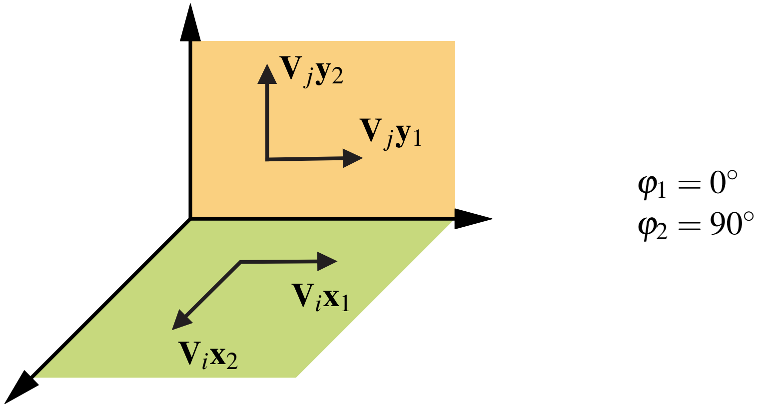
Detection of Inconsistencies

The angles between the subspaces spanned by the two orthonormal bases $\mathbf{V}_i$ and $\mathbf{V}_j$ are computed by first performing an SVD on the following product [ATF15]:

$$\mathbf{V}_i^{\mathrm{H}}\mathbf{V}_j = \mathbf{X}\boldsymbol{\Sigma}\mathbf{Y}^{\mathsf{T}}, \qquad i,j = 1,\ldots,N \tag{12}$$

The subspace angles can then be found as

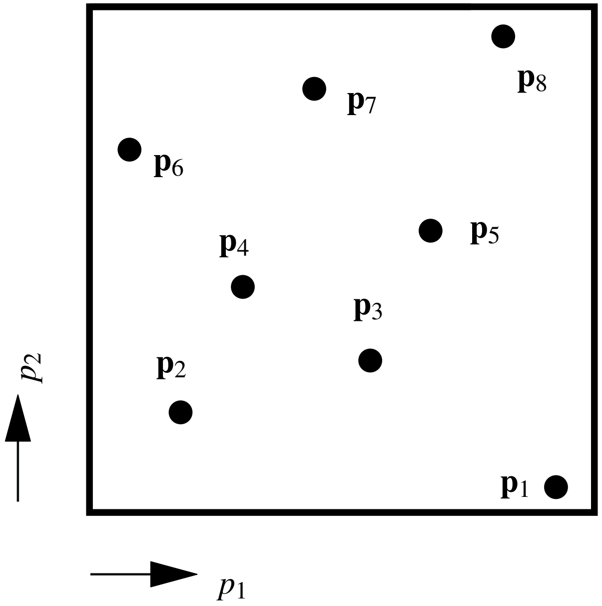
$$\varphi_l = \arccos(\sigma_l), \qquad l = 1,\ldots,r. \tag{13}$$



Removal of Inconsistencies via Adaptive Sampling and Clustering

Adaptive Sampling and Clustering

1. Sampling of local reduced systems
2. Transformation to generalized coordinate system
3. Interpolation/ regression of reduced operators



Adaptive Sampling and Clustering

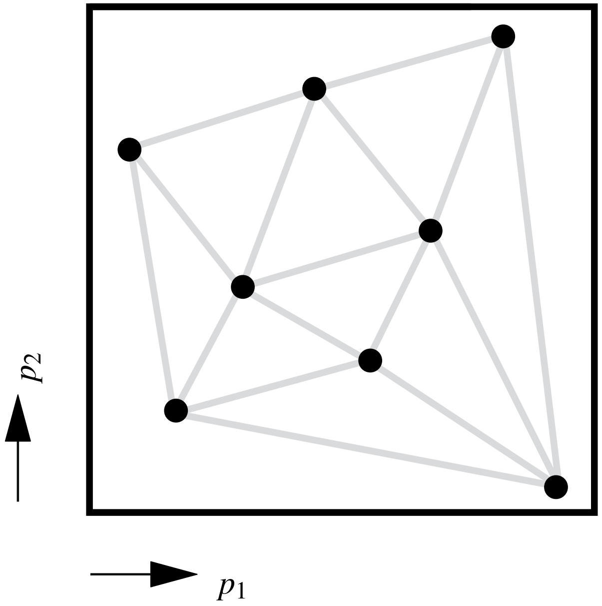
1. Sampling of local reduced systems → 1.1 Triangulation

↓

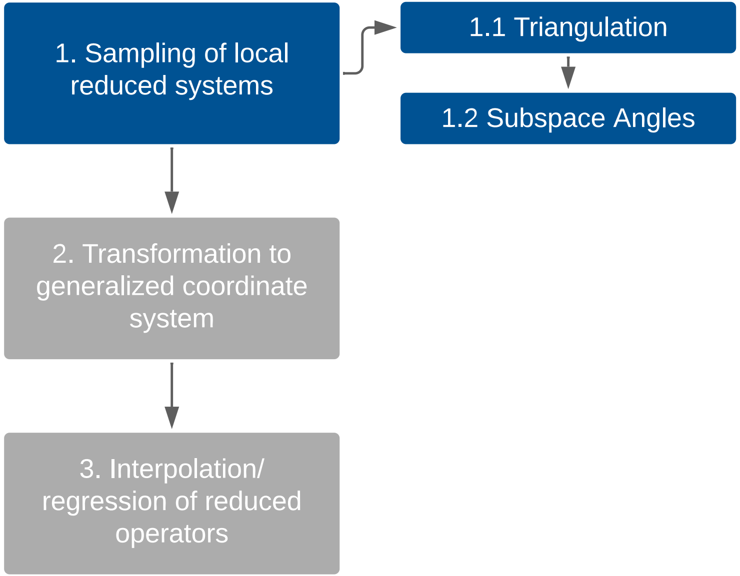
2. Transformation to generalized coordinate system

