Simultaneous spatio-parameter adaptivity for parametrized problems in CFD: An adaptive DG-RBE method

Masayuki Yano

University of Toronto

SIAM AN 2016
Boston, United States
14 July 2016

Model reduction and general CFD analysis

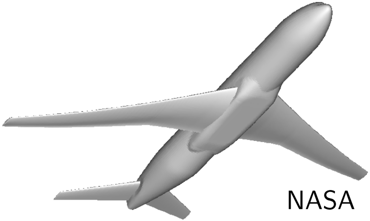
A reduced order model (ROM)

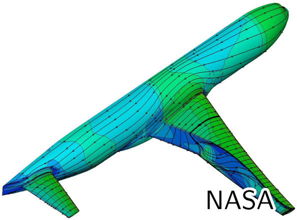
provides rapid and reliable responses

for the real-time/many-query case for which it was trained.

But, ROMs lack the versatility of FEM for general CFD analysis.

general geometry & condition $\Rightarrow$ flow solution





Question. Can we incorporate model reduction concepts to accelerate/compress general CFD analysis?

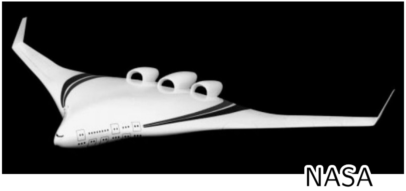
Model reduction and general CFD analysis

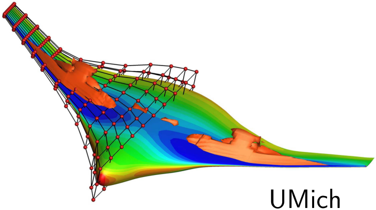
A reduced order model (ROM)

provides rapid and reliable responses
for the real-time/many-query case for which it was trained.

But, ROMs lack the versatility of FEM for general CFD analysis.

general geometry & condition $\Rightarrow$ flow solution





Question. Can we incorporate model reduction concepts to accelerate/compress general CFD analysis?

Model reduction and general CFD analysis

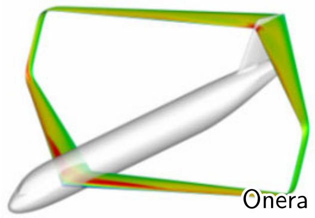
A reduced order model (ROM)

provides rapid and reliable responses

for the real-time/many-query case for which it was trained.

But, ROMs lack the versatility of FEM for general CFD analysis.

general geometry & condition $\Rightarrow$ flow solution

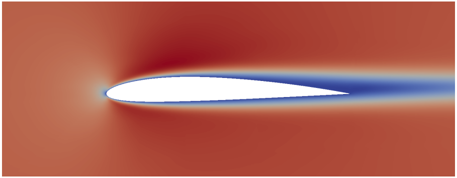


Question. Can we incorporate model reduction concepts to accelerate/compress general CFD analysis?

Requirements

An effective CFD solver must be versatile and provide

- support for general equations — nonlinear, convection, ...
- geometric flexibility — non-parametric, topology, ...
- parametric flexibility — M_∞, Re, α, ...
- reliable predictions — output error estimate & adaptivity
- minimal case-specific training — ~ 10 runs unaffordable.

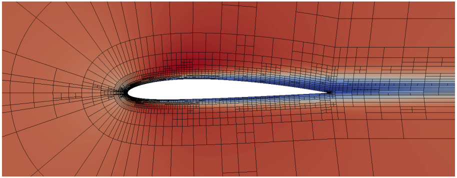


Localized reduction: boundary layer

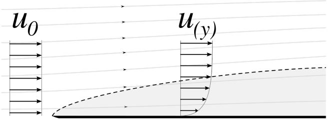
Idea: *forgo reduction of a specific case,*
and focus on reducible & reusable features

In this talk: focus on boundary layers (BLs)

Reason 1: BLs require high resolution for high Re.



Reason 2: BLs are similarly shaped (Blasius, Falkner-Skan).



