Adaptive Isogeometric Methods and Reduced Order Modeling

T. Kvamsdal[1,2] E. H. van Brummelen[3] E. Fonn[2]
K. Johannessen[2] A. M. Kvarving[2] A. Rasheed[2]

[1]Department of Mathematical Sciences, NTNU

[2]Applied Mathematics and Cybernetics, SINTEF Digital

[3]Department of Mechanical Engineering, TU/e

Contents

- Basics of reduced basis methods
- Reduced Order Models (ROM) for Navier-Stokes
- First step towards certified ROM

Basics of Reduced Basis Methods

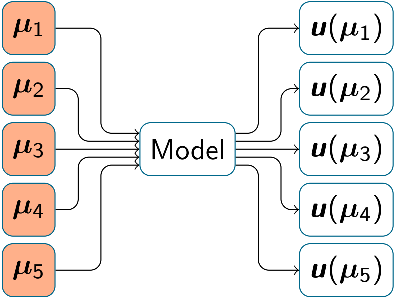
Parameter-dependent models

$$\boxed{\boldsymbol{\mu}} \longrightarrow \boxed{\text{Model}} \longrightarrow \boxed{\boldsymbol{u}(\boldsymbol{\mu})}$$

- We are interested generating solutions $\boldsymbol{u}(\boldsymbol{\mu})$ to a physical model that depend on a set of pre-determined *parameters*, $\boldsymbol{\mu} \in \mathcal{P}$.
- Parameters can be: viscosity, heat conductivity, varying boundary conditions, geometry changes, etc.

Parameter-dependent models

Motivation: *many-query* applications. E.g. control systems, optimization, inverse problems and real-time responsiveness.



Dimensional reduction

- With conventional (read: FEM, FVM, FDM, and yes, even IGA) methods, this may be impractical if not impossible.
- Too many DoFs N to finish in a realistic timeframe.
- Usually,

$$M = \dim \text{“span”} \left(\{ \boldsymbol{u}(\boldsymbol{\mu}) \,|\, \boldsymbol{\mu} \in \mathcal{P} \} \right) \ll N$$

- Idea: create a model with number of DoFs closely matching the physical dimension of the problem.
- Often, $M \sim 100$ or so!

The vision[1]

Parametrized PDE

High-fidelity discr.

$$\boldsymbol{A}_h(\boldsymbol{\mu})\boldsymbol{u}_h(\boldsymbol{\mu}) = \boldsymbol{f}_h(\boldsymbol{\mu})$$

$$\boldsymbol{A}_h(\boldsymbol{\mu}) = \sum_i \xi_i(\boldsymbol{\mu})\boldsymbol{A}_i$$

$$\boldsymbol{f}_h(\boldsymbol{\mu}) = \sum_i \chi_i(\boldsymbol{\mu})\boldsymbol{f}_i$$

Generate a RB by applying POD to snapshot solutions, and construct projection matrix $\boldsymbol{V}$

Projection

$$\boldsymbol{A}_i^M = \boldsymbol{V}^\intercal \boldsymbol{A}_i \boldsymbol{V}$$

$$\boldsymbol{f}_i^M = \boldsymbol{V}^\intercal \boldsymbol{f}_i$$

$\boldsymbol{\mu}$

RB system assembly

$$\boldsymbol{A}_M = \sum_i \xi_i(\boldsymbol{\mu})\boldsymbol{A}_i^M$$

$$\boldsymbol{f}_M = \sum_i \chi_i(\boldsymbol{\mu})\boldsymbol{f}_i^M$$

RB system solution

$$\boldsymbol{A}_M(\boldsymbol{\mu})\boldsymbol{u}_M(\boldsymbol{\mu}) = \boldsymbol{f}_M(\boldsymbol{\mu})$$

Error estimate

$$\|\boldsymbol{u}_h(\boldsymbol{\mu}) - \boldsymbol{V}\boldsymbol{u}_M(\boldsymbol{\mu})\|$$

Recovery, visualization and post-processing

$$\boldsymbol{u}_M^h(\boldsymbol{\mu}) = \boldsymbol{V}\boldsymbol{u}_M(\boldsymbol{\mu})$$

Evaluating functionals of interest

[1] See Quarteroni, Manzoni, and Negri 2016.

Our guiding principle

"**Any**" extra cost in the offline stage is worth paying,
no matter how much, if it makes the online stage faster.

Our guiding principle

All is fair in love, war *and the offline stage*.
— John Lyly (*Euphues*; 1579)

Assembly

- Why the insistence on forms like

$$\boldsymbol{A}_h(\boldsymbol{\mu}) = \sum_i \xi_i(\boldsymbol{\mu})\boldsymbol{A}_i, \qquad \boldsymbol{f}_h(\boldsymbol{\mu}) = \sum_i \chi_i(\boldsymbol{\mu})\boldsymbol{f}_i$$

- Because it makes *assembly* of reduced models fast.
- Each $\boldsymbol{A}_i$ and $\boldsymbol{f}_i$ can be projected independently onto a reduced basis and stored.
- This makes the online stage completely high-fidelity-agnostic.
- Deriving these *affine representations* is the core detail of RBM.

Reduced Order Models (ROM) for Navier-Stokes

Navier-Stokes equations

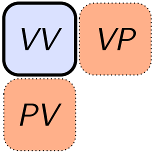
$$
\begin{aligned}
-\nu\Delta\boldsymbol{u} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nabla p = \boldsymbol{f} \qquad & \text{in } \Omega,\\
\nabla\cdot\boldsymbol{u} = 0 \qquad & \text{in } \Omega,\\
\boldsymbol{u} = \boldsymbol{g} \qquad & \text{on } \Gamma_{\mathrm{D}}, |\Gamma_D| > 0\\
-p\boldsymbol{n} + \nu(\nabla\boldsymbol{u})\boldsymbol{n} = \boldsymbol{h} \qquad & \text{on } \Gamma_{\mathrm{N}}.
\end{aligned}
$$

Divergence-conforming methods

- Balance between velocity and pressure space is delicate.
- Too many velocity DoFs: poor enforcement of the continuity equation. Too many pressure DoFs: pressure instability.
- A pair of spaces (V, P) is divergence-conforming if $\operatorname{div} V = P$. Ensures the strong (pointwise) form of the continuity equation, and no pressure instabilities.
- IGA unlocks easy construction of such spaces.[2]

[2] Buffa, Sangalli, and Vázquez 2010; Evans and Hughes 2013.

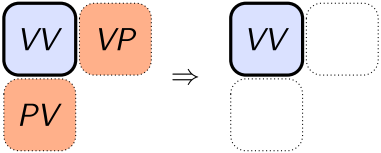
Anatomy of a reduced system



- Usually, the reduced method does not inherit the stability properties of the high-fidelity method.
- Rank-deficient velocity-pressure blocks are common.
- Carefully selecting number of velocity modes and pressure modes seems to help?

$$M_{\mathrm{V}} \approx d M_{\mathrm{P}}$$

Anatomy of a reduced system (cont.)



- A divergence-free reduced basis eliminates the coupling, leading to a fully stable velocity-only formulation.
- IGA[3] enables divergence-free high-fidelity solutions,[4] therefore also divergence-free reduced bases.

[3] Hughes, Cottrell, and Bazilevs 2005; Cottrell, Hughes, and Bazilevs 2009.
[4] Evans and Hughes 2013.