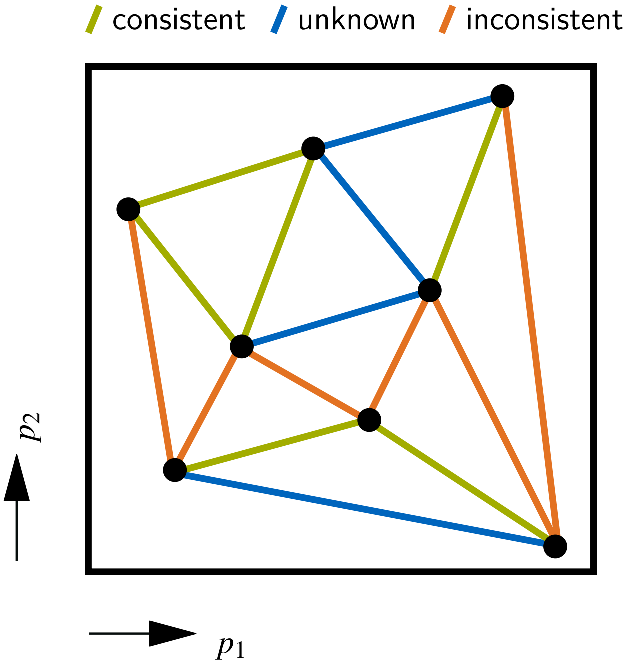
↓

3. Interpolation/ regression of reduced operators

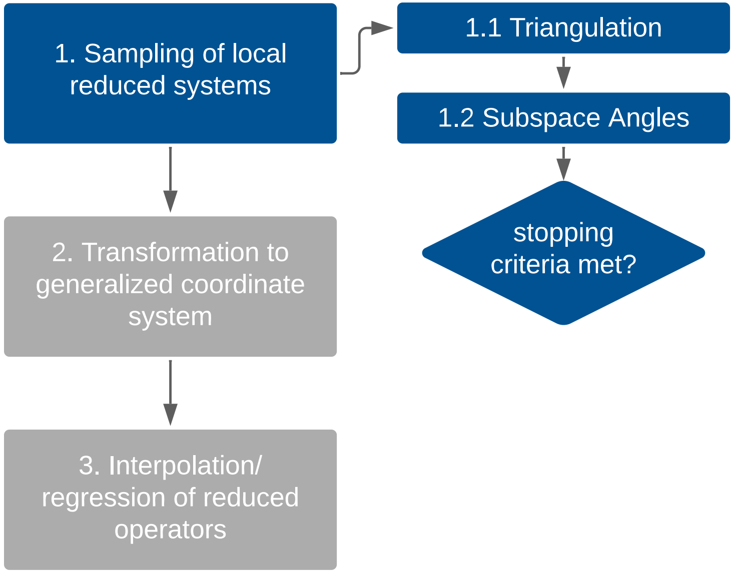


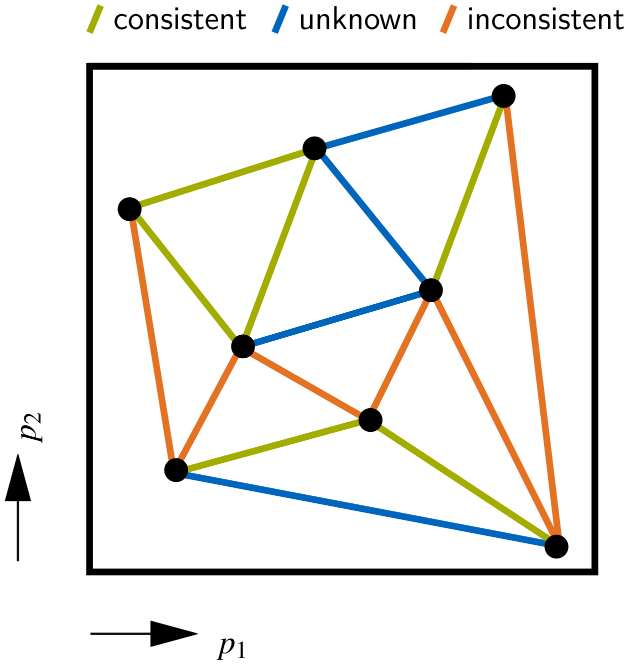
Adaptive Sampling and Clustering



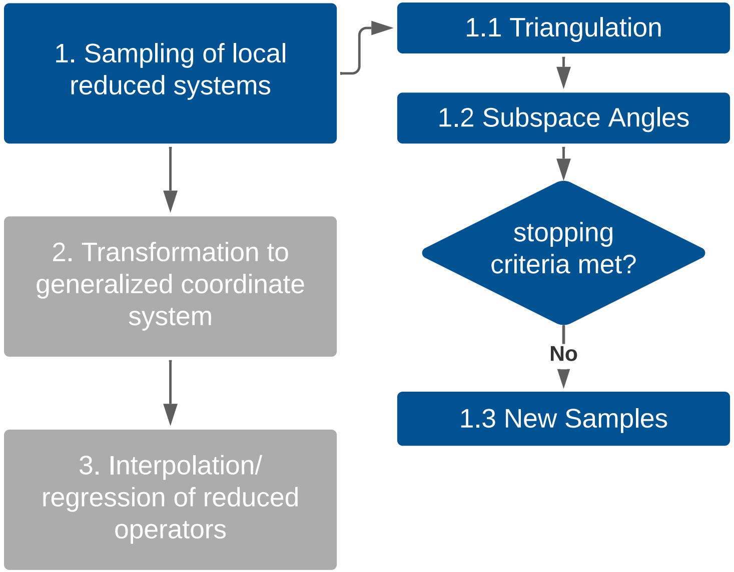


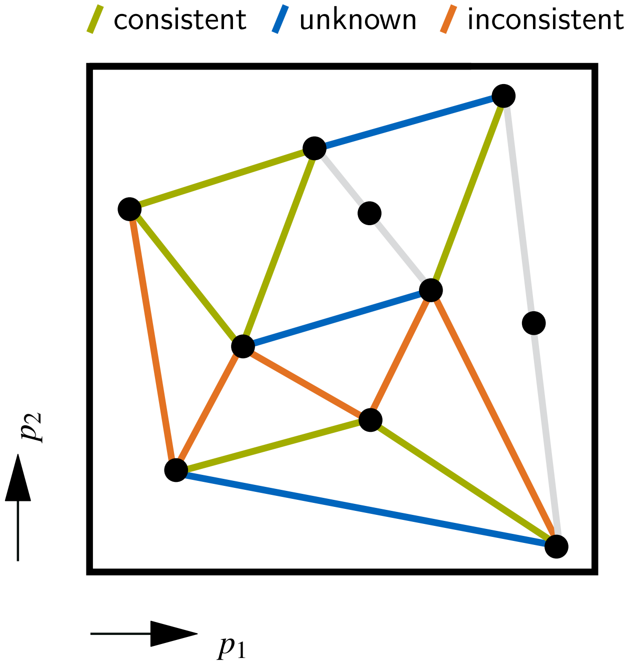
Adaptive Sampling and Clustering



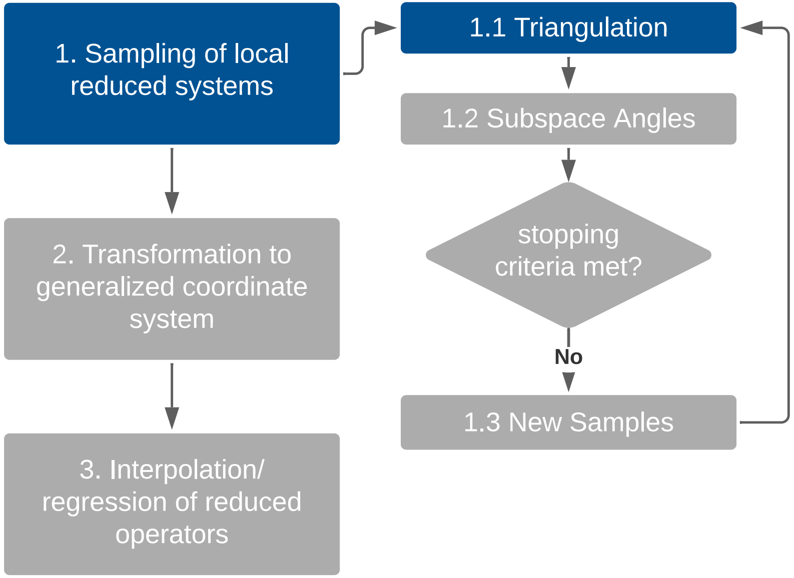


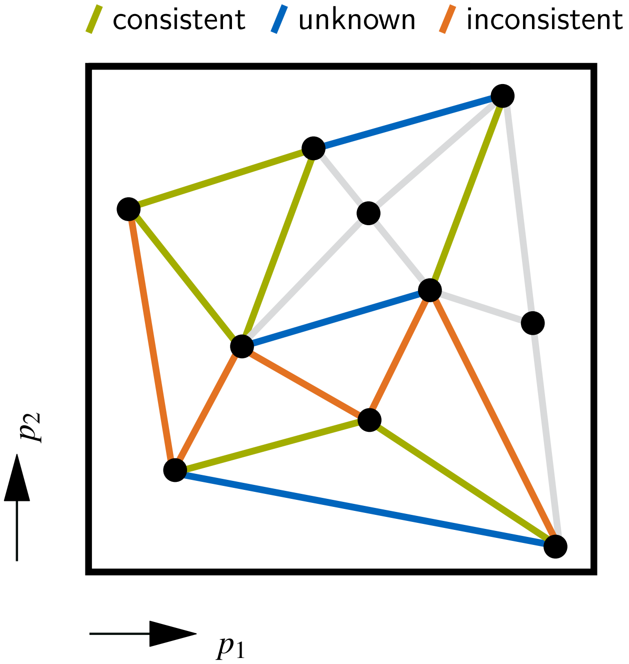
Adaptive Sampling and Clustering



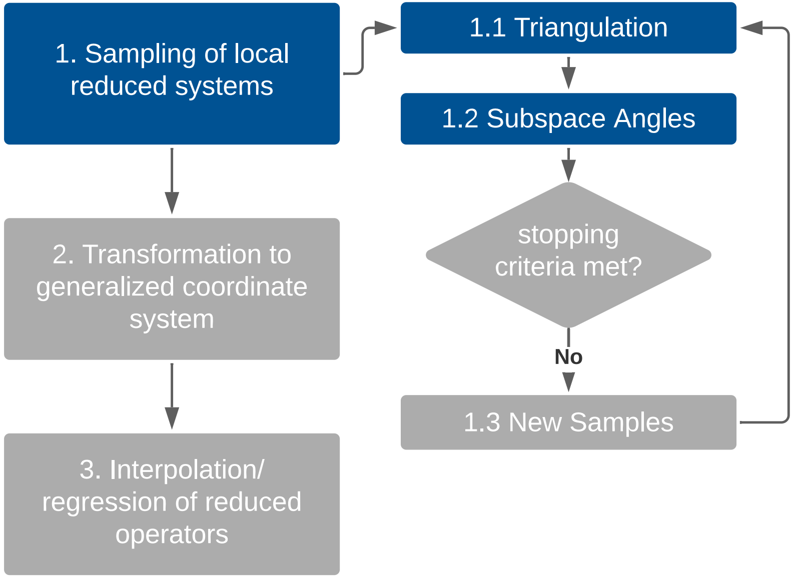


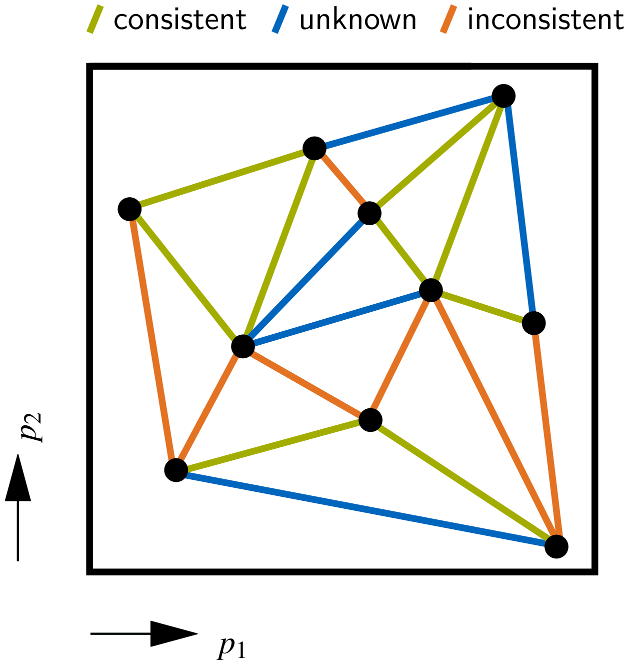
Adaptive Sampling and Clustering



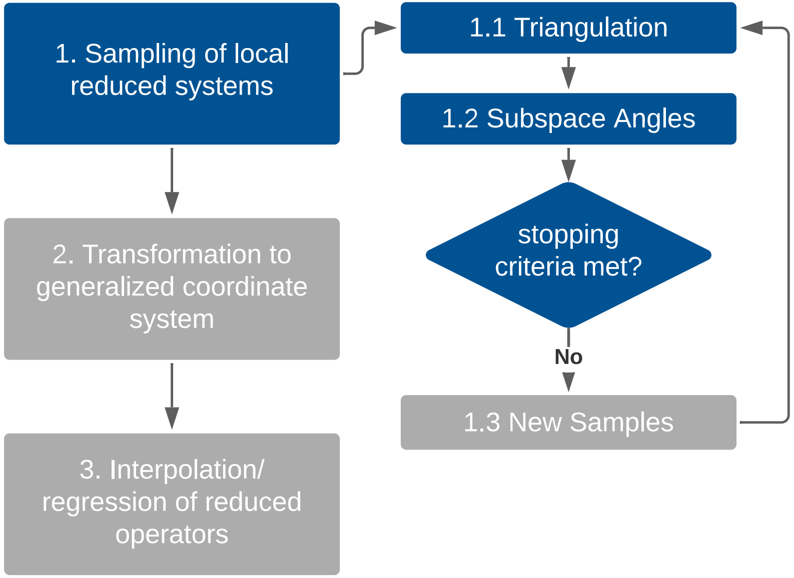


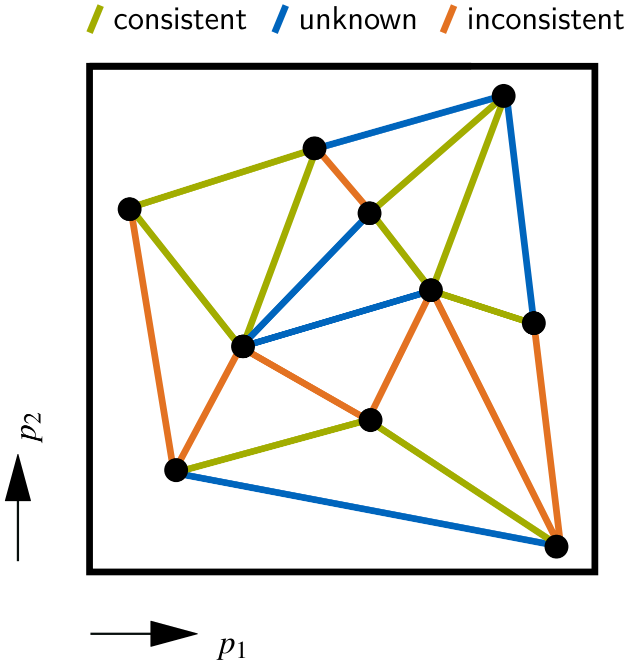
Adaptive Sampling and Clustering



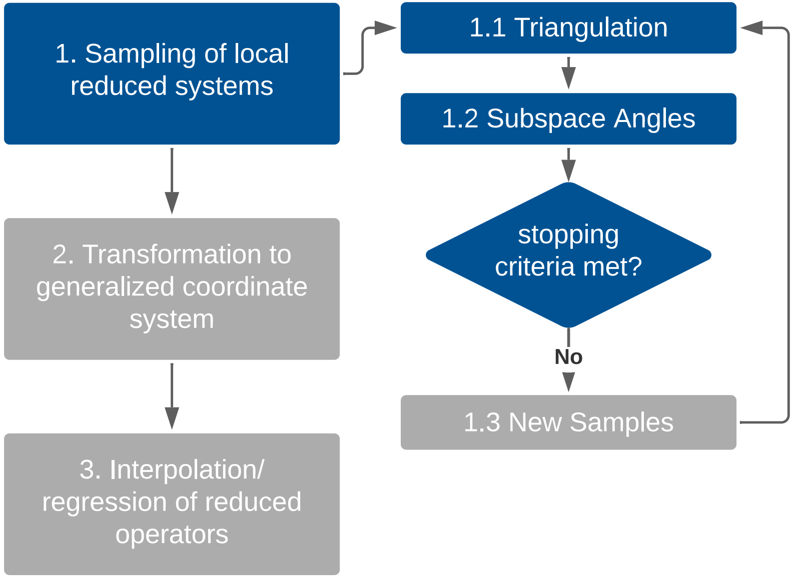


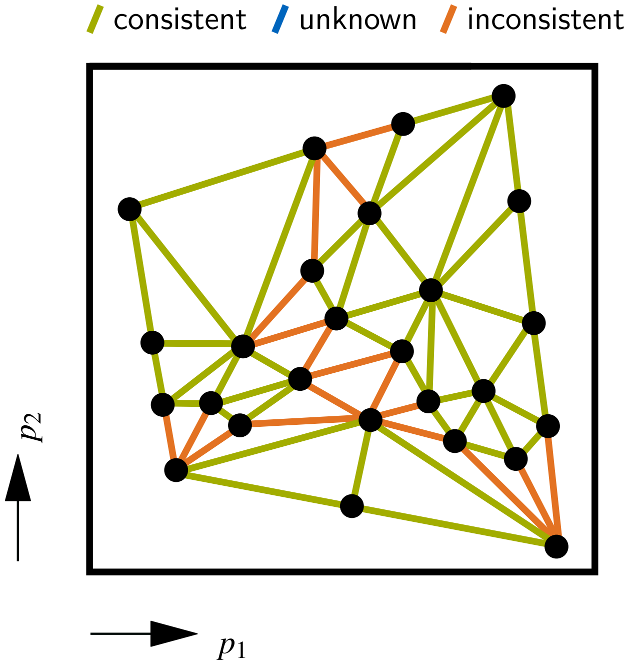
Adaptive Sampling and Clustering



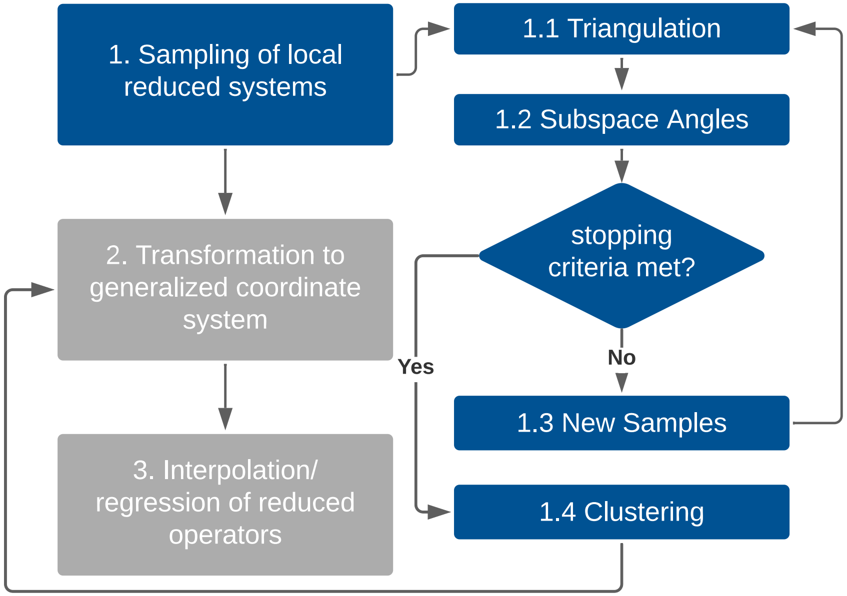


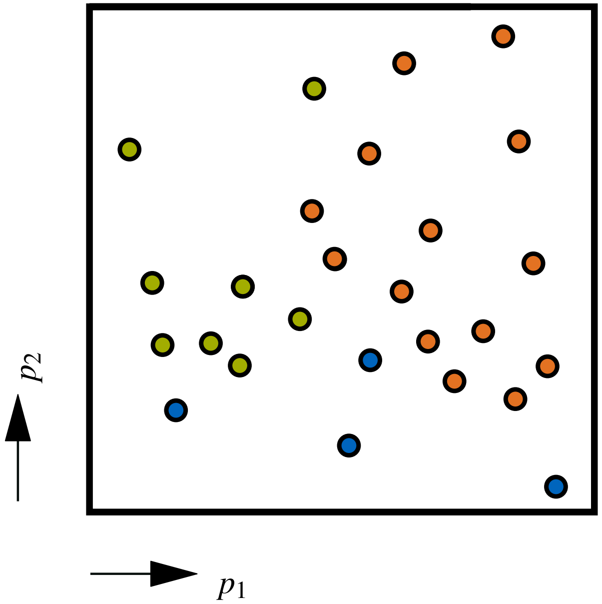
Adaptive Sampling and Clustering



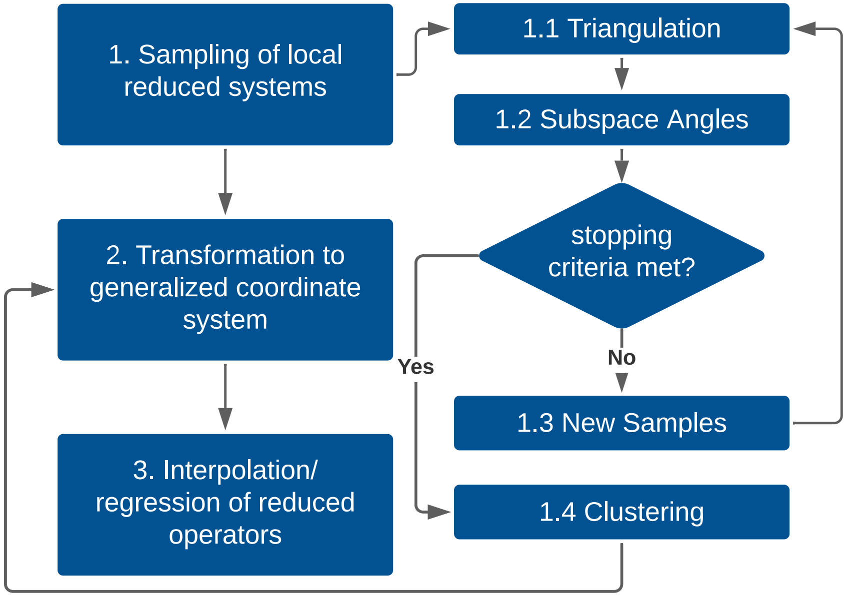


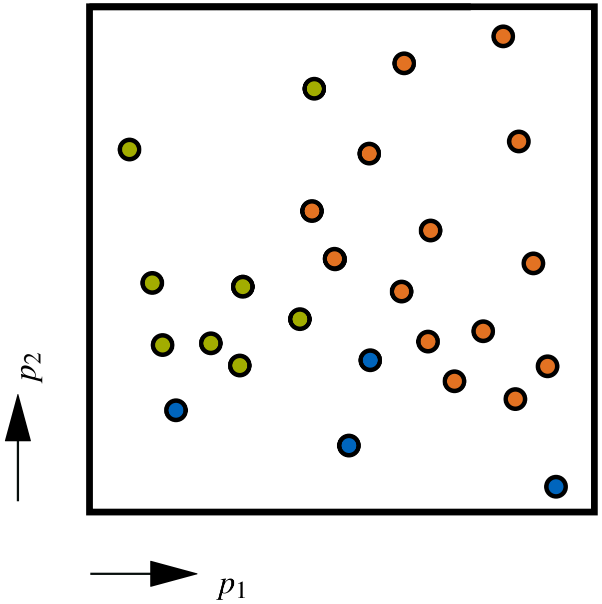
Adaptive Sampling and Clustering



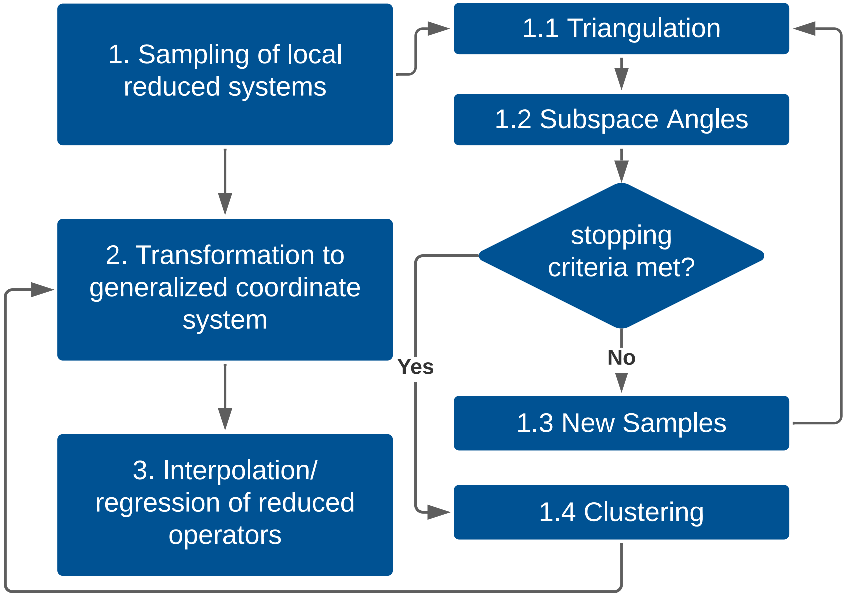


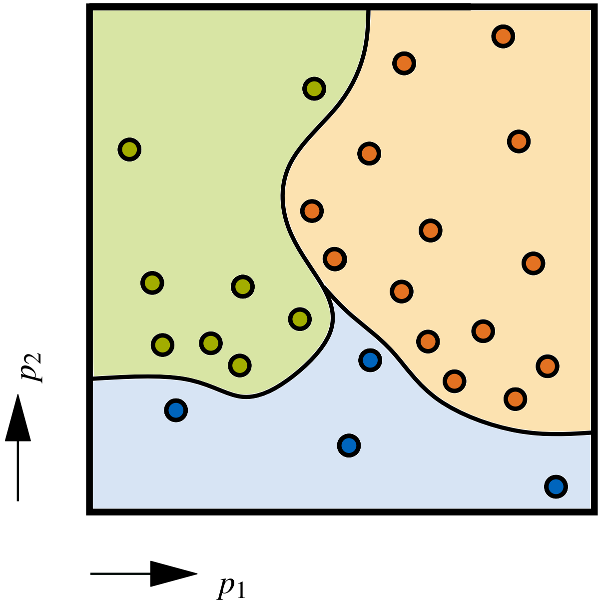
Adaptive Sampling and Clustering





Adaptive Sampling and Clustering

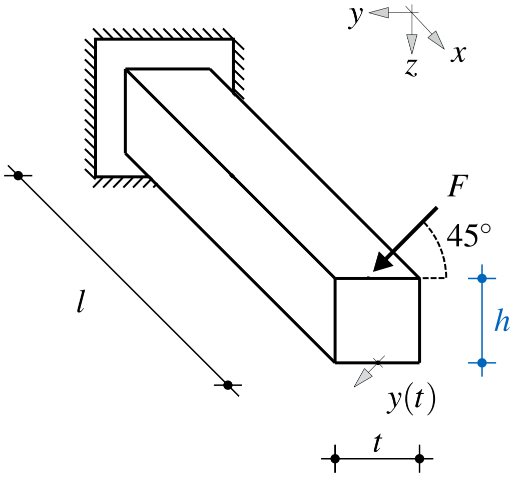




Results

Results – Timoshenko Beam – Beam Height h

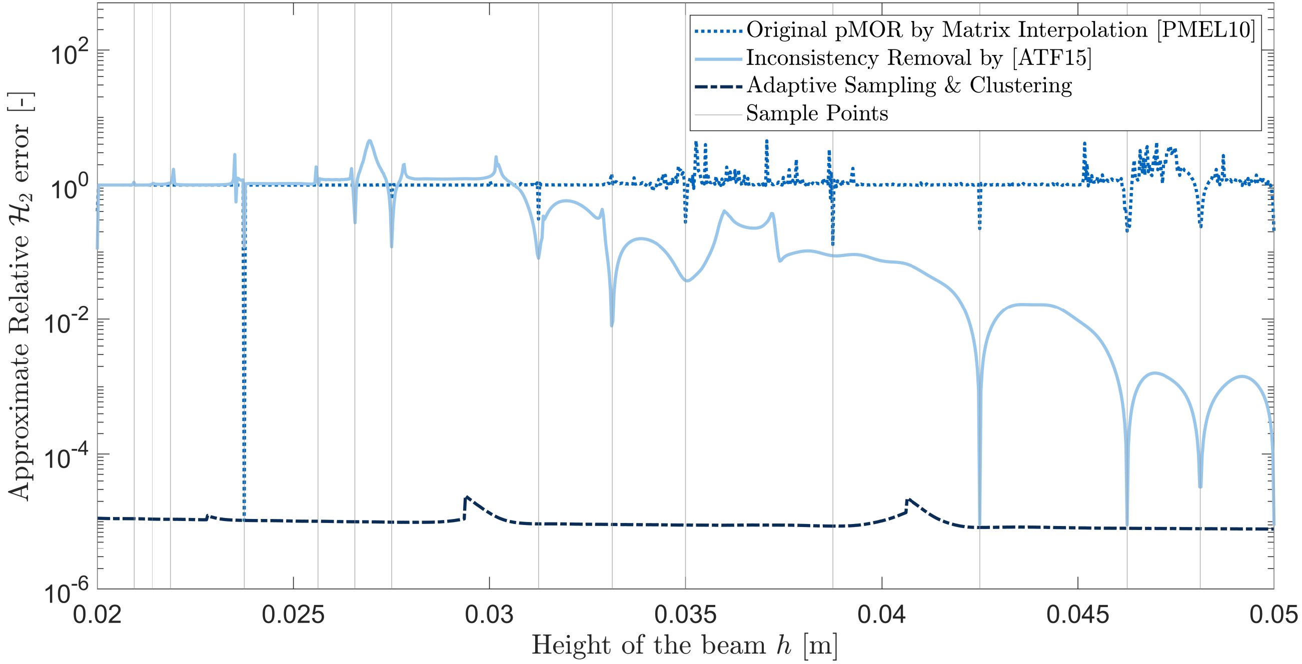