Formulation

- Discretization
- Reduced basis element (RBE)
- Stability
- Error estimation and adaptation
- Related work

Formulation

- Discretization
- Reduced basis element (RBE)
- Stability
- Error estimation and adaptation
- Related work

Discontinuous Galerkin (DG) method

System of steady conservation laws ($d+2$ eqs for NS)

$$\nabla \cdot F^{\text{inv}}(u) + \nabla \cdot F^{\text{visc}}(u, \nabla u) = 0 \quad \text{in } \Omega\,; \quad (+\text{BCs}).$$

DG: introduce a discontinous FE space $\mathcal{V}_n$;

find $u_n \in \mathcal{V}_n$ such that

$$R_n(u_n, v_n) = 0 \quad \forall v_n \in \mathcal{V}_n,$$

where

$$R_n(w_n, v_n) = -\int_{\Omega_h} \nabla v_n \cdot F^{\text{inv}}(w_n) dx + \int_{\Sigma_h} v_n^+ \hat{F}^{\text{inv}}(\cdots) ds + \cdots .$$

Features:

- Flexible choice of FE spaces
- Stability for conservation laws

  Roe's Riemann solver for $\hat{F}^{\text{inv}}$ and BR2 for F^{visc}

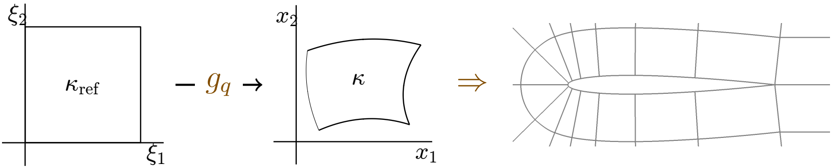
Polynomial FE spaces & geometry representation

FE space:

$$\mathcal{V}_n = \mathcal{V}_{h,p} = \{\underbrace{v \in L^2(\Omega)}_{\text{discontinuous}} : \underbrace{(v \circ g_q)|_\kappa \in \mathbb{P}_p(\kappa_{\text{ref}})}_{\text{polynomial in each ref element}}, \underbrace{\kappa_{\text{ref}} \in \mathcal{T}_{h,\text{ref}}}_{\text{tessellation}}\}.$$

Geometry mapping: degree-q polynomial

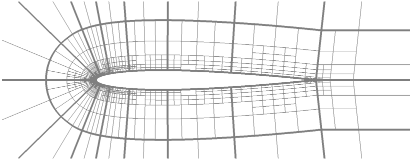
$$g_q : \kappa_{\text{ref}} \to \kappa, \qquad \xi \mapsto x.$$



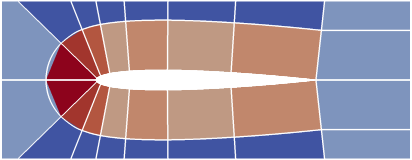
Polynomial FE spaces: adaptivity

FE space: $\mathcal{V}_{h,p} = \{v \in L^2(\Omega) \, : \, (v \circ g_q)|_\kappa \in \mathbb{P}_p(\kappa_{\rm ref}), \;\; \kappa_{\rm ref} \in \mathcal{T}_{h,{\rm ref}}\}.$

h-adaptivity: $u_h \to u$ as $h \to 0$



p-adaptivity: $u_p \to u$ as $p \to \infty$



Polynomial spaces with good general approximability in each κ.

Formulation

- Discretization
- Reduced basis element (RBE)
- Stability
- Error estimation and adaptation
- Related work

Reduced basis element (RBE) [Maday & Rønquist, 2005; ...]

Idea. Construct spaces with feature-specific approximability.

Discontinuous non-polynomial FE space

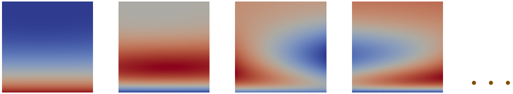
$$\mathcal{V}_n = \{v \in L^2(\Omega) \,:\, (v \circ g_q)|_\kappa \in \mathbb{FE}_n(\kappa_{\mathrm{ref}}),\ \kappa_{\mathrm{ref}} \in \mathcal{T}_{h,\mathrm{ref}}\}.$$

where

$$\mathbb{FE}_n(\kappa_{\mathrm{ref}}) = \begin{cases} \mathbb{RBE}_N(\kappa_{\mathrm{ref}}), & \kappa \in \text{reducible region} \\ \mathbb{P}_p(\kappa_{\mathrm{ref}}), & \text{otherwise} \end{cases}.$$

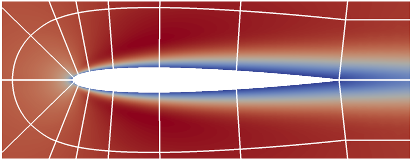
Reduced basis elements ($\mathbb{RBE}_N$) are

1. tailored for a specific reducible feature (e.g. BLs)
2. hierarchical ($N = 1, 2, \ldots$).