Divergence-free basis

Hi-Fi solutions

↓

Reference geometry

↓

Unlifted solutions

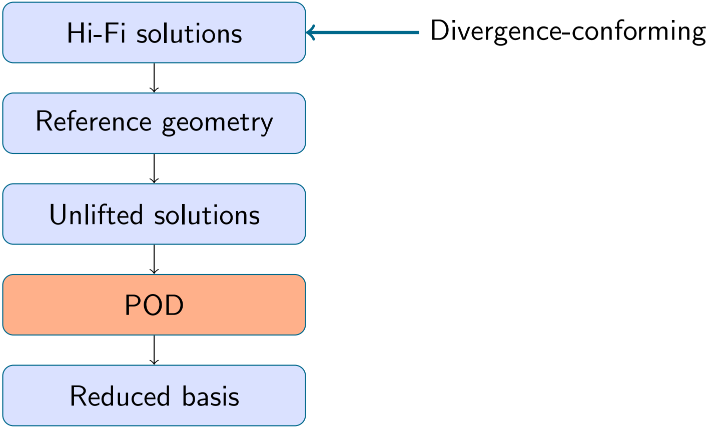
↓

POD

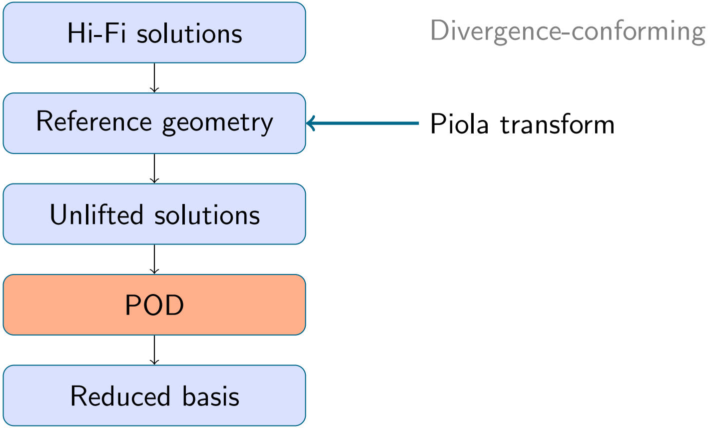
↓

Reduced basis

Divergence-free basis



Divergence-free basis



Divergence-free basis

Hi-Fi solutions → Divergence-conforming

↓

Reference geometry → Piola transform

↓

Unlifted solutions ← Divergence-free lift

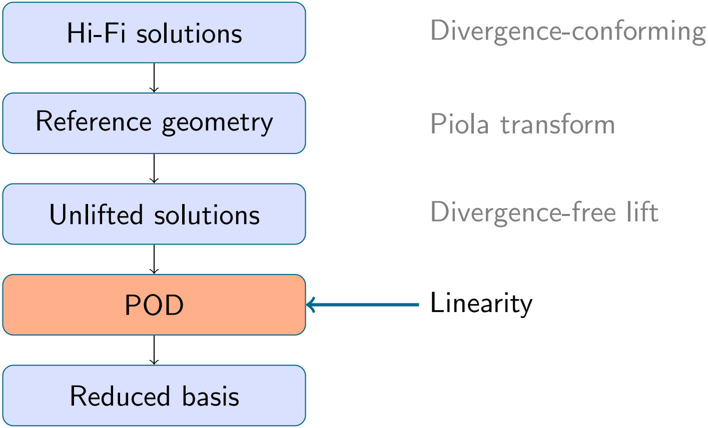
↓

POD

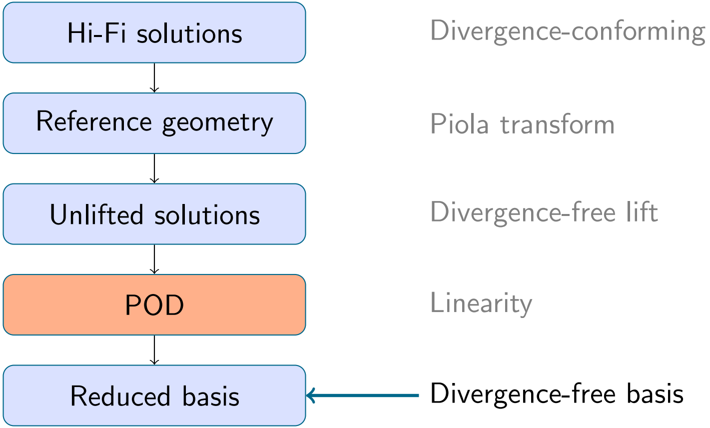
↓

Reduced basis

Divergence-free basis

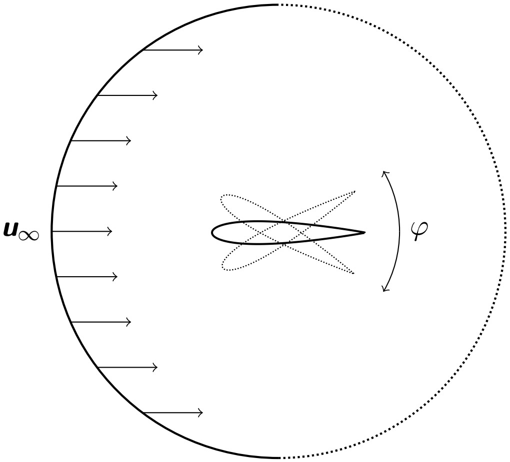


Divergence-free basis

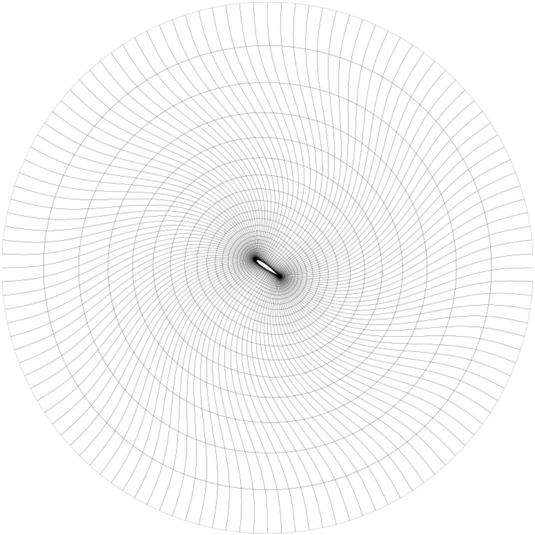


Numerical example: Flow around airfoil

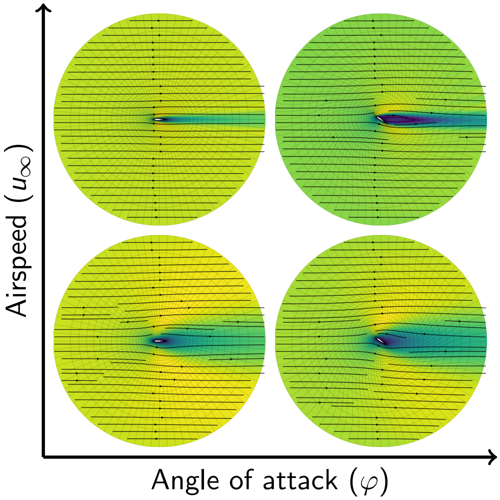
Flow around airfoil



Domain transformation



Parameter space



Problem specification

- We will try two high-fidelity methods: a Taylor-Hood (1,2)-method and an IGA (1,2) divergence-conforming method, with both approaches to pressure recovery.
- The parameter domain was chosen as

$$\mathcal{P} = [-35^\circ, +35^\circ] \times [1\,\mathrm{m/s}, 20\,\mathrm{m/s}].$$

- Only *stationary* Navier-Stokes, with $\nu = \frac{1}{6}$.
- We chose equal number of modes in all spaces: $N_{\mathrm{V}} = N_{\mathrm{S}} = N_{\mathrm{P}} = M$.