A 3D cantilevered beam discretized with Timoshenko beam elements is investigated. The beam is excited at the tip with a harmonic force of varying frequency ($[0, 1000]$ Hz). The adaptive sampling and clustering algorithm is compared to the original version of pMOR by Matrix interpolation [PMEL10] and a method for inconsistency removal by [ATF15].



Parameter	Range/Value	Unit
Height h	$[0.02, 0.05]$	m
Thickness t	0.01	m
Length l	1.0	m
Young's modulus E	$2.1 \cdot 10^{11}$	N/m^2
Poisson's ratio ν	0.3	-
Density ρ	7860	kg/m^3
Rayleigh damping α	$8 \cdot 10^{-6}$	$1/s$
Rayleigh damping β	8	s

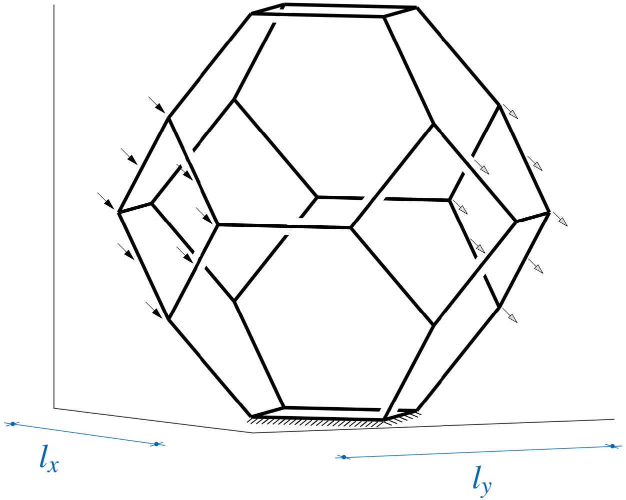
Table: Geometry and material parameters of the 3D cantilevered beam.

Results – Timoshenko Beam – Beam Height h



Results – Kelvin Cell – Dimensions l_x and l_y

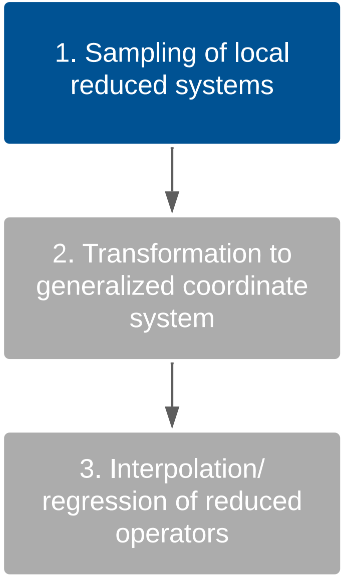
A 3D cantilevered beam discretized with Timoshenko beam elements is investigated. The beam is excited at the tip with a harmonic force of varying frequency ($[0, 1000]$ Hz). Rayleigh damping is used: $\mathbf{C} = \alpha\mathbf{K} + \beta\mathbf{M}$.

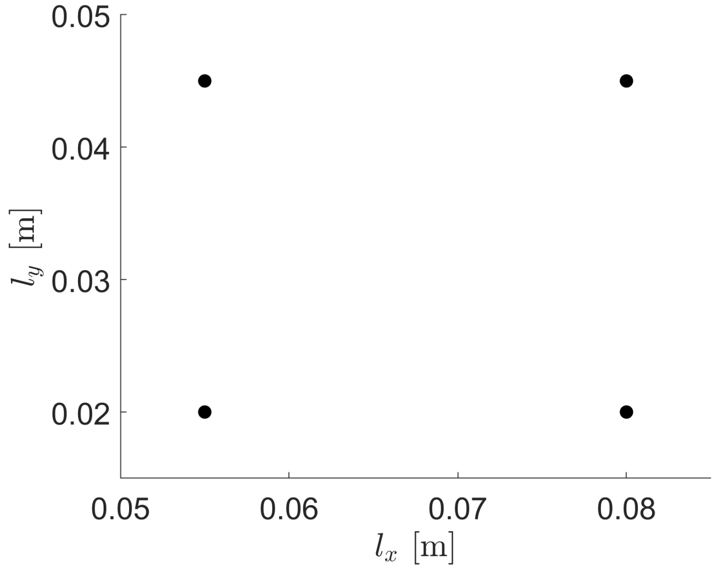


Parameter	Range/Value	Unit
Length l_x	$[0.055, 0.080]$	m
Length l_y	$[0.020, 0.045]$	m
Length l_z	0.05	m
Beam thickness t	0.001	m
Young's modulus E	$4.35 \cdot 10^9$	N/m^2
Poisson's ratio ν	0.3	-
Density ρ	1180	kg/m^3
Rayleigh damping α	$8 \cdot 10^{-6}$	$1/s$
Rayleigh damping β	8	s

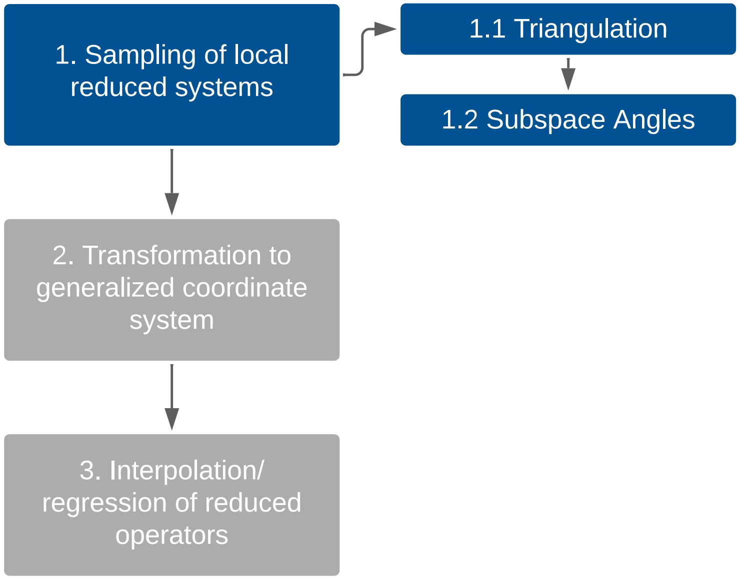
Table: Geometry and material parameters of the Kelvin Cell.