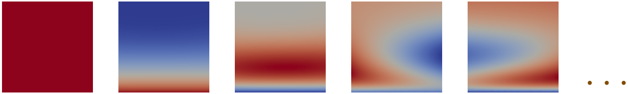
DG-RBE: offline training for a feature (*not* for a case)

Step 1. Solve training cases using h/p-adaptive FEM



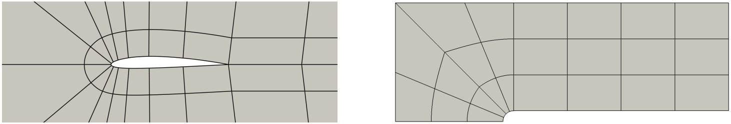
Step 2. Extract wall functions in reference element κ_{ref}

Step 3. Identify wall modes using POD $\rightarrow$ RBEs

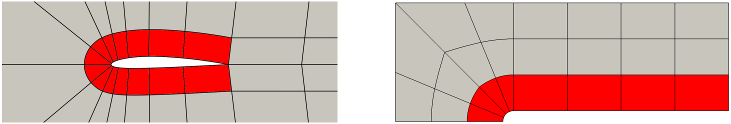


DG-RBE: online

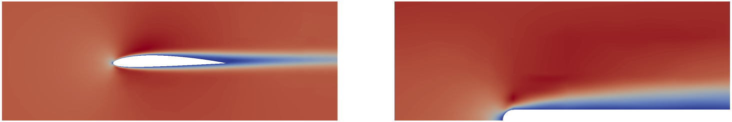
Step 1. Tessellate the domain (as in FEM)



Step 2. Mark reducible features (e.g. BLs)



Step 3. Solve the DG system using RBEs for reducible features and polynomial FEs for the rest.



Formulation

- Discretization
- Reduced basis element (RBE)
- **Stability**
- Error estimation and adaptation
- Related work

Stability

For entropy variables, the discontinuous Galkerin method is stable.
[Barth, 1999; Harten 1983; Hughes et al 1986]

Linear equations: energy stability

$$\|u_n(T)\|_M^2 \leq \|u_n(0)\|_M^2 + (\text{inflow data}).$$

Nonlinear equations: entropy stability

$$U(u_n(T)) \leq U(u_n(0)) + (\text{inflow data}),$$

where $U(\cdot)$ is a generalized convex entropy function.

Formulation

- Discretization
- Reduced basis element (RBE)
- Stability
- Error estimation and adaptation
- Related work

Output error estimation: dual-weighted residual (DWR)

[Becker & Rannacher, 1996; Ainsworth & Oden, 1998; ...]

Adjoint: for an output functional J, find $\psi_n \in \mathcal{V}_n^{\mathrm{adj}} \neq \mathcal{V}_n$ s.t.

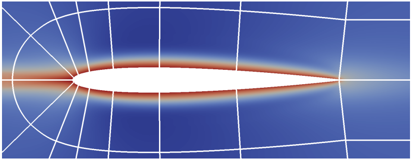
$$R_n'(u_n; v_n, \psi_n) = J_n'(u_n; v_n) \quad \forall v_n \in \mathcal{V}_n^{\mathrm{adj}}.$$

Global error estimate (wrt exact PDE *not* FE "truth"):

$$E \equiv J(u) - J_n(u_n) \approx -\mathcal{R}_n(u_n, \psi_n).$$

Element-wise error estimate:

$$\eta_\kappa \equiv |R_n(u_n, \psi_n|_\kappa)|, \quad \kappa \in \mathcal{T}_h.$$



Adjoint approximation

Effective error estimate requires an accurate adjoint
$\Rightarrow$ crucial for aggressive model reduction.