Affine representations

- Not possible to express the Navier-Stokes problem as finite sums

$$\boldsymbol{A}_h(\boldsymbol{\mu}) = \sum_i \xi_i(\boldsymbol{\mu})\boldsymbol{A}_i, \qquad \boldsymbol{f}_h(\boldsymbol{\mu}) = \sum_i \chi_i(\boldsymbol{\mu})\boldsymbol{f}_i$$

- Instead, we use truncated polynomial series in φ.[5]
- We can expect about 10 digits of accuracy with a reasonable number of terms (∼25 for TH, ∼75 for DC).
- Recall: the intention is to encode *all* parameters explicitly in the representation of the bi- or trilinear forms.

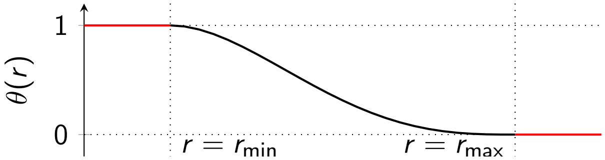
[5] Fonn et al. 2019.

Affine representations (cont.)

- The Jacobian of the domain transformation can be expressed as

$$\boldsymbol{J} = \boldsymbol{R}(a)(\boldsymbol{I} + \varphi \boldsymbol{P}\boldsymbol{Q})$$

- $\boldsymbol{R}$ is a rotation matrix through angle a.
- $a(r) = \varphi\theta(r)$, where θ is any suitably smooth "canonical" angle function



- $\boldsymbol{Q} = \frac{\theta'}{r}\hat{\boldsymbol{r}}\hat{\boldsymbol{r}}^\intercal$, and $\boldsymbol{P} = \boldsymbol{R}(\pi/2)$.

Affine representations (cont.)

- It can also be determined that, as expected, $\det \boldsymbol{J} = 1$.
- And
$$\boldsymbol{J}^{-1} = (\boldsymbol{I} - \varphi \boldsymbol{P}\boldsymbol{Q})\, \boldsymbol{R}(a)^{\mathsf{T}}.$$
- As for $\boldsymbol{R}$, we have
$$\boldsymbol{R}(a) = \sum_i \frac{a^i}{i!} \boldsymbol{P}^i = \sum_i \varphi^i \boldsymbol{R}_i$$
where $\boldsymbol{R}_i = \theta^i \boldsymbol{P}^i / i!$
- Summary: any expression involving $\boldsymbol{J}$, $\det \boldsymbol{J}$ and $\boldsymbol{J}^{-1}$ can be given an affine representation in terms of a truncated series expansion of $\boldsymbol{R}$.

Affine representations (TH)

$$
\begin{aligned}
(\boldsymbol{\pi}_{\boldsymbol{\mu}}^{*} a)(\hat{\boldsymbol{u}}, \hat{\boldsymbol{w}}; \varphi) &= \nu \int_{\hat{\Omega}} (\boldsymbol{J}^{-\intercal}\nabla)\hat{\boldsymbol{u}} : (\boldsymbol{J}^{-\intercal}\nabla)\hat{\boldsymbol{w}} |\boldsymbol{J}| \\
&= \nu \int_{\hat{\Omega}} \nabla\hat{\boldsymbol{u}} : (\boldsymbol{J}^{-1}\boldsymbol{J}^{-\intercal}\nabla)\hat{\boldsymbol{w}} |\boldsymbol{J}|,
\end{aligned}
$$

and we find (based on the above) that

$$
\begin{aligned}
\boldsymbol{J}^{-1}\boldsymbol{J}^{-\intercal} &= (\boldsymbol{I} - \varphi \boldsymbol{PQ})\boldsymbol{R}^{\intercal}\boldsymbol{R}(\boldsymbol{I} + \varphi \boldsymbol{QP}) \\
&= \boldsymbol{I} + \varphi \underbrace{(\boldsymbol{QP} - \boldsymbol{PQ})}_{\boldsymbol{D}_1} - \varphi^2 \underbrace{\boldsymbol{PQ}^2\boldsymbol{P}}_{\boldsymbol{D}_2} \\
&= \boldsymbol{I} + \varphi \boldsymbol{D}_1 - \varphi^2 \boldsymbol{D}_2.
\end{aligned}
$$

Affine representations (TH; cont.)

Thus we have, in three terms

$$\begin{aligned}(\boldsymbol{\pi}_{\boldsymbol{\mu}}^* a)(\hat{\boldsymbol{u}}, \hat{\boldsymbol{w}}; \varphi) = \nu \int_{\hat{\Omega}} \nabla \hat{\boldsymbol{u}} : \nabla \hat{\boldsymbol{w}} \\ + \nu\varphi \int_{\hat{\Omega}} \nabla \hat{\boldsymbol{u}} : (\boldsymbol{D}_1 \nabla) \hat{\boldsymbol{w}} \\ - \nu\varphi^2 \int_{\hat{\Omega}} \nabla \hat{\boldsymbol{u}} : (\boldsymbol{D}_2 \nabla) \hat{\boldsymbol{w}}.\end{aligned}$$

Affine representations (TH; cont.)

Similarly,

$$(\boldsymbol{\pi}_{\boldsymbol{\mu}}^{*} b)(\hat{p}, \hat{\boldsymbol{w}}; \varphi) = \int_{\hat{\Omega}} \hat{p}(\boldsymbol{J}^{-\mathsf{T}} \nabla) \cdot \hat{\boldsymbol{w}} |\boldsymbol{J}| = \int_{\hat{\Omega}} \hat{p} \boldsymbol{J}^{-\mathsf{T}} : \nabla \hat{\boldsymbol{w}} |\boldsymbol{J}|,$$

and we have

$$\begin{aligned} \boldsymbol{J}^{-\mathsf{T}} &= \boldsymbol{R}(a)(\boldsymbol{I} - \varphi \boldsymbol{Q}^{\mathsf{T}} \boldsymbol{P}^{\mathsf{T}}) = \boldsymbol{R}(a)(\boldsymbol{I} + \varphi \boldsymbol{Q}\boldsymbol{P}) = \sum_i \varphi^i \boldsymbol{R}_i (\boldsymbol{I} + \varphi \boldsymbol{Q}\boldsymbol{P}) \\ &= \sum_i \varphi^i \boldsymbol{R}_i + \varphi^{i+1} \boldsymbol{R}_i \boldsymbol{Q}\boldsymbol{P} = \sum_i \varphi^i \underbrace{(\boldsymbol{R}_i + \boldsymbol{R}_{i-1} \boldsymbol{Q}\boldsymbol{P})}_{\boldsymbol{B}_i^{(-)}} = \sum_i \varphi^i \boldsymbol{B}_i^{(-)}, \end{aligned}$$

Affine representations (TH; cont.)