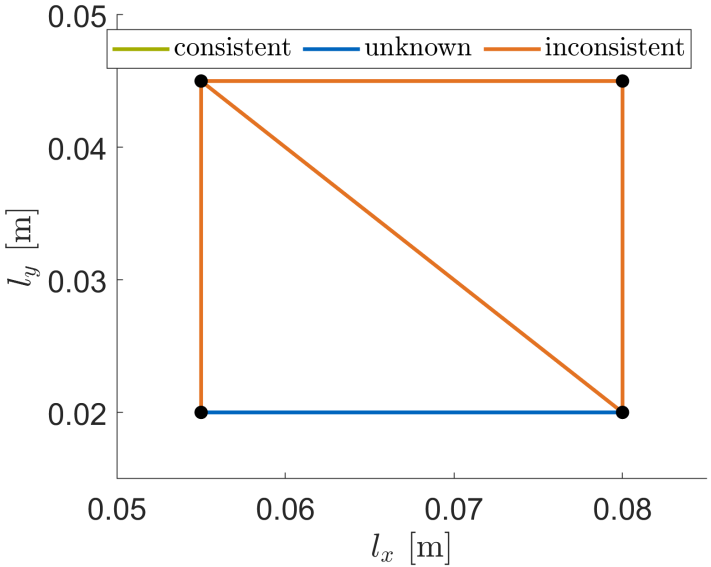
Results – Kelvin Cell – Adaptive Sampling & Classification



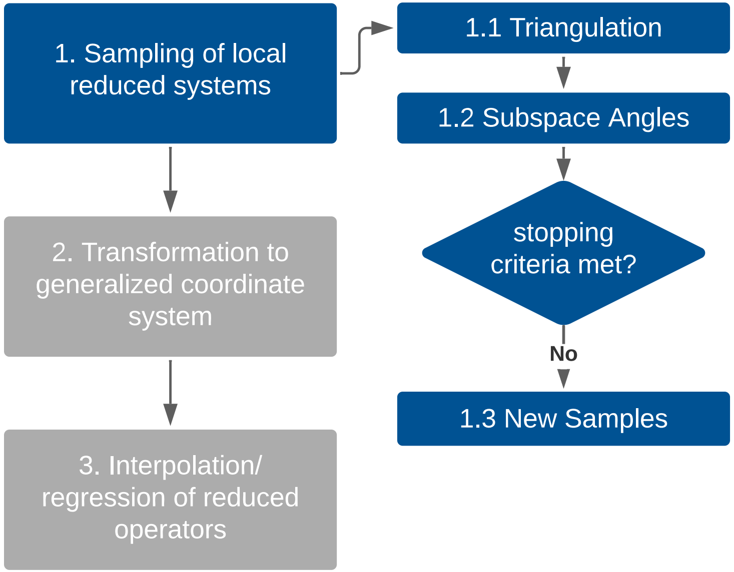


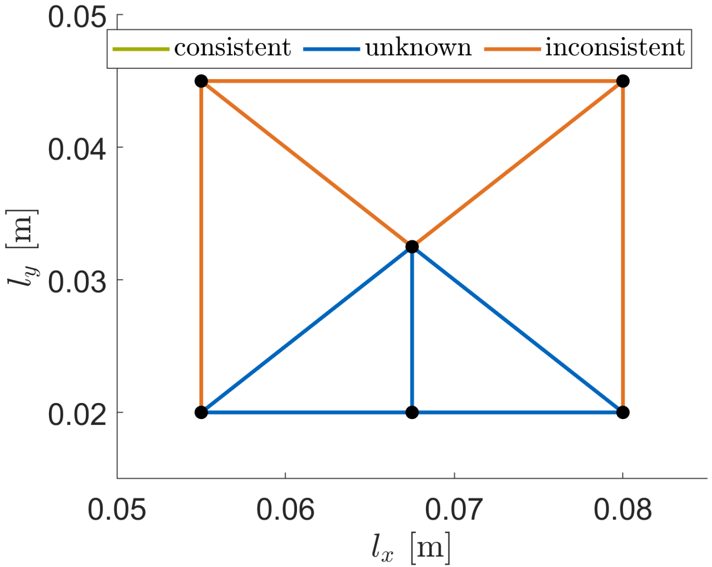
Results – Kelvin Cell – Adaptive Sampling & Classification



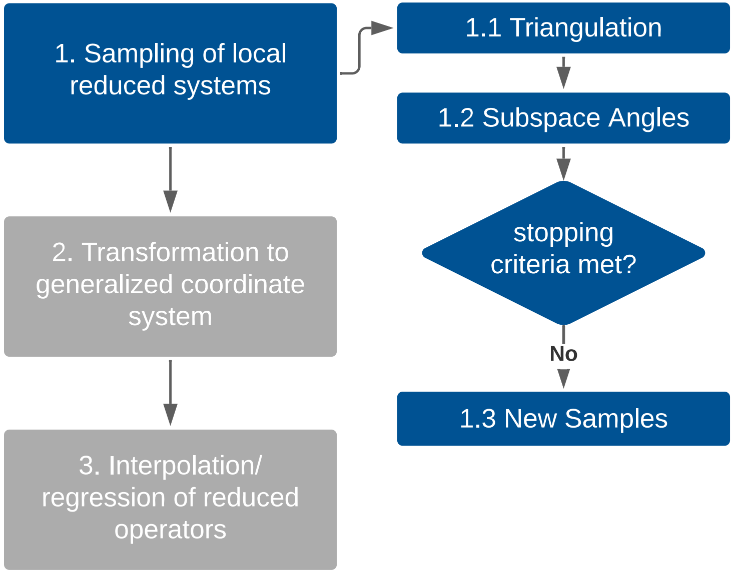


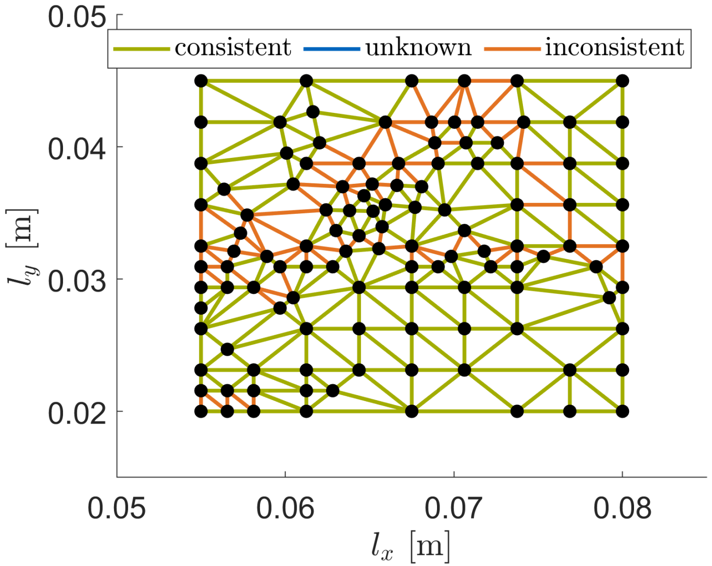
Results – Kelvin Cell – Adaptive Sampling & Classification



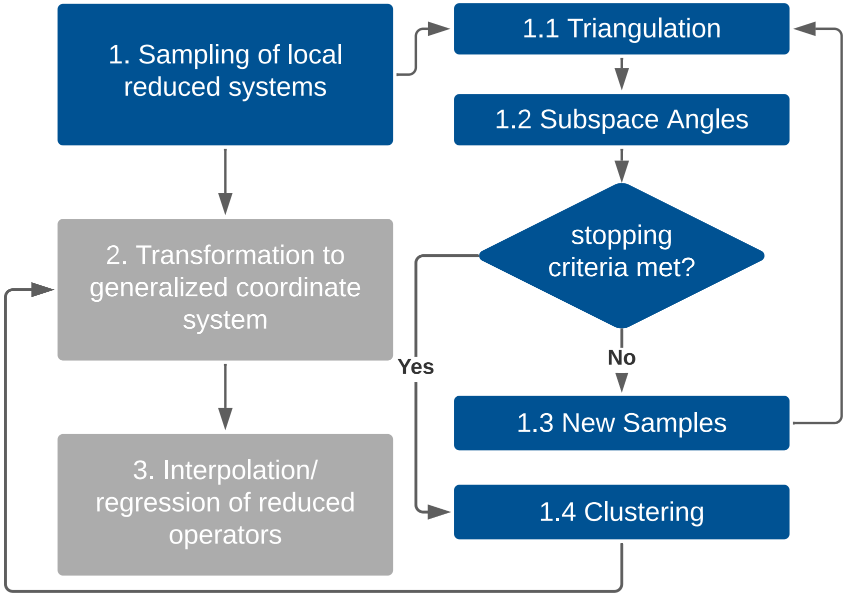


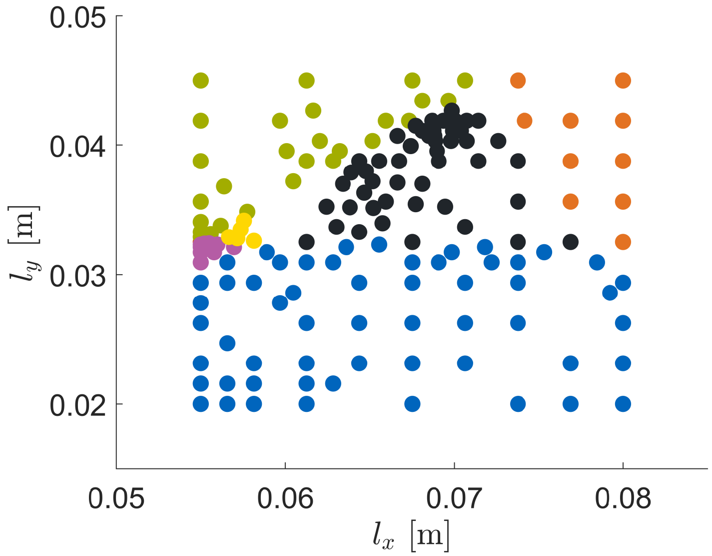
Results – Kelvin Cell – Adaptive Sampling & Classification



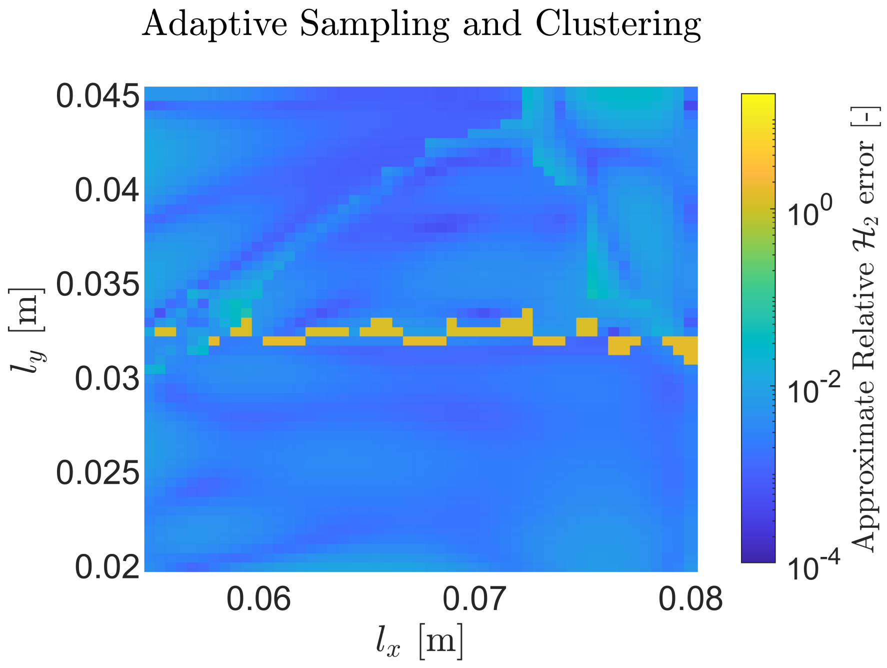


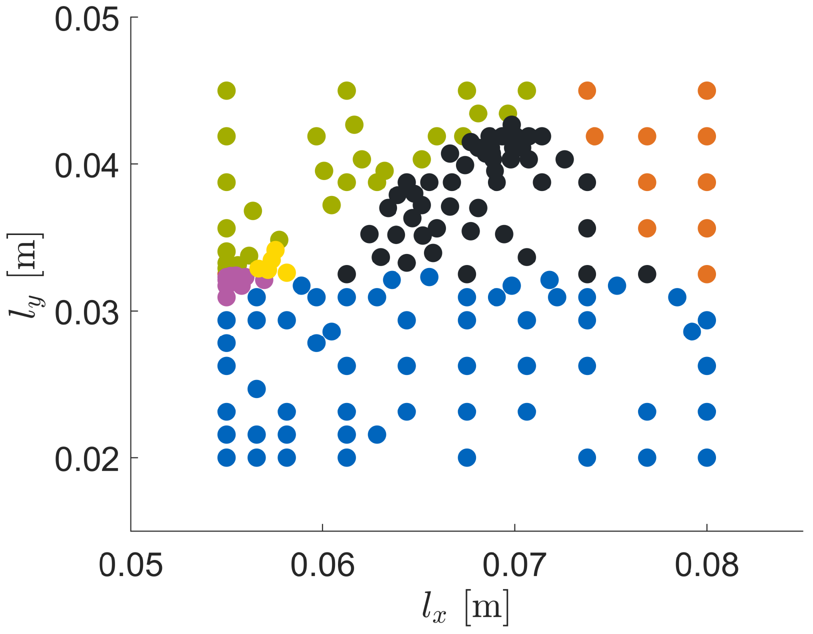
Results – Kelvin Cell – Adaptive Sampling & Classification



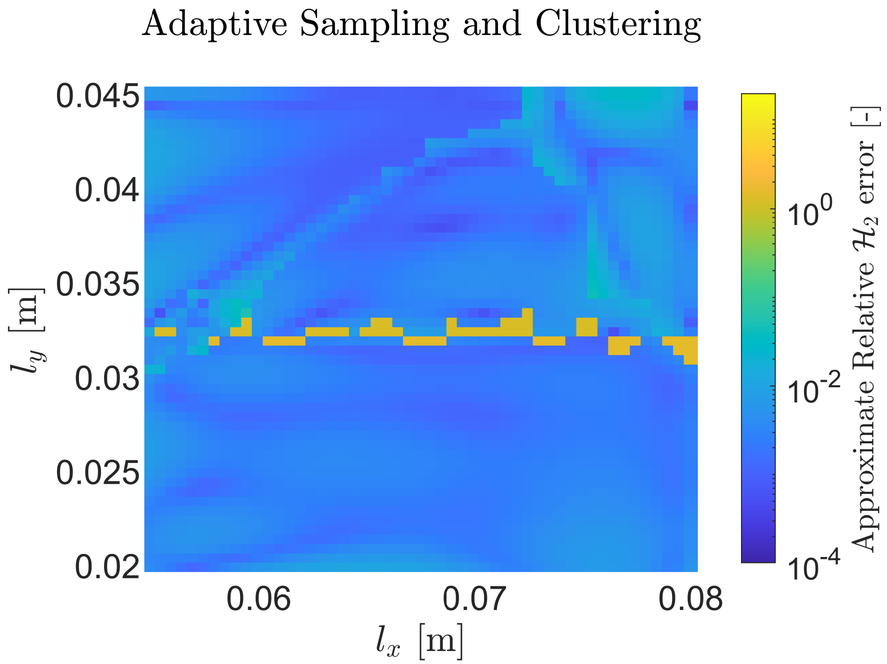


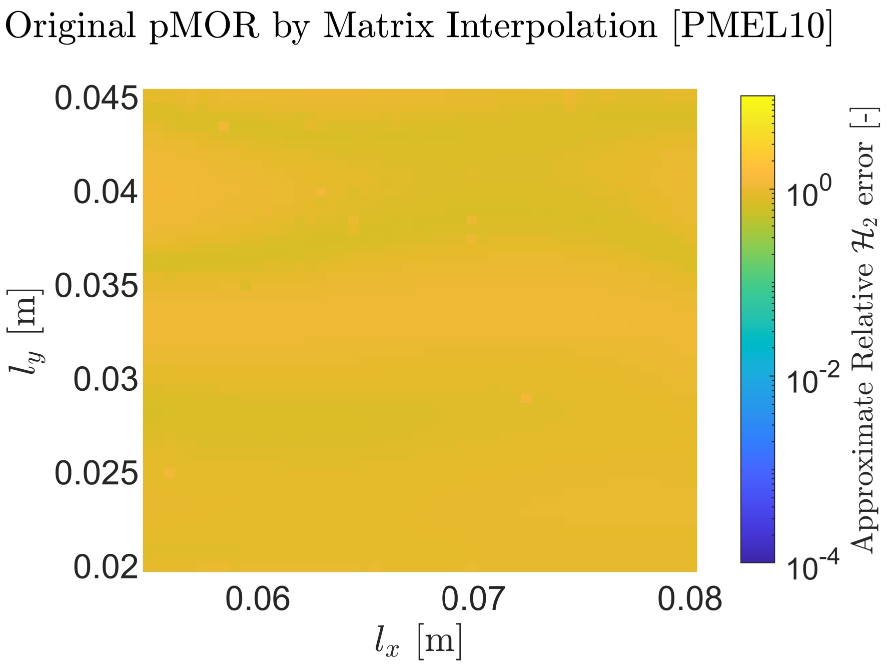
Results – Kelvin Cell – Dimensions l_x and l_y