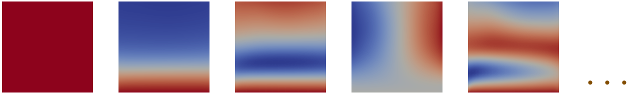
Adjoint feature-specific FE spaces:

$$\mathcal{V}_n^{\text{adj}} = \{v \in L^2(\Omega) \,:\, (v \circ g_q)|_\kappa \in \mathbb{FE}_n^{\text{adj}}(\kappa_{\text{ref}}),\ \kappa_{\text{ref}} \in \mathcal{T}_{h,\text{ref}}\}.$$

where

$$\mathbb{FE}_n(\kappa_{\text{ref}}) = \begin{cases} \mathbb{RBE}_N^{\text{adj}}(\kappa_{\text{ref}}), & \kappa \in \text{reducible region} \\ \mathbb{P}_{p+1}(\kappa_{\text{ref}}), & \text{otherwise} \end{cases}.$$

Adjoint RBEs ($\mathbb{RBE}_N^{\text{adj}}$) extracted from training + POD



Adaptivity

Employ

$$\texttt{Solve} \to \texttt{Estimate} \to \texttt{Mark} \to \texttt{Refine}.$$

`Solve:` DG method

`Estimate:` DWR error estimate

`Mark:` top 10% of elements with largest error estimate

`Refine`:

- h-refine: split element into four, $h \to h/2$
- p-refine: increase poly degree by 1, $p \to p+1$
- e-refine: increase number of modes by 3, $N \to N+3$

Formulation

- Discretization
- Reduced basis element (RBE)
- Stability
- Error estimation and adaptation
- Related work

Related work

Viscous-inviscid coupling via thin-shear layer (TSL) Navier-Stokes

- Potential/Euler-TSL [Le Balleur 1978–; Drela & Giles 1987; . . .]

  $\neq$ mathematical reduction; error estimation wrt full equation

FEM with non-polynomial, special basis functions

- PUM, XFEM, GFEM, DEM

  [Babuška 1994–; Belytschko 1999–; Farhat 2001–; . . .]

  $\neq$ analytical functions vs empirical training

Reduced basis element methods

- RBE [Maday & Rønquist 2005–; . . .]
- Static-condensation RBE [Patera, Knezevic, & Huynh 2013–; . . .]
- Localized RB multiscale method [Ohlberger & Schindler 2015–]

Numerical results

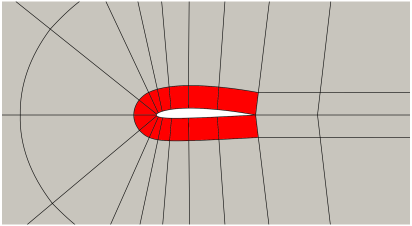
- RBE training
- Laminar airfoil
- Flat plate

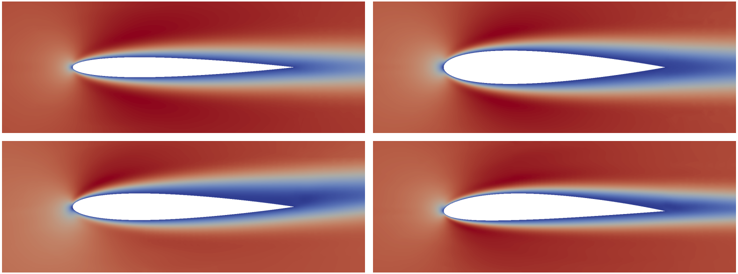
Numerical results

- RBE training
- Laminar airfoil
- Flat plate

Training cases: NACA 4-digit family

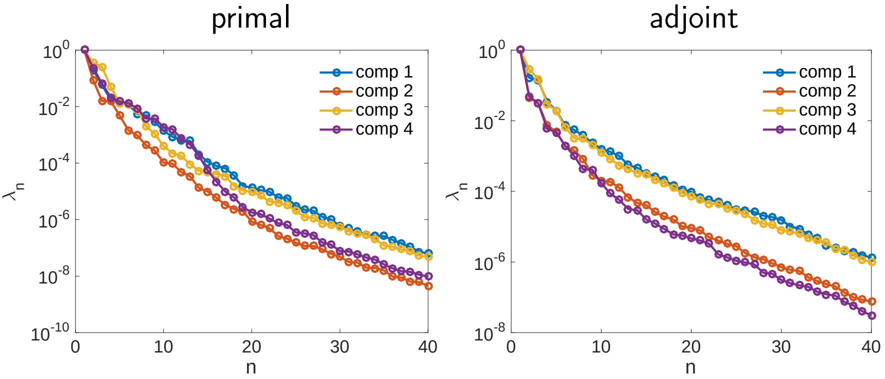
	Geom.	M_∞	Re_c	α
1	0009	0.4	1000	0°
2	0015	0.2	1000	0°
3	0012	0.2	1000	5°
4	2412	0.2	2000	0°
5	0012	0.2	500	0°



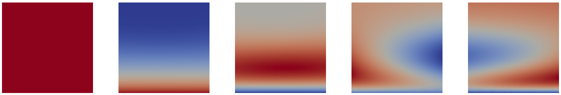


POD eigenvalues and modes

Eigenvalues:



Modes: primal, component 2



. . .