So the last two terms can be expressed as

$$
\begin{aligned}
(\boldsymbol{\pi}^*_{\boldsymbol{\mu}} b)(\hat{p}, \hat{\boldsymbol{w}}; \varphi) &= \int_{\hat{\Omega}} \hat{p} \boldsymbol{J}^{-\mathsf{T}} : \nabla \hat{\boldsymbol{w}} |\boldsymbol{J}|, \\
&\approx \sum_{i=0}^{2n} \varphi^i \int_{\hat{\Omega}} \hat{p} \boldsymbol{B}_i^{(-)} : \nabla \hat{\boldsymbol{w}} \\
(\boldsymbol{\pi}^*_{\boldsymbol{\mu}} c)(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}, \hat{\boldsymbol{w}}; \varphi) &= \int_{\hat{\Omega}} (\hat{\boldsymbol{u}} \cdot \boldsymbol{J}^{-\mathsf{T}} \nabla) \hat{\boldsymbol{v}} \cdot \hat{\boldsymbol{w}} |\boldsymbol{J}| \\
&\approx \sum_{i=0}^{2n} \varphi^i \int_{\hat{\Omega}} (\hat{\boldsymbol{u}} \cdot \boldsymbol{B}_i^{(-)} \nabla) \hat{\boldsymbol{v}} \cdot \hat{\boldsymbol{w}}.
\end{aligned}
$$

Affine representations (IGA)

This proceeds as for TH, except the Piola transform mandates that every vector field be pre-multiplied by $\boldsymbol{J}/\det\boldsymbol{J} = \boldsymbol{J}$. Note the similarity between $\boldsymbol{J}$ and $\boldsymbol{J}^{-\intercal}$:

$$\boldsymbol{J} = \boldsymbol{R}(a)(\boldsymbol{I} + \varphi\boldsymbol{PQ}) = \sum_{i=0}^{\infty} \varphi^i \boldsymbol{R}_i(\boldsymbol{I} + \varphi\boldsymbol{PQ}) = \sum_{i=0}^{\infty} \varphi^i \boldsymbol{R}_i + \varphi^{i+1}\boldsymbol{PQ}$$

$$= \sum_{i=0}^{\infty} \varphi^i \underbrace{(\boldsymbol{R}_i + \boldsymbol{R}_{i-1}\boldsymbol{PQ})}_{\boldsymbol{B}_i^{(+)}} = \sum_{i=0}^{\infty} \varphi^i \boldsymbol{B}_i^{(+)}.$$

Affine representations (IGA; cont.)

For b,

$$
\begin{aligned}
(\boldsymbol{\pi}_{\boldsymbol{\mu}}^{*} b)(\hat{p}, \hat{\boldsymbol{w}}; \varphi) &= \int_{\hat{\Omega}} \hat{p} \boldsymbol{J}^{-\mathsf{T}} : \nabla(\boldsymbol{J} \hat{\boldsymbol{w}}) |\boldsymbol{J}| \\
&\approx \int_{\hat{\Omega}} \hat{p} \left(\sum_{i=0}^{2n} \varphi^{i} \boldsymbol{B}_{i}^{(-)} \right) : \nabla \left(\sum_{j=0}^{2n} \varphi^{j} \boldsymbol{B}_{j}^{(+)} \hat{\boldsymbol{w}} \right) \\
&= \sum_{i,j=0}^{2n} \varphi^{i+j} \int_{\hat{\Omega}} \hat{p} \boldsymbol{B}_{i}^{(-)} : \nabla \left(\boldsymbol{B}_{j}^{(+)} \hat{\boldsymbol{w}} \right).
\end{aligned}
$$

Affine representations (IGA; cont.)

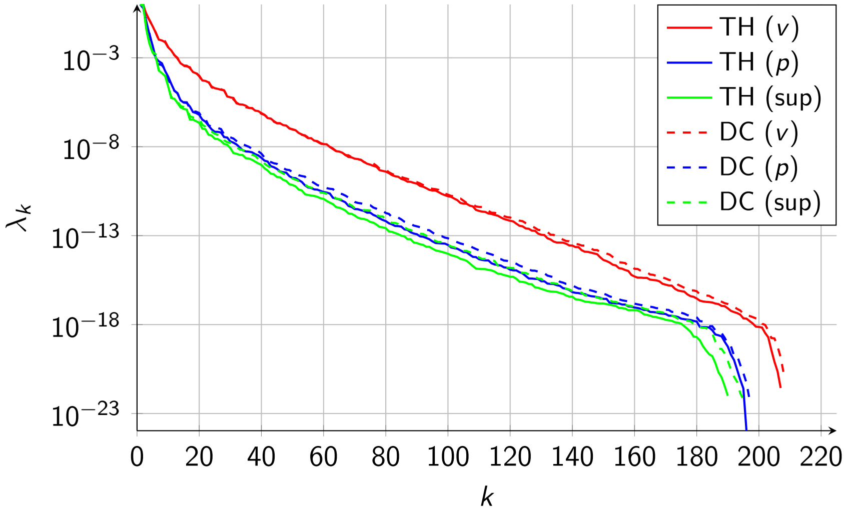
For c,

$$
\begin{aligned}
(\boldsymbol{\pi}_{\boldsymbol{\mu}}^{*} c)(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}, \hat{\boldsymbol{w}}; \varphi) &= \int_{\hat{\Omega}} (\boldsymbol{J}\hat{\boldsymbol{u}} \cdot \boldsymbol{J}^{-\mathsf{T}}\nabla)\boldsymbol{J}\hat{\boldsymbol{v}} \cdot \boldsymbol{J}\hat{\boldsymbol{w}}|\boldsymbol{J}| \\
&\approx \int_{\hat{\Omega}} (\hat{\boldsymbol{u}} \cdot \nabla) \left(\sum_{i=0}^{2n} \varphi^{i} \boldsymbol{B}_{i}^{(+)} \hat{\boldsymbol{v}} \right) \cdot \left(\sum_{i=0}^{2n} \varphi^{j} \boldsymbol{B}_{j}^{(+)} \hat{\boldsymbol{w}} \right) \\
&= \sum_{i,j=0}^{2n} \varphi^{i+j} \int_{\hat{\Omega}} (\hat{\boldsymbol{u}} \cdot \nabla) \boldsymbol{B}_{i}^{(+)} \hat{\boldsymbol{v}} \cdot \boldsymbol{B}_{j}^{(+)} \hat{\boldsymbol{w}}.
\end{aligned}
$$

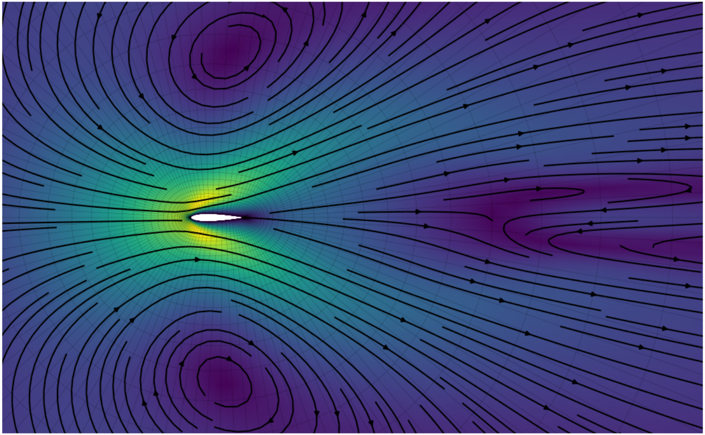
Affine representaitons (IGA; cont.)

$$
\begin{aligned}
(\pi_{\mu}^{*}a)(\hat{\boldsymbol{u}},\hat{\boldsymbol{w}};\varphi) &= \int_{\hat{\Omega}}(\boldsymbol{J}^{-\intercal}\nabla)(\boldsymbol{J}\hat{\boldsymbol{u}}):(\boldsymbol{J}^{-\intercal}\nabla)(\boldsymbol{J}\hat{\boldsymbol{w}})|\boldsymbol{J}| \\
&= \int_{\hat{\Omega}}\nabla(\boldsymbol{J}\hat{\boldsymbol{u}}):(\boldsymbol{J}^{-1}\boldsymbol{J}^{-\intercal}\nabla)(\boldsymbol{J}\hat{\boldsymbol{w}})|\boldsymbol{J}| \\
&\approx \int_{\hat{\Omega}}\nabla\left(\sum_{i=0}^{2n}\varphi^{i}\boldsymbol{B}_i^{(+)}\hat{\boldsymbol{u}}\right) \\
&\qquad\qquad :\left((\boldsymbol{I}+\varphi\boldsymbol{D}_1-\varphi^2\boldsymbol{D}_2)\nabla\right)\left(\sum_{j=0}^{2n}\varphi^{j}\boldsymbol{B}_j^{(+)}\hat{\boldsymbol{w}}\right) \\
&= \sum_{i,j=0}^{2n}\varphi^{i+j}\int_{\hat{\Omega}}\nabla(\boldsymbol{B}_i^{(+)}\hat{\boldsymbol{u}}):\nabla(\boldsymbol{B}_j^{(+)}\hat{\boldsymbol{w}}) \\
&+ \sum_{i,j=0}^{2n}\varphi^{i+j+1}\int_{\hat{\Omega}}\nabla(\boldsymbol{B}_i^{(+)}\hat{\boldsymbol{u}}):(\boldsymbol{D}_1\nabla)(\boldsymbol{B}_j^{(+)}\hat{\boldsymbol{w}}) \\
&- \sum_{i,j=0}^{2n}\varphi^{i+j+2}\int_{\hat{\Omega}}\nabla(\boldsymbol{B}_i^{(+)}\hat{\boldsymbol{u}}):(\boldsymbol{D}_2\nabla)(\boldsymbol{B}_j^{(+)}\hat{\boldsymbol{w}}),
\end{aligned}
$$