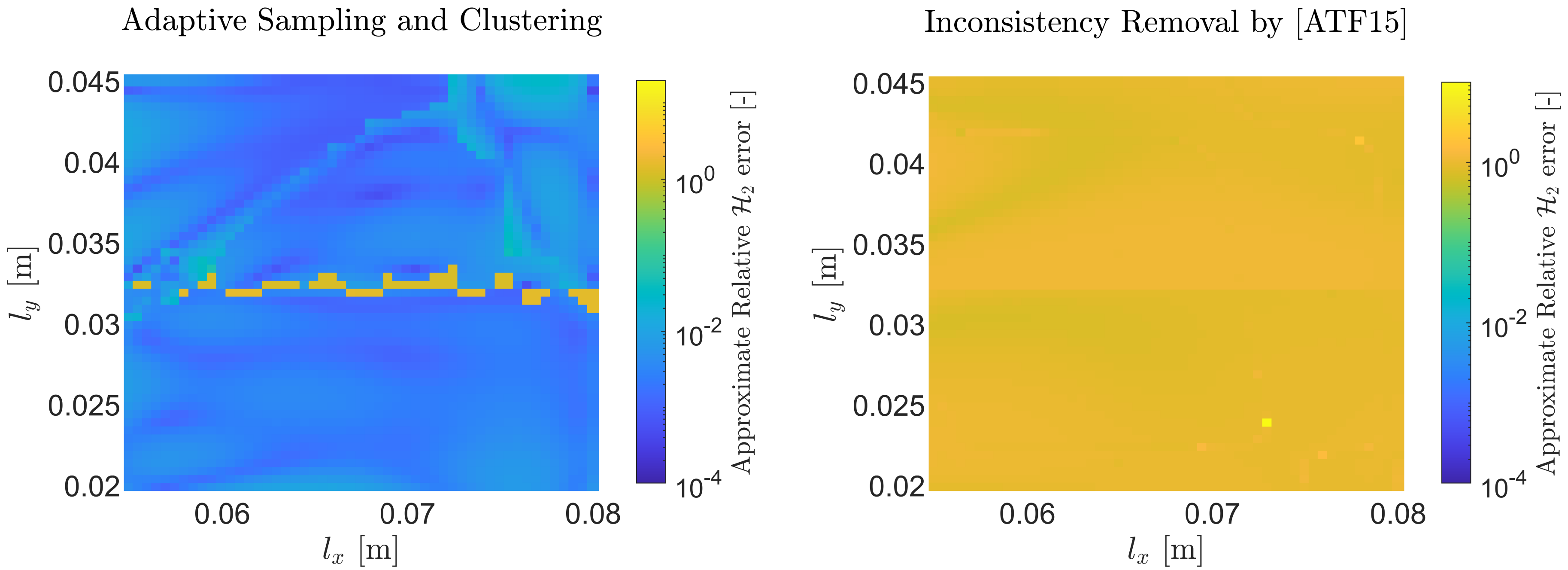


Results – Kelvin Cell – Dimensions l_x and l_y





Results – Kelvin Cell – Dimensions l_x and l_y



Conclusion and Future Work

Conclusion & Future Work

Our objective was to generate a parametric reduced-order model (pROM) that allows to solve the transfer function

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{14}$$

efficiently and

- does not require an affine representation of the parametric dependency,
- is valid for a large range of the parameters
- is generated via an adaptive algorithm.

Conclusion & Future Work

Our objective was to generate a parametric reduced-order model (pROM) that allows to solve the transfer function

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{14}$$

efficiently and

- does not require an affine representation of the parametric dependency,

→ Parametric Model Order Reduction by Matrix Interpolation

- is valid for a large range of the parameters

- is generated via an adaptive algorithm.

Conclusion & Future Work

Our objective was to generate a parametric reduced-order model (pROM) that allows to solve the transfer function

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p})\left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1}\mathbf{f}(\mathbf{p}), \tag{14}$$

efficiently and

- does not require an affine representation of the parametric dependency,

→ Parametric Model Order Reduction by Matrix Interpolation

- is valid for a large range of the parameters

→ Partitioning of the parameter space, generation of several local pROMs

- is generated via an adaptive algorithm.

Conclusion & Future Work

Our objective was to generate a parametric reduced-order model (pROM) that allows to solve the transfer function

$$H(s,\mathbf{p}) = \mathbf{g}(\mathbf{p}) \left(s^2\mathbf{M}(\mathbf{p}) + s\mathbf{C}(\mathbf{p}) + \mathbf{K}(\mathbf{p})\right)^{-1} \mathbf{f}(\mathbf{p}), \tag{14}$$

efficiently and

- does not require an affine representation of the parametric dependency,

$\rightarrow$ Parametric Model Order Reduction by Matrix Interpolation

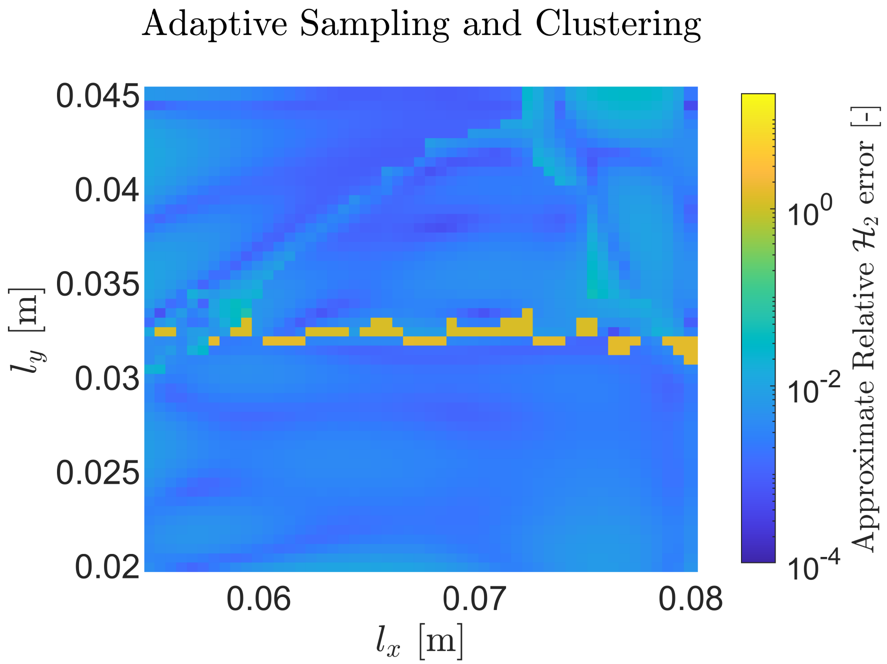
- is valid for a large range of the parameters

$\rightarrow$ Partitioning of the parameter space, generation of several local pROMs

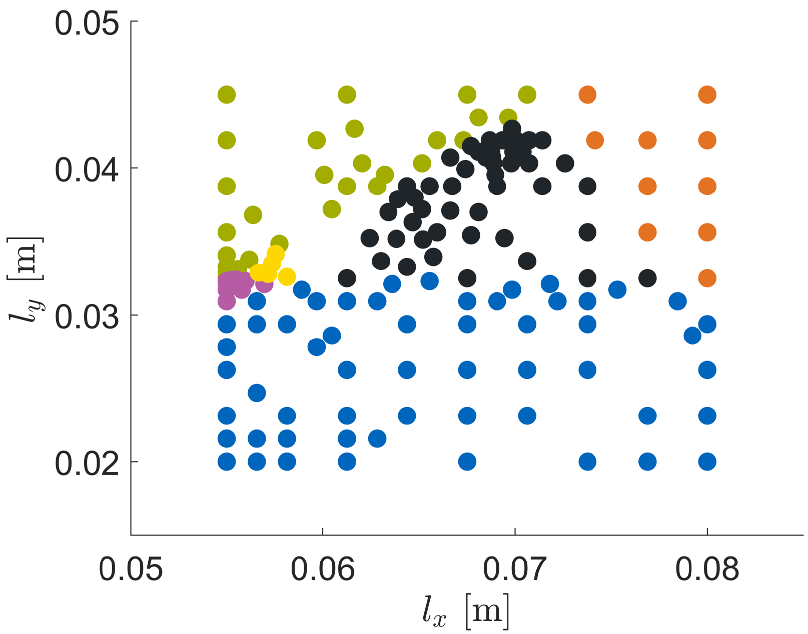
- is generated via an adaptive algorithm.

$\rightarrow$ Adaptive sampling

Future Work



Adaptive Sampling and Clustering
0.045
0.04
0.035
0.03
0.025
0.02
l_y [m]
0.06
0.07
0.08
l_x [m]
10^0
10^{-2}
10^{-4}
Approximate Relative $\mathcal{H}_2$ error [-]



0.05
0.04
0.03
0.02
l_y [m]
0.05
0.06
0.07
0.08
l_x [m]

References

[ATF15] David Amsallem, Radek Tezaur, and Charbel Farhat. Real-time solution of computational problems using databases of parametric linear reduced-order models with arbitrary underlying meshes. *Journal of Computational Physics*, 326, 06 2015.

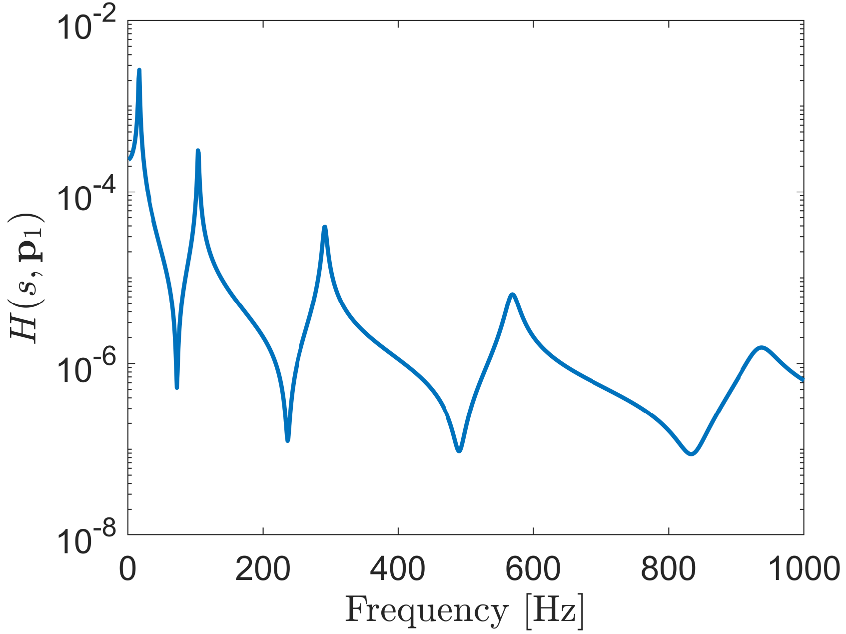
[BGW15] Peter Benner, Serkan Gugercin, and Karen Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, jan 2015.

[BNN+15] M. A. Bazaz, S. A. Nahve, M. Nabi, S. Janardhanan, and M. U. Rehman. Adaptive parameter space sampling in matrix interpolatory pmor. In *2015 International Conference on Recent Developments in Control, Automation and Power Engineering (RDCAPE)*, pages 83–89, March 2015.

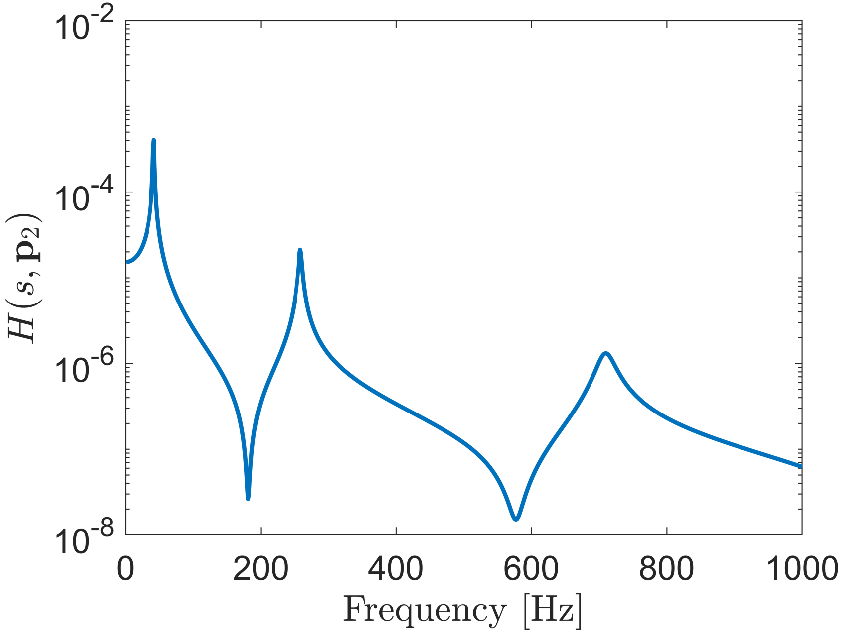
[FE15] Michael Fischer and Peter Eberhard. Application of parametric model reduction with matrix interpolation for simulation of moving loads in elastic multibody systems. *Advances in Computational Mathematics*, 41(5):1049–1072, October 2015.

[PMEL10] Heiko Peuscher, Jan Mohring, Rudy Eid, and Boris Lohmann. Parametric model order reduction by matrix interpolation. *Automatisierungstechnik*, 58:475–484, 08 2010.