Numerical results

- RBE training
- Laminar airfoil
- Flat plate

Case 1. NACA 2410, $M_\infty = 0.5$, $Re_c = 3000$, $\alpha = 1^\circ$

Parameters outside of the training range

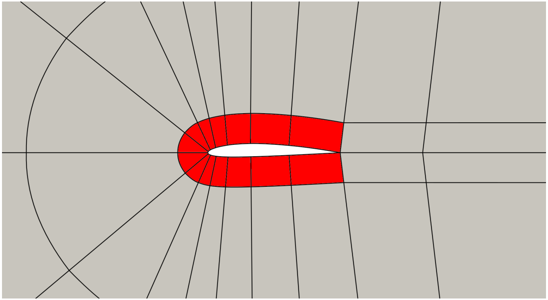
$M_\infty = 0.5 > 0.4$

$Re_c = 3000 > 2000$

We employ

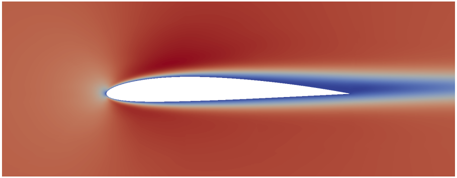
e-refinement for boundary layer RBEs

p-refinement for the rest.

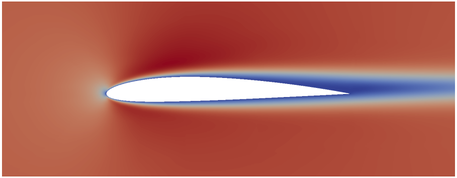


Case 1. NACA 2410, $M_\infty = 0.5$, $Re_c = 3000$, $\alpha = 1^\circ$

Target output: drag (at $\pm 1\%$ error level)



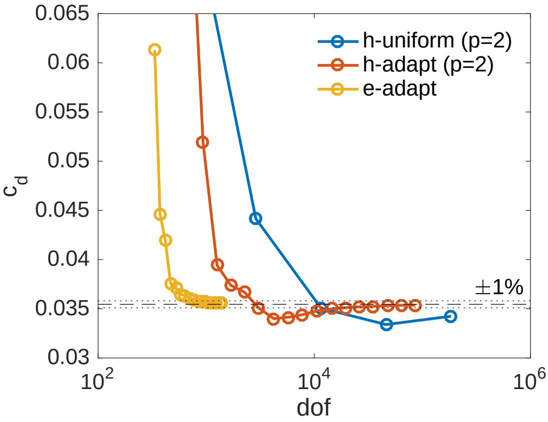
h-adapt: $c_d \approx 351.1 \pm 3.2$ counts ($\mathrm{dof} = 19485$)



e-adapt: $c_d \approx 356.9 \pm 3.0$ counts ($\mathrm{dof} = 972$)