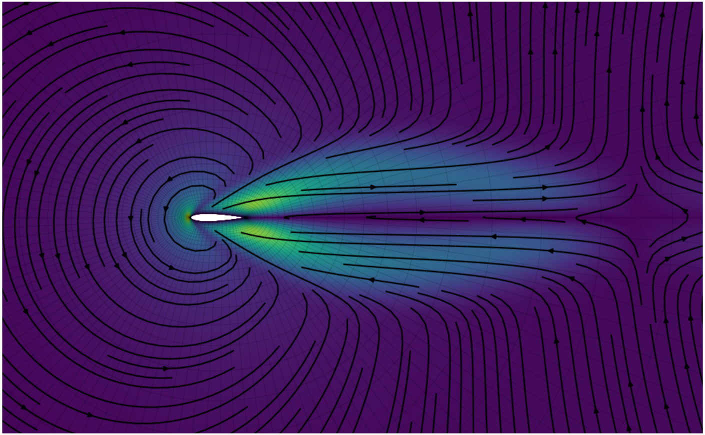
Spectrum


10^{-3}
10^{-8}
10^{-13}
10^{-18}
10^{-23}
λ_k
0
20
40
60
80
100
120
140
160
180
200
220
k
TH (v)
TH (p)
TH (sup)
DC (v)
DC (p)
DC (sup)

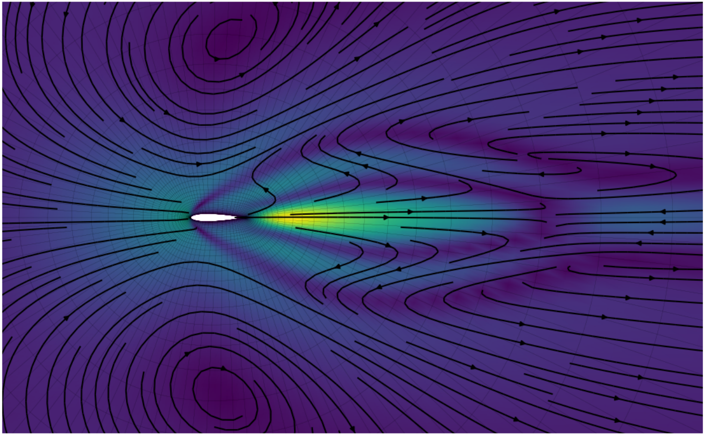
Basis functions (v, TH, 1)



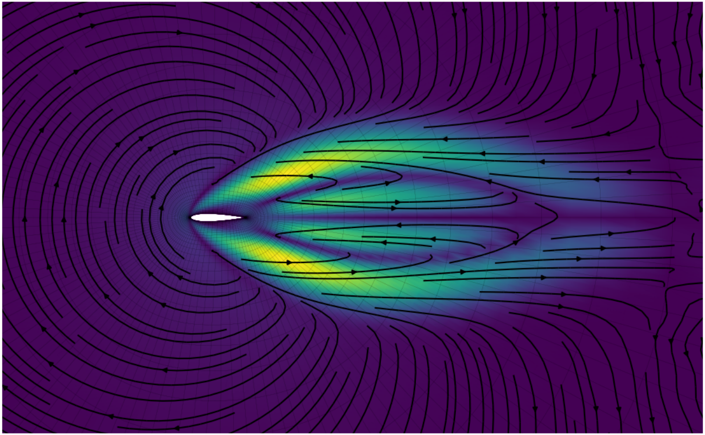
Basis functions (v, TH, 2)



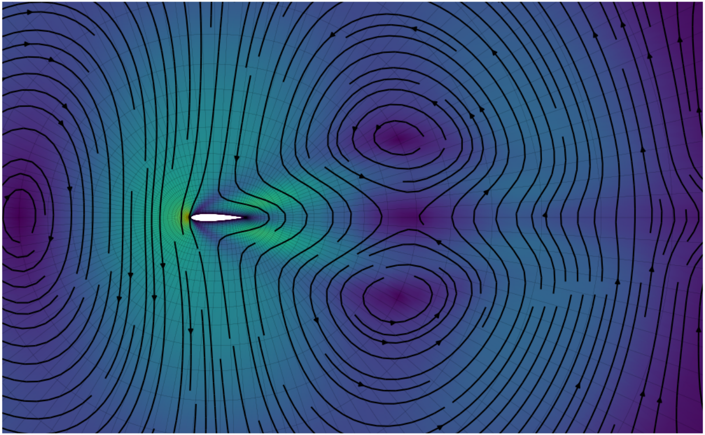
Basis functions (v, TH, 3)



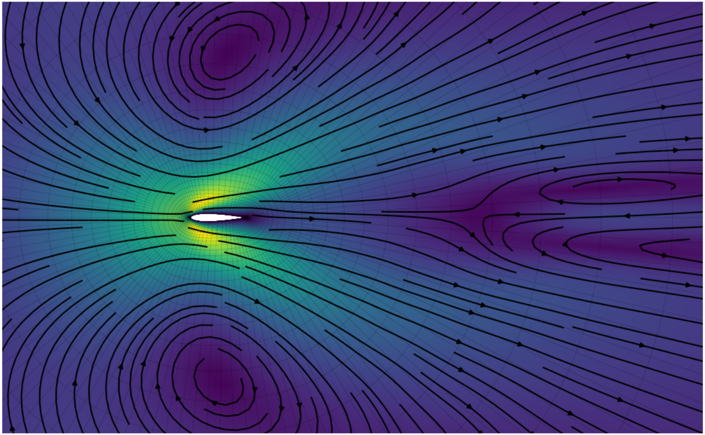
Basis functions (v, TH, 4)



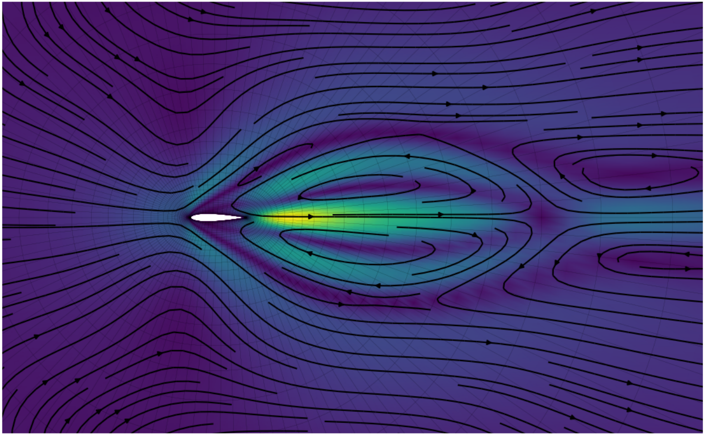
Basis functions (v, DC, 1)