MOR Method

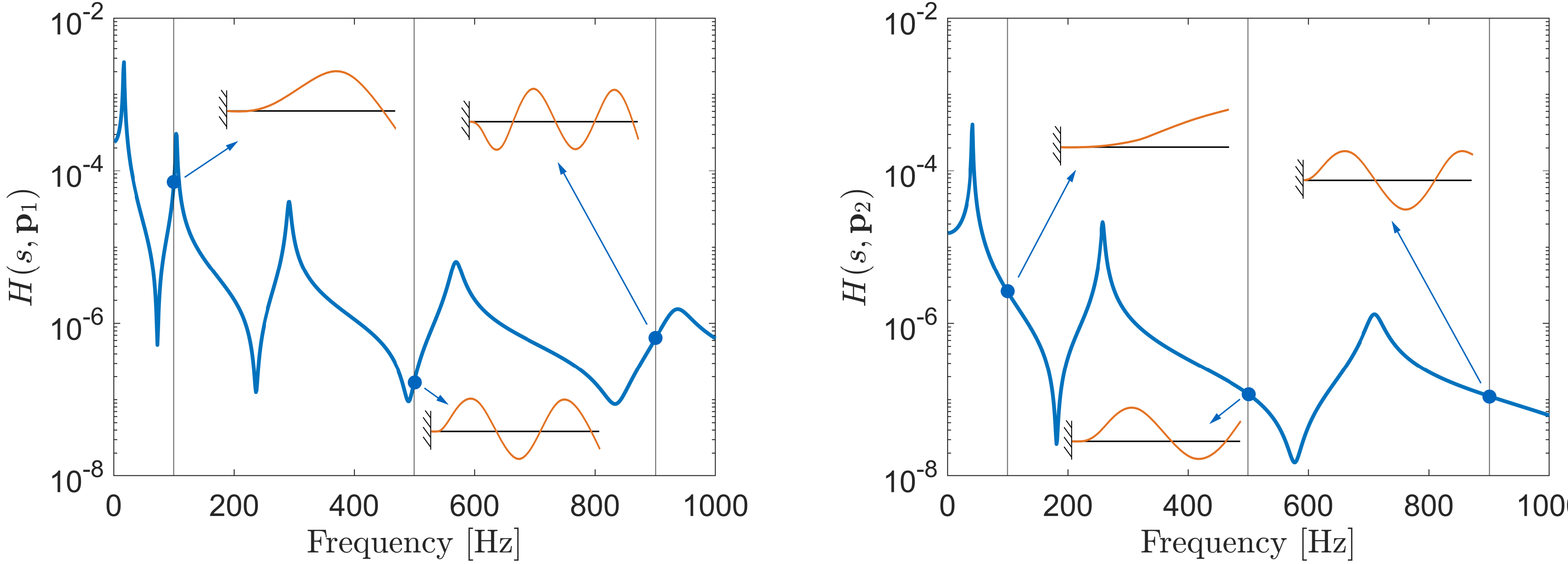


10^{-2}
10^{-4}
10^{-6}
10^{-8}
0
200
400
600
800
1000
$H(s, \mathbf{p}_1)$
Frequency [Hz]

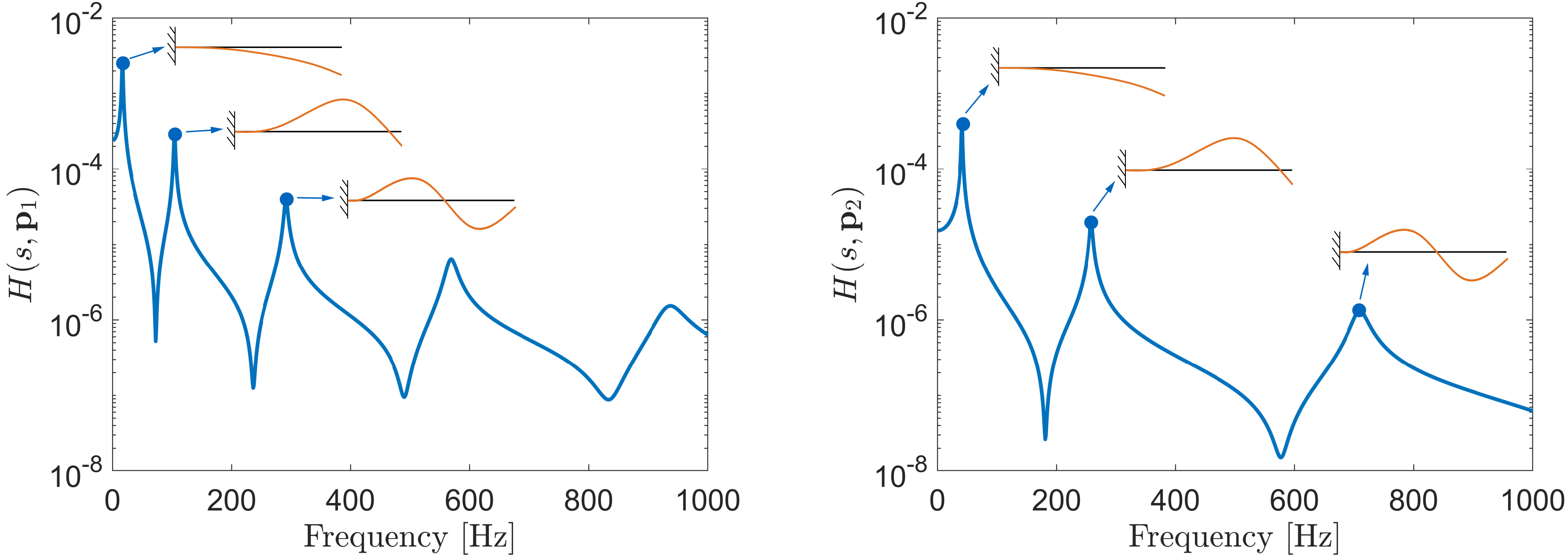


10^{-2}
10^{-4}
10^{-6}
10^{-8}
0
200
400
600
800
1000
$H(s, \mathbf{p}_2)$
Frequency [Hz]

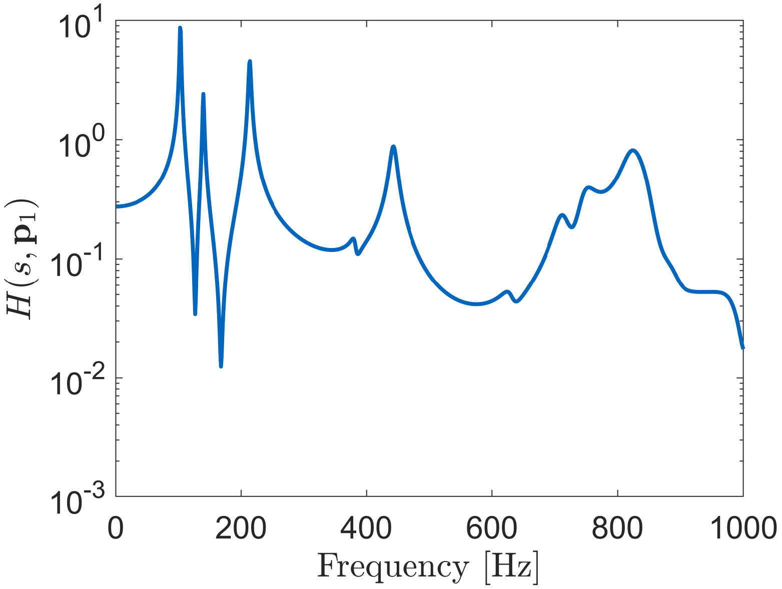
MOR Method – Proper Orthogonal Decomposition