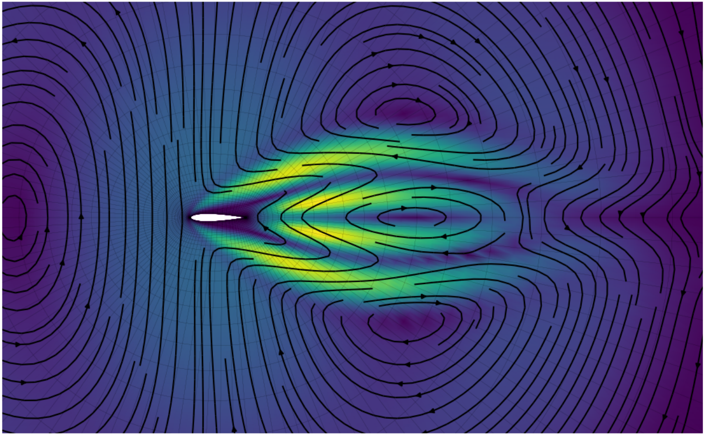
Basis functions (v, DC, 2)



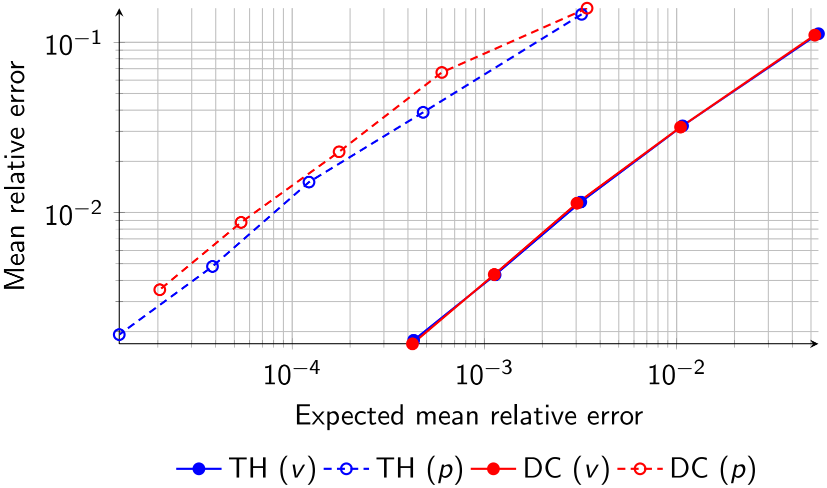
Basis functions (v, DC, 3)



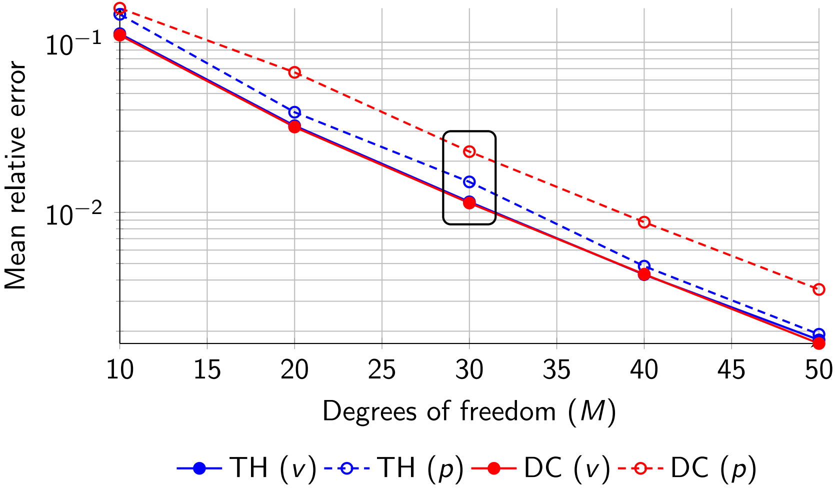
Basis functions (v, DC, 4)



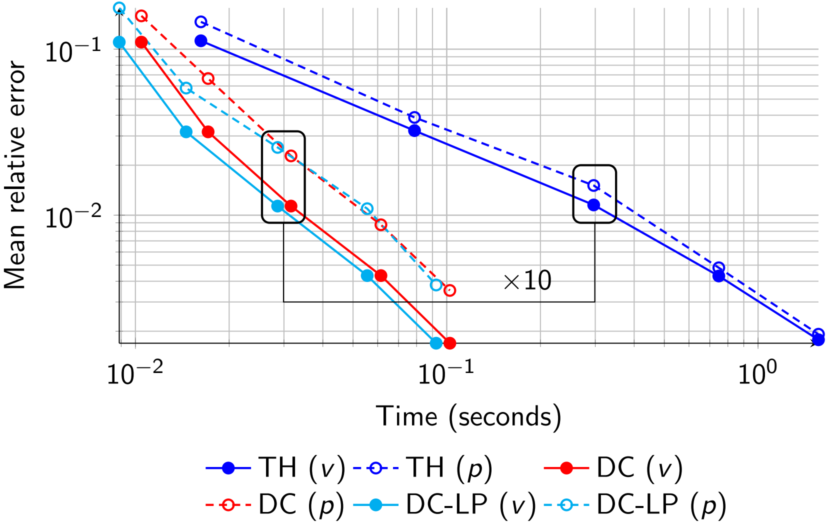
Convergence



Convergence



Convergence



Speedup factors

$\sharp$ DoFs (M)	**Taylor-Hood**	**Conforming**
10	1483	6890
20	390	2616
30	111	1441
40	52	843
50	27	502

First step towards certified ROM

First step towards certified ROM

- High-Fidelity snapshots of prescribed accuracy using adaptive IGA
- ROM with verified accuracy on training set (+ test set)

Total error in the reduced order model can be bounded as follows:

$$\|u - u_{\mathrm{M}}\| \leq \|u - u_h\| + \|u_h - u_{\mathrm{M}}\|$$

High Fidelity snapshots of presribed accuracy

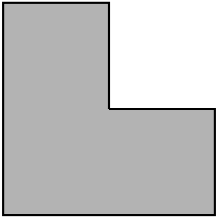
- Want guaranteed upper bound error estimates that is not too conservative
  - May be satisfactory with asymptotic exact error estimates
  - We use here recovery based error estimates (CGL2)
- May need to merge snapshots from different adapted meshes

ROM with verified accuracy on training set (+ test set)

- Select number of RBs based on error compared to HF:
  - Compute average error in the training set
  - Compute the maximum error in the training set
- Can extend the comparison on a randomly chosen test set

Numerical example: L-shape deformation (adaptive)

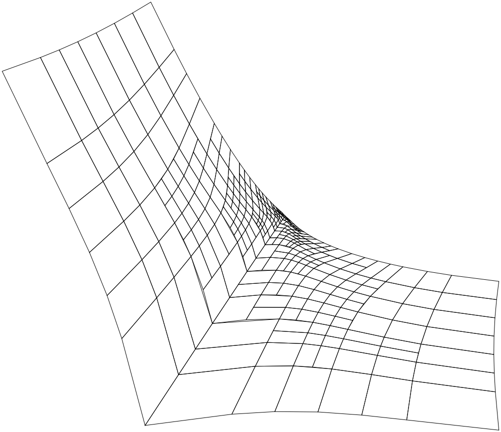
L-shape deformation



Neumann conditions (almost) everywhere, based on analytic solution.

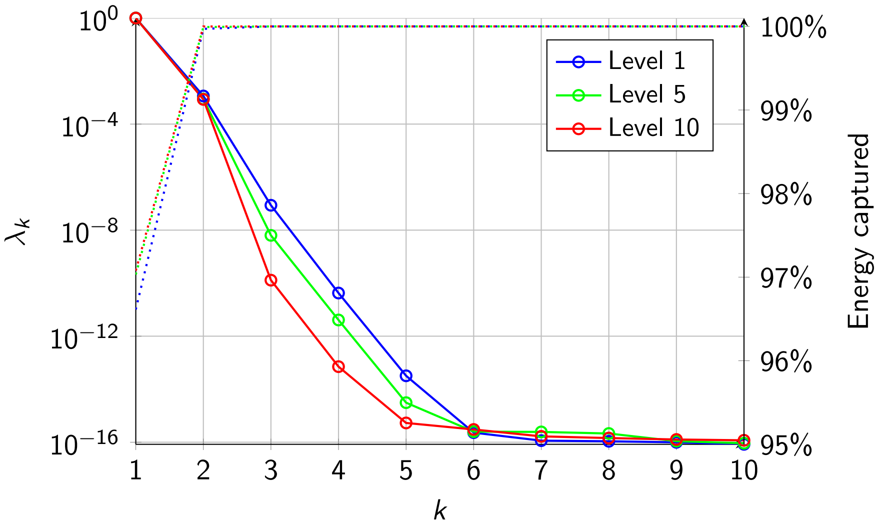
- $E \in [1 \times 10^5\,\text{Pa}, 3 \times 10^5\,\text{Pa}]$
- $\nu \in [0.2, 0.4]$

L-shape deformation



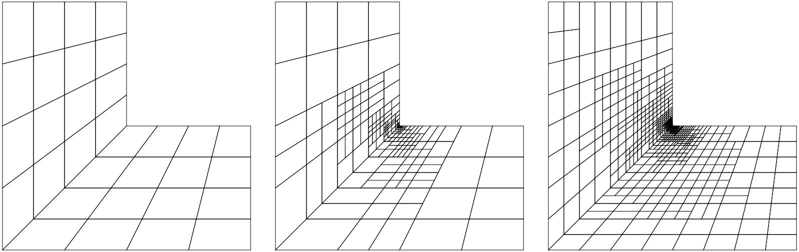
Typical solution with singularity in the corner.

Spectra



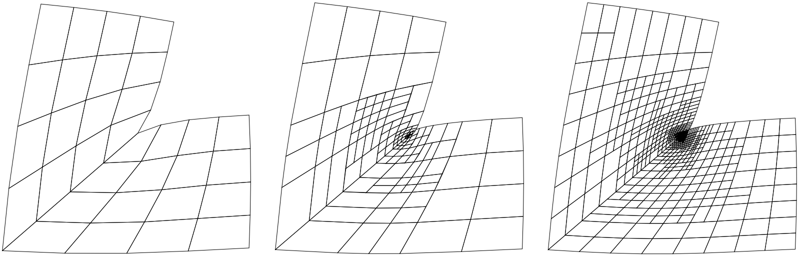
10^0
10^{-4}
10^{-8}
10^{-12}
10^{-16}
λ_k
1 2 3 4 5 6 7 8 9 10
k
100%
99%
98%
97%
96%
95%
Energy captured
Level 1
Level 5
Level 10

Meshes



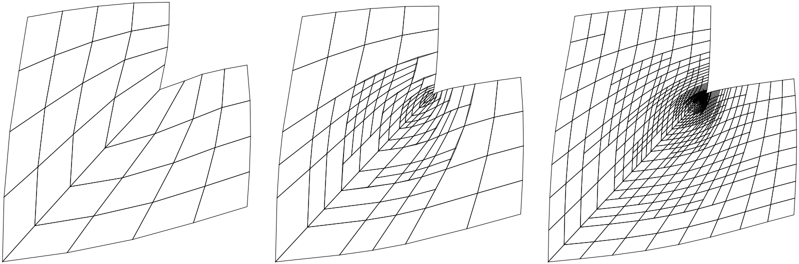
Level 1, 5 and 10.

Mode 1



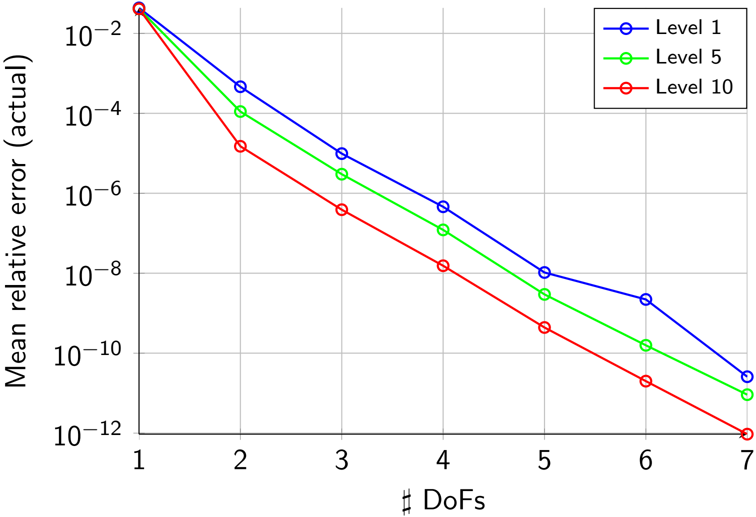
Level 1, 5 and 10.

Mode 2

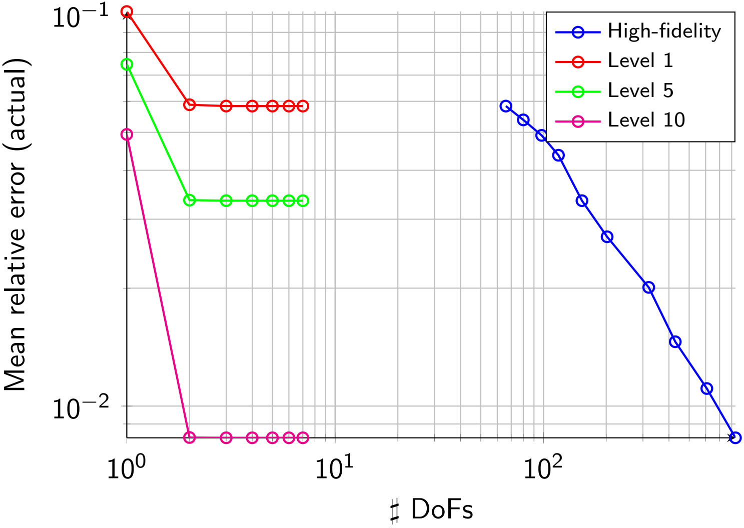


Level 1, 5 and 10.

Mean relative ROM error per DoF



Mean relative total error per DoF

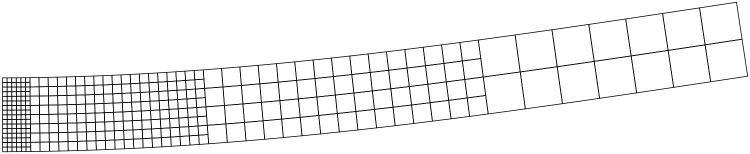


Numerical example: Beam deformation (adaptive)