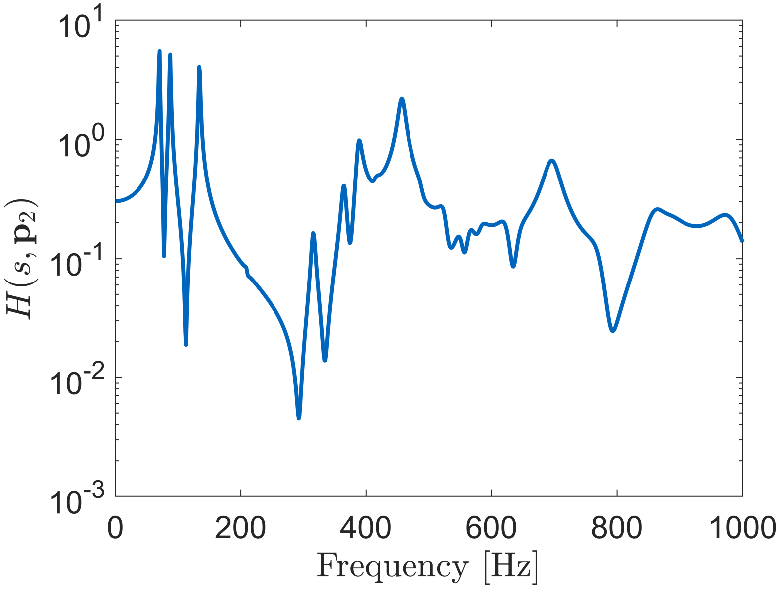


MOR Method – Modal Truncation

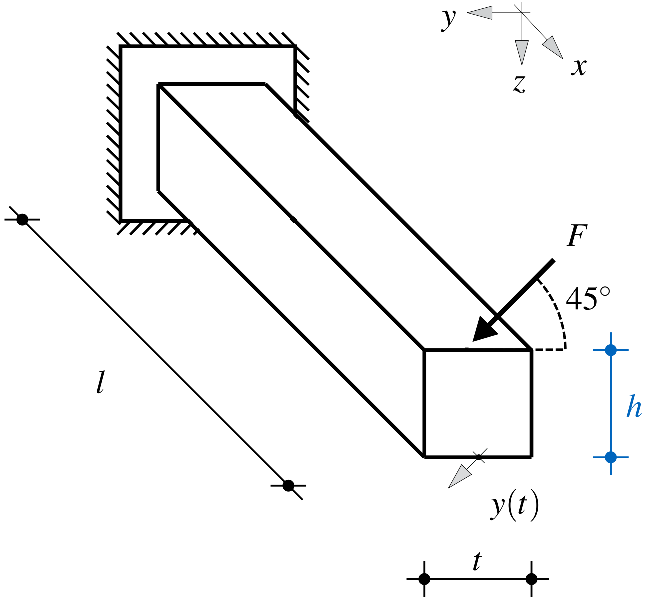


Change of System Dynamics

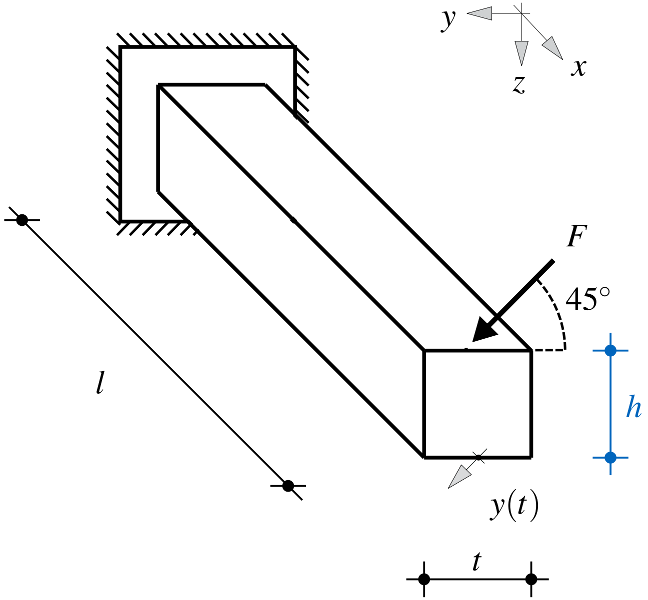


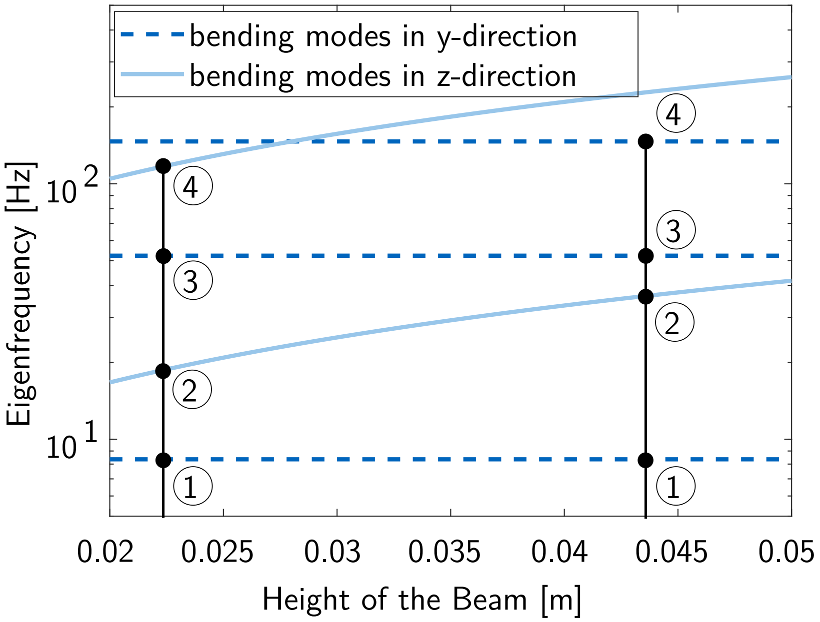


Mode Switching and Truncation

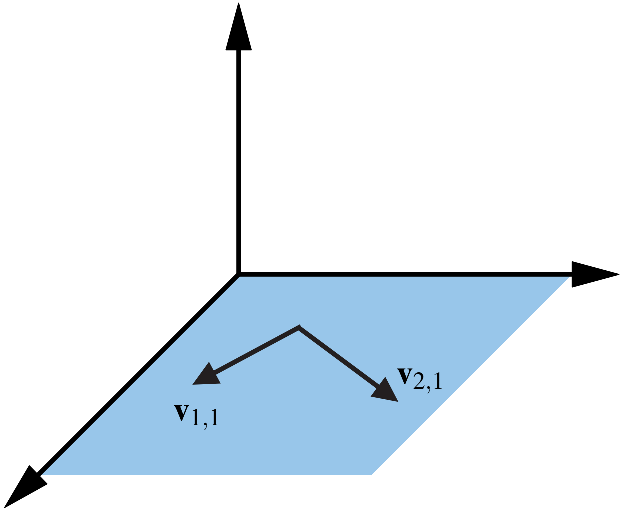


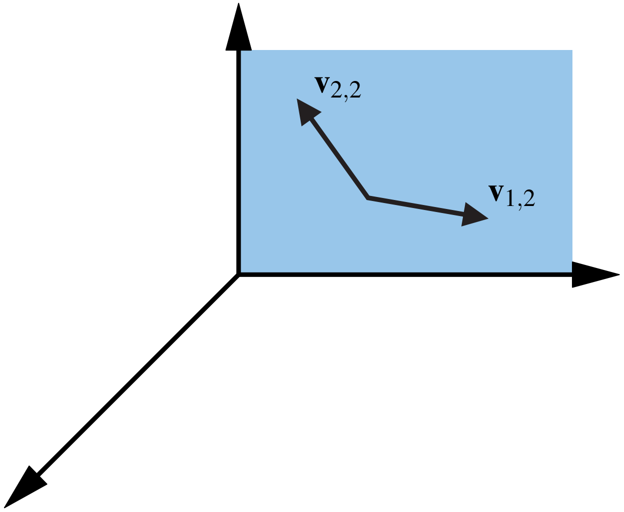
Mode Switching and Truncation



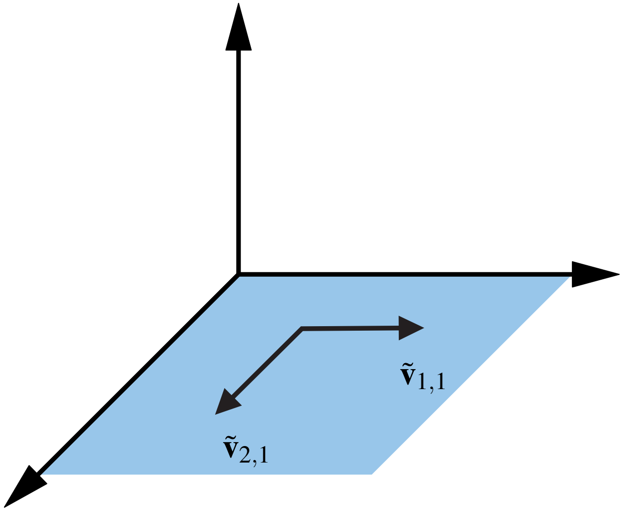


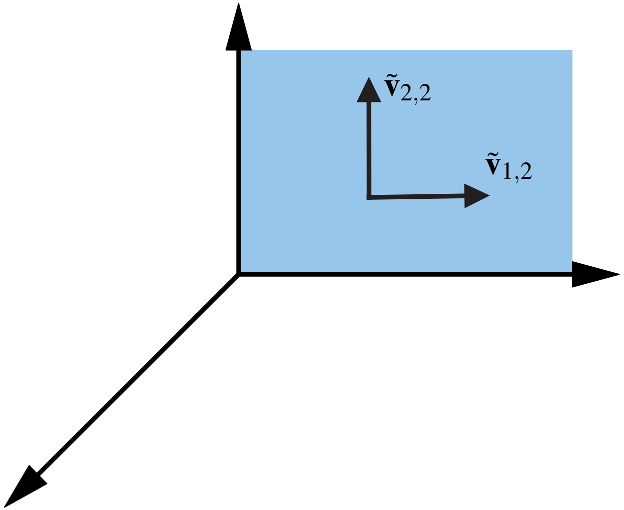
Mode Switching and Truncation



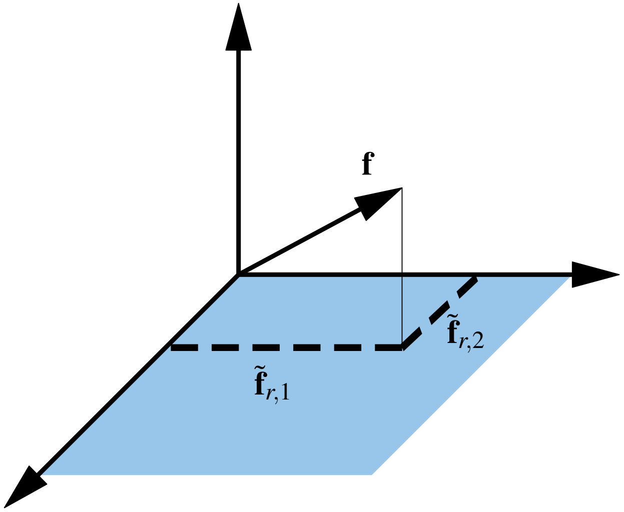


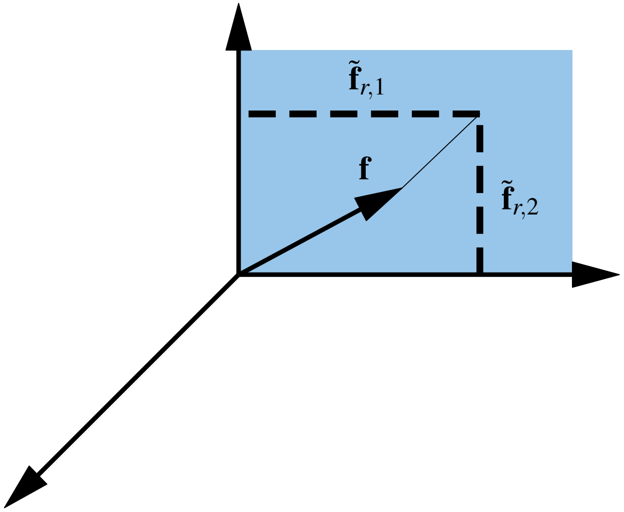
Mode Switching and Truncation



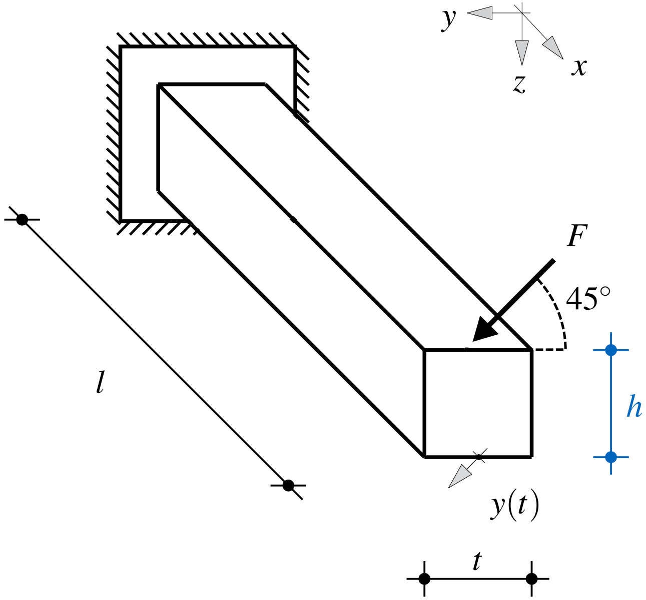


Mode Switching and Truncation

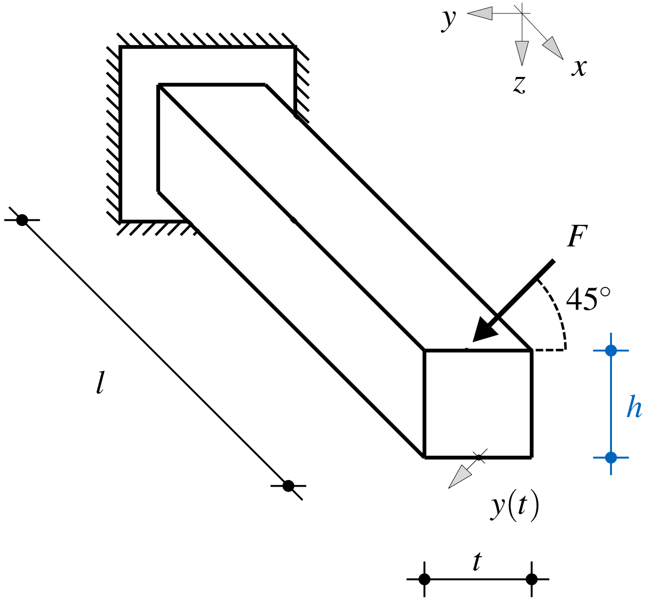


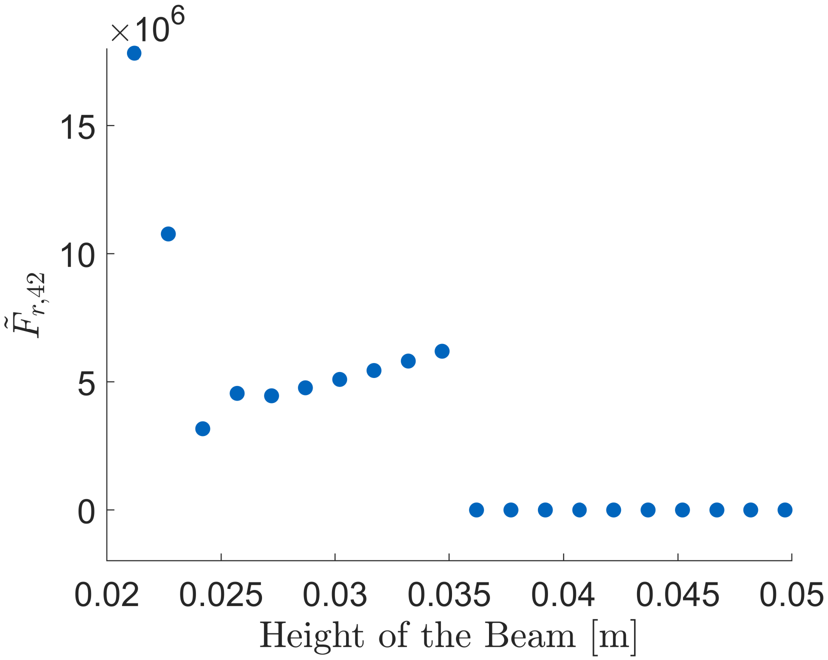


Mode Switching and Truncation

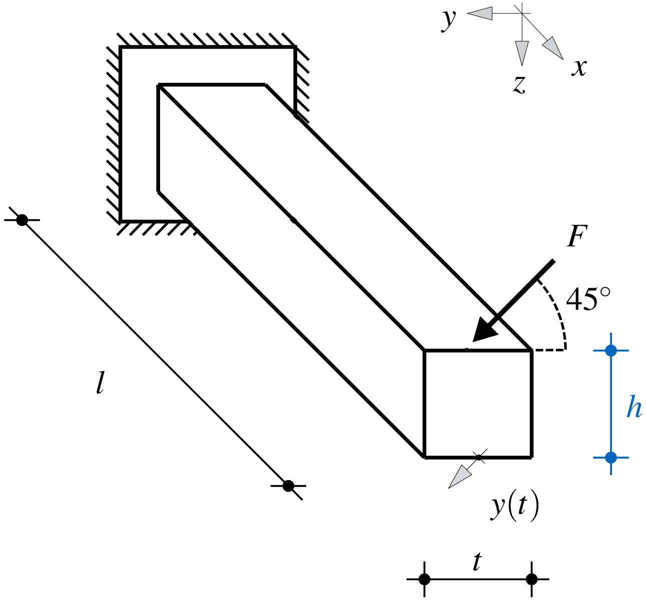


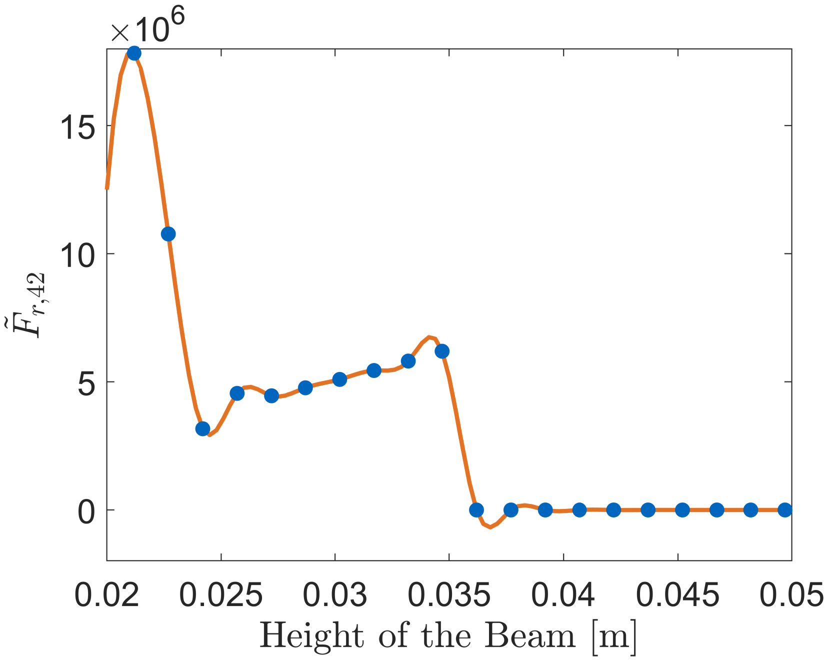