Beam deformation

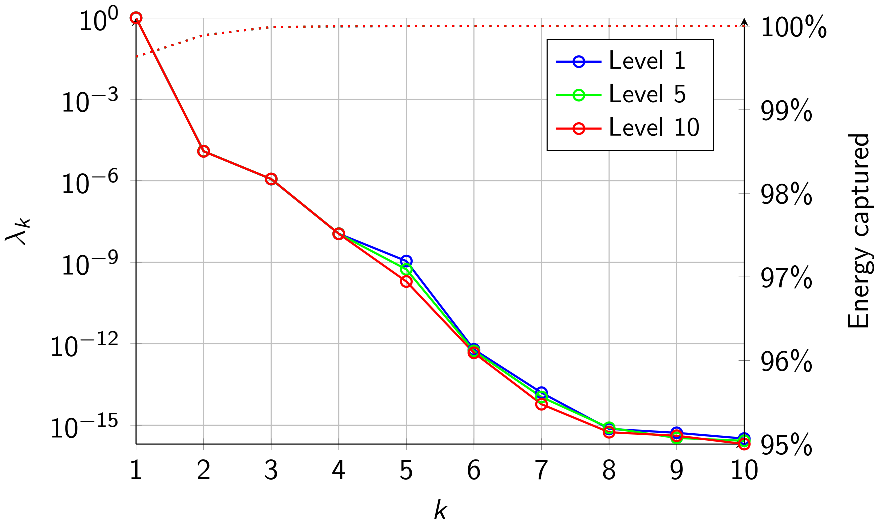
- $E \in [1 \times 10^5\,\mathrm{Pa}, 5 \times 10^5\,\mathrm{Pa}]$
- $\nu \in [0.2, 0.4]$
- $\boldsymbol{f}_x, \boldsymbol{f}_y \in [-10\,\mathrm{Pa}, 10\,\mathrm{Pa}]$

Beam deformation



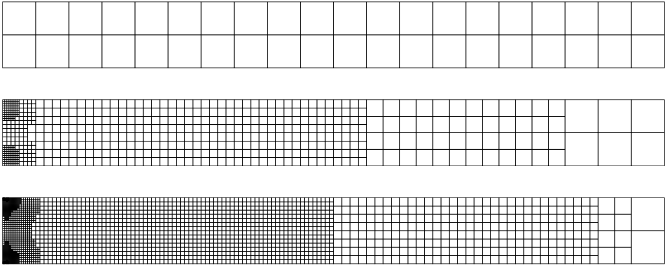
Typical solution with refinement towards the Dirichlet boundary.

Spectra



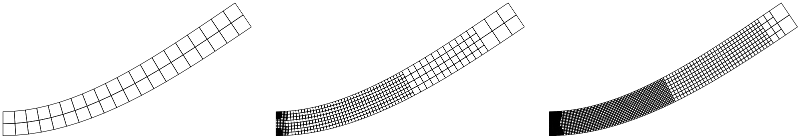
10^{0}
10^{-3}
10^{-6}
10^{-9}
10^{-12}
10^{-15}
λ_k
1 2 3 4 5 6 7 8 9 10
k
100%
99%
98%
97%
96%
95%
Energy captured
Level 1
Level 5
Level 10

Meshes



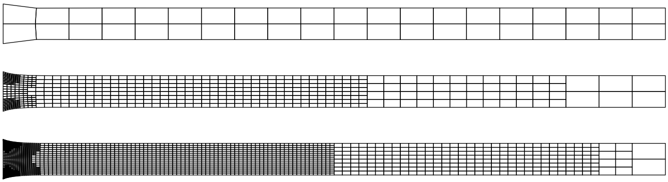
Level 1, 5 and 8.

Mode 1



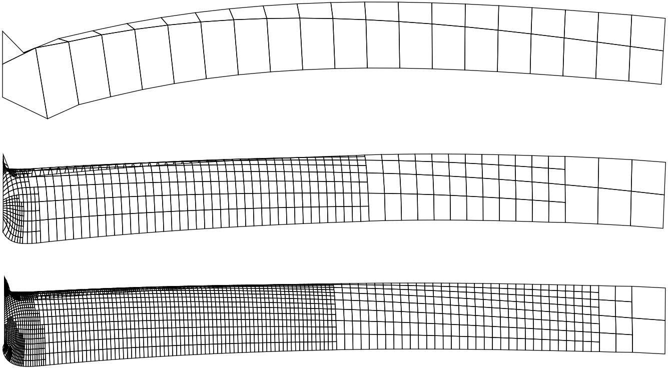
Level 1, 5 and 8.

Mode 2



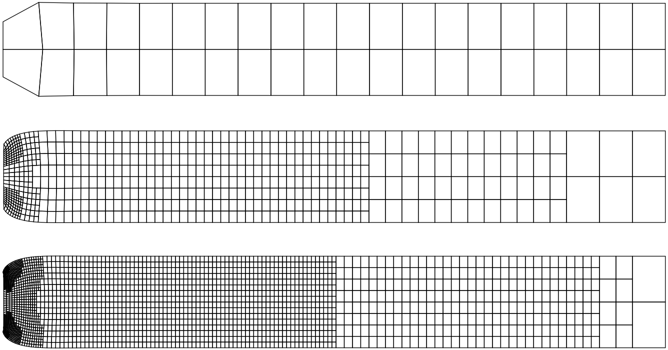
Level 1, 5 and 8.

Mode 3



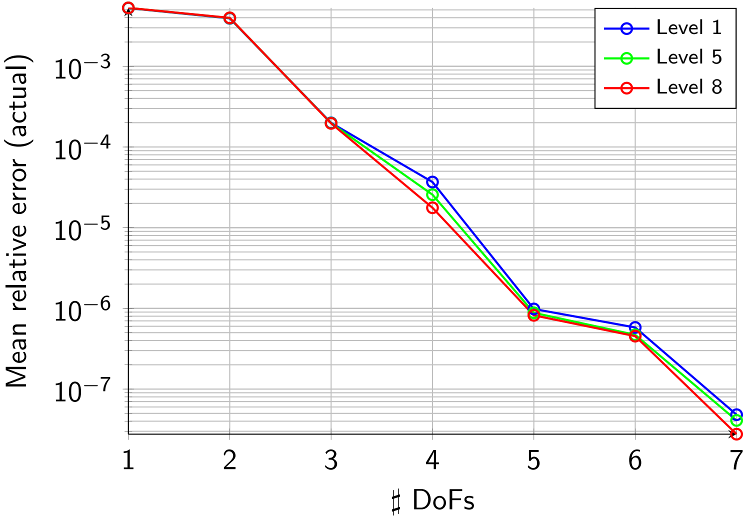
Level 1, 5 and 8.

Mode 4



Level 1, 5 and 8.

Mean relative ROM error per DoF (Beam)



Summary

Summary

- Reduced order models offer dramatic speed-ups for certain applications.
- They combine nicely with IGA and div-compatible spaces to form fully divergence-free function spaces without need for pressure fields.
- Divergence-free RBMs can be much faster than conventional RBMs.
- The number of reduced basis functions is moderately affected by highly refined snapshot meshes.
- Adaptive IGA combined with ROM is a promising enabling technology for real-time computational modelling.

Thank you!

References

Buffa, Annalisa, Giancarlo Sangalli, and Rafael Vázquez (2010). "Isogeometric analysis in electromagnetics: B-splines approximation". In: *Computer Methods in Applied Mechanics and Engineering* 199.17-20, pp. 1143–1152.

Cottrell, J Austin, Thomas R J Hughes, and Yuri Bazilevs (2009). *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley.

Evans, John A. and Thomas J R Hughes (2013). "Isogeometric divergence-conforming B-splines for the steady Navier-Stokes equations". In: *Mathematical Models and Methods in Applied Sciences* 23.8, pp. 1421–1478. ISSN: 0218-2025. DOI: 10.1142/S0218202513500139.

Fonn, E et al. (2019). "Fast divergence-conforming reduced basis methods for steady Navier–Stokes flow". In: *Computer Methods in Applied Mechanics and Engineering* 346, pp. 486–512.

Hughes, Thomas JR, John A Cottrell, and Yuri Bazilevs (2005). "Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement". In: *Computer methods in applied mechanics and engineering* 194.39, pp. 4135–4195.

Quarteroni, Alfio, Andrea Manzoni, and Federico Negri (2016). *Reduced basis methods for partial differential equations*. Springer International Publishing.