Mode Switching and Truncation



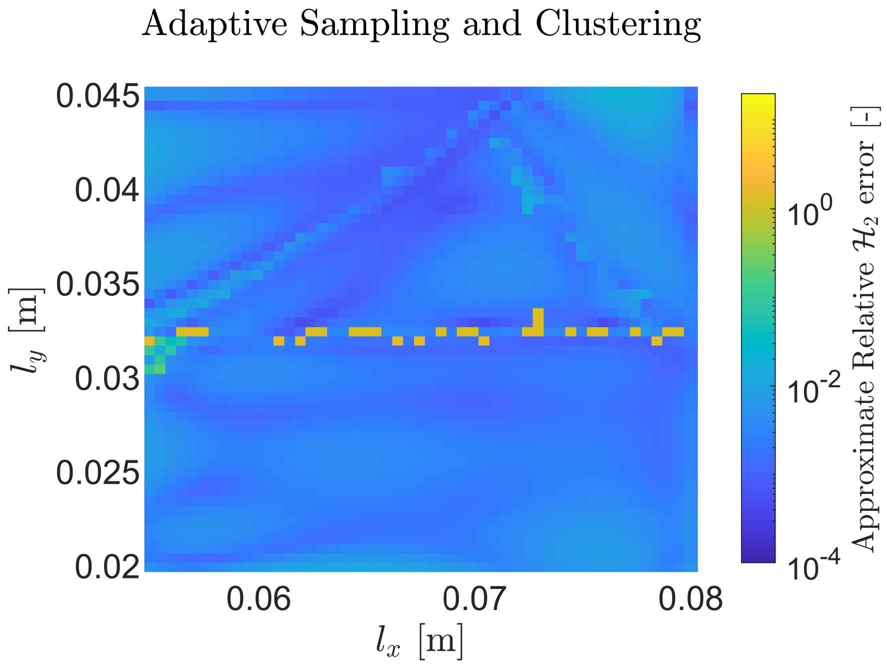


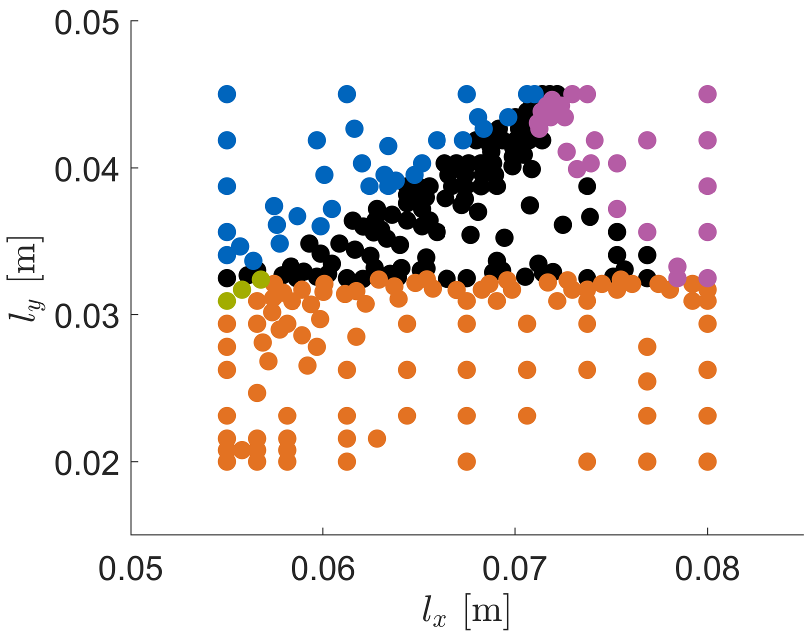
Mode Switching and Truncation



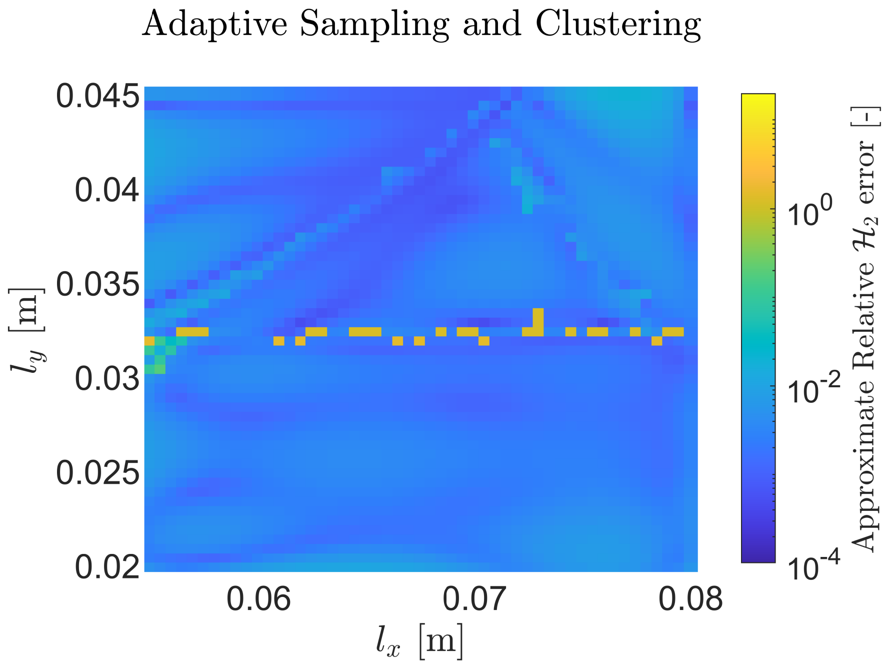


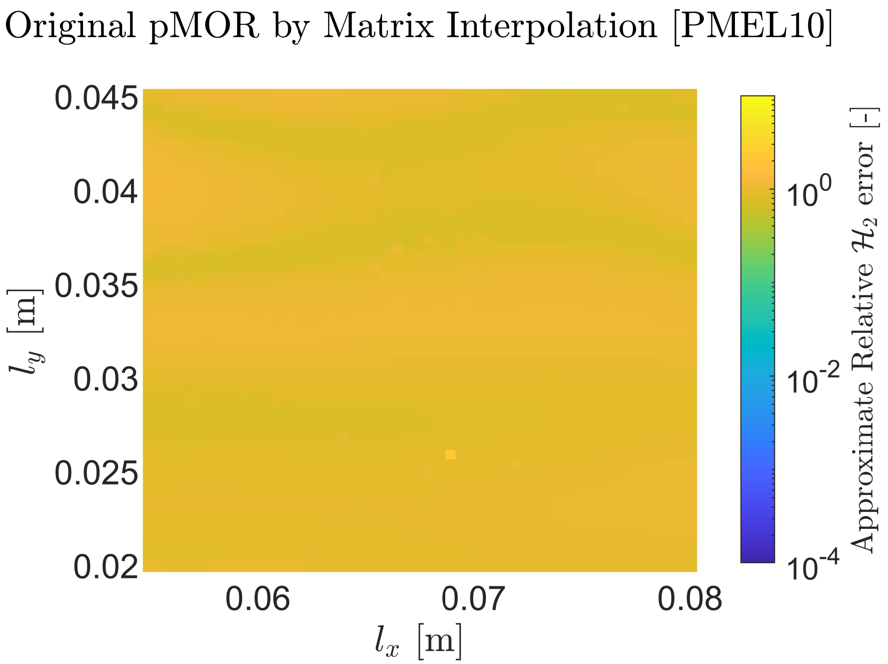
Results – Kelvin Cell – Dimensions l_x and l_y



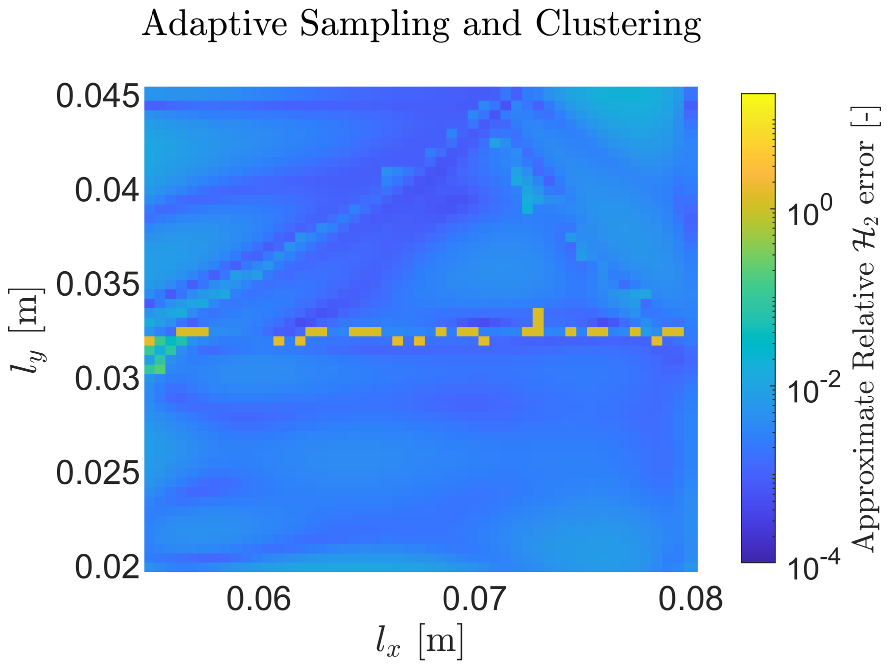


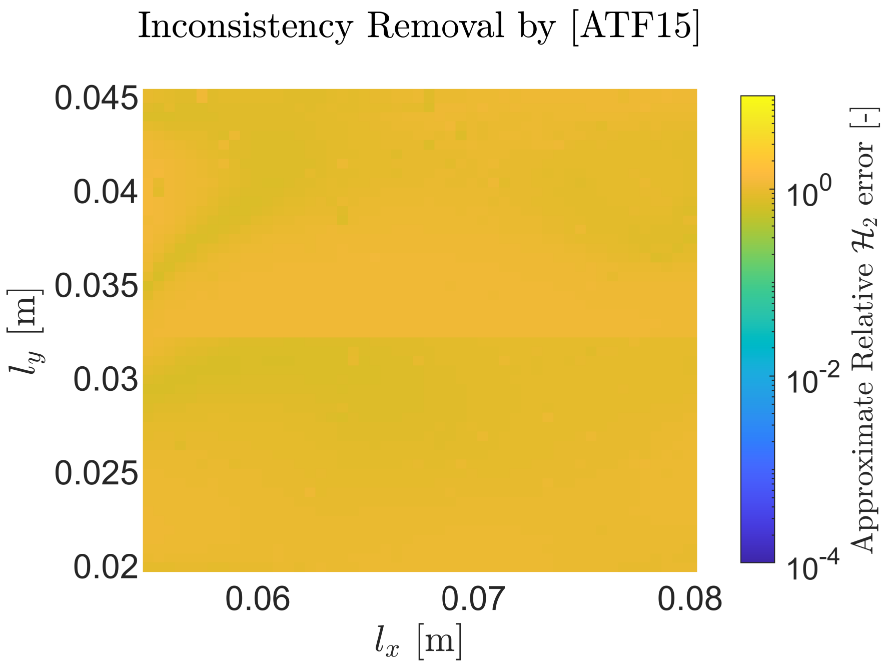
Results – Kelvin Cell – Dimensions l_x and l_y



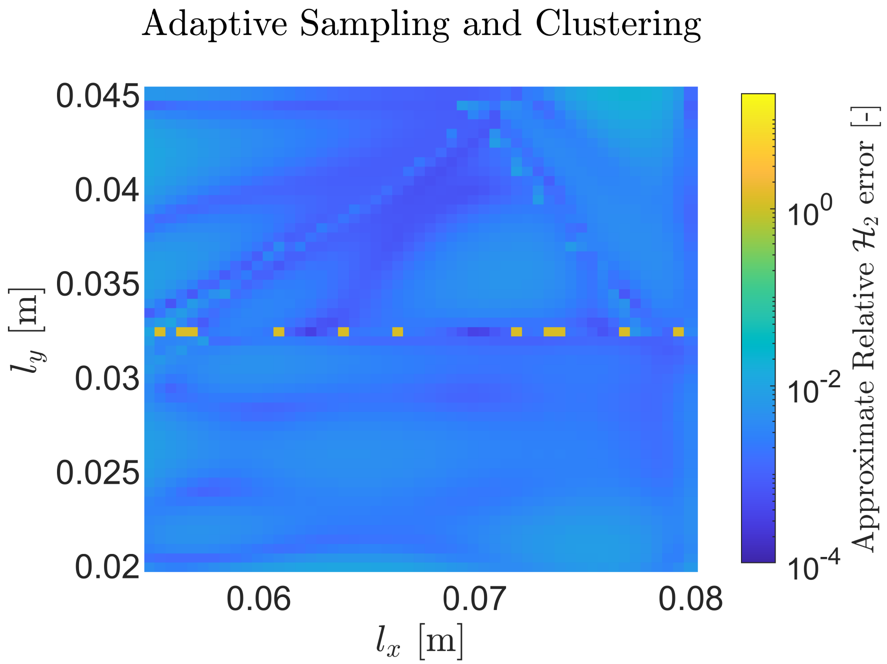


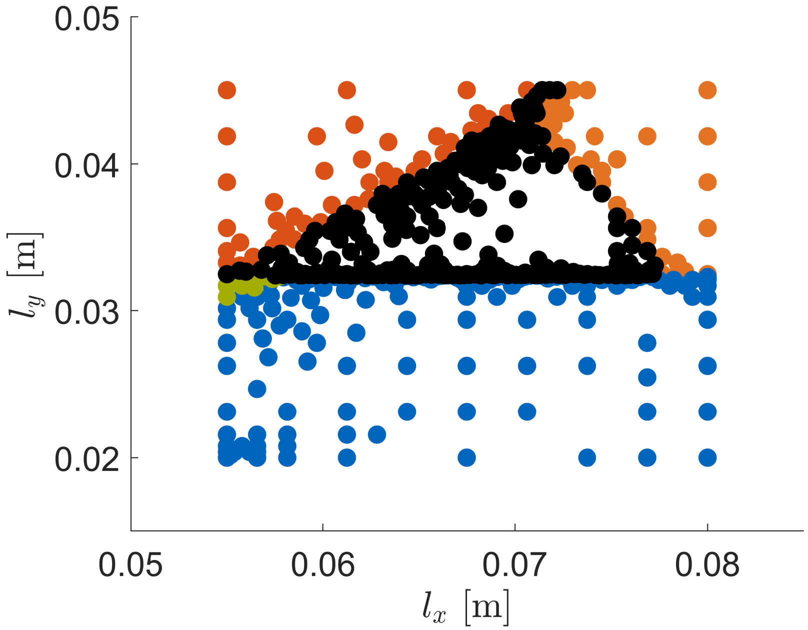
Results – Kelvin Cell – Dimensions l_x and l_y





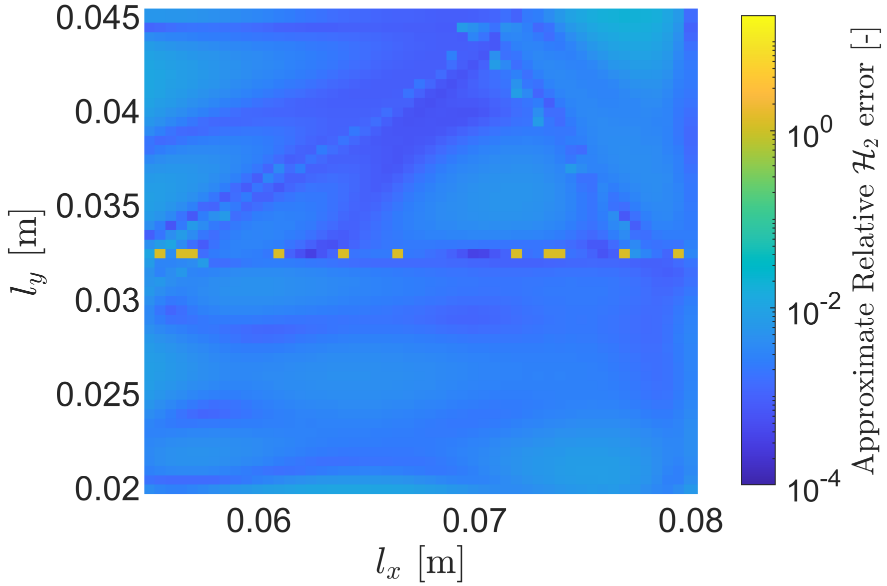
Results – Kelvin Cell – Dimensions l_x and l_y



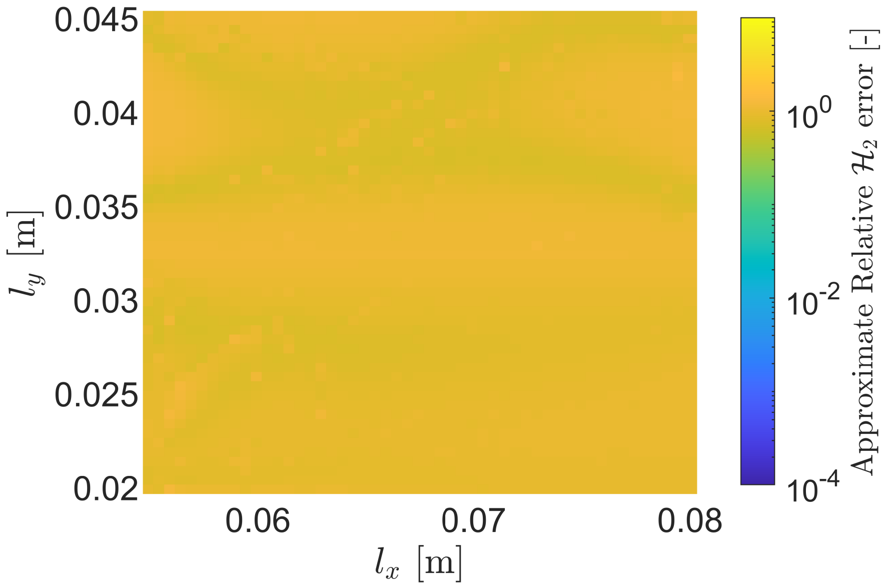


Results – Kelvin Cell – Dimensions l_x and l_y

Adaptive Sampling and Clustering



Original pMOR by Matrix Interpolation [PMEL10]



Results – Kelvin Cell – Dimensions l_x and l_y