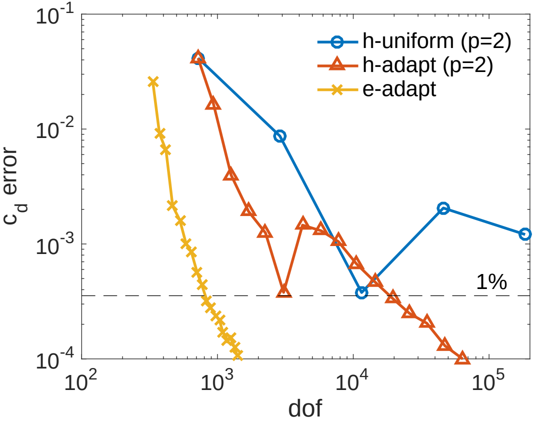
Case 1. Convergence

✔ rapid convergence



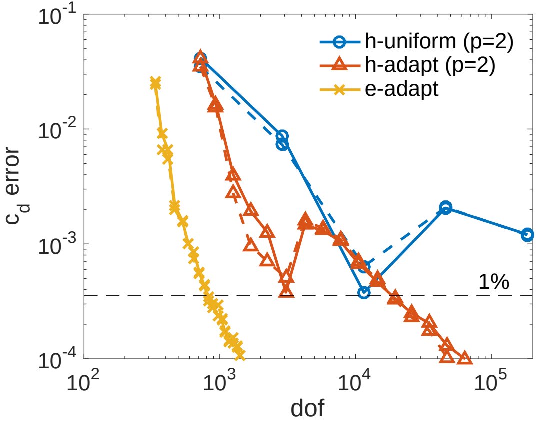
Case 1. Convergence

✔ rapid convergence

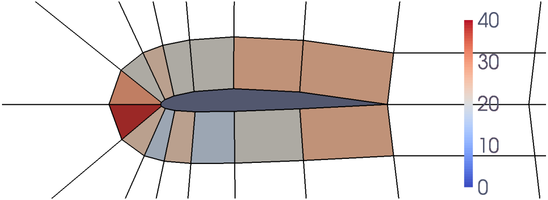


Case 1. Convergence & error estimate effectivity

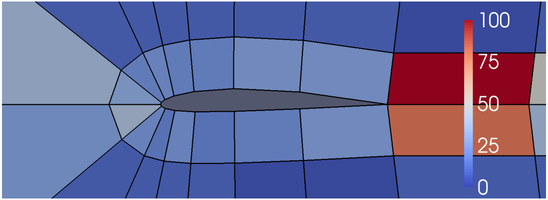
- ✔ rapid convergence
- ✔ reliable error estimate



RBE dof (range: 4–38)



RBE and polynomial dof (range: 4–100)



Next reducible feature: trailing edge singularity

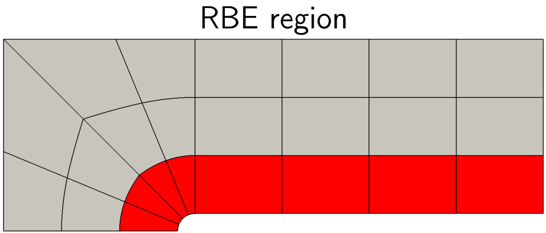
Numerical results

- RBE training
- Laminar airfoil
- Flat plate

Case 2. Flat plate, $M_\infty = 0.2$, $Re_c = 500$

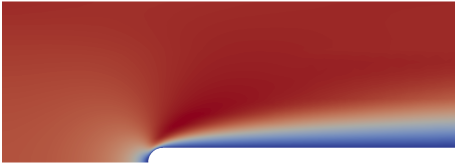
Use the same airfoil-trained boundary-layer RBEs.

Topologically different from an airfoil.

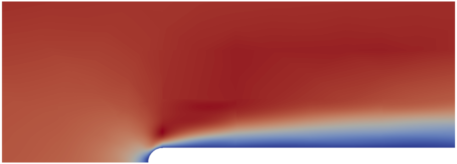


Case 2. Flat plate, $M_\infty = 0.2$, $Re_c = 500$

Target output: drag (at $\pm 0.1\%$ error level)



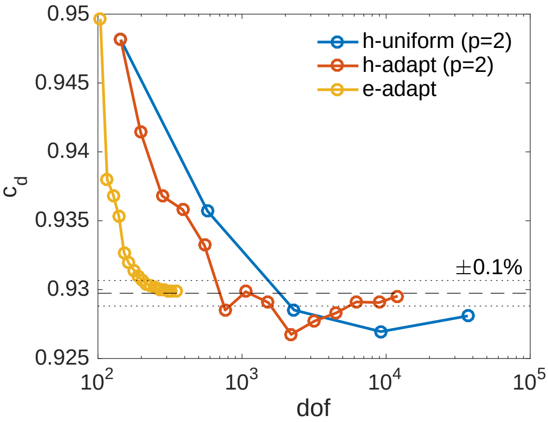
h-adapt: $c_d \approx 9294.2 \pm 4.3$ counts ($\mathrm{dof} = 5346$)



e-adapt: $c_d \approx 9306.5 \pm 7.7$ counts ($\mathrm{dof} = 204$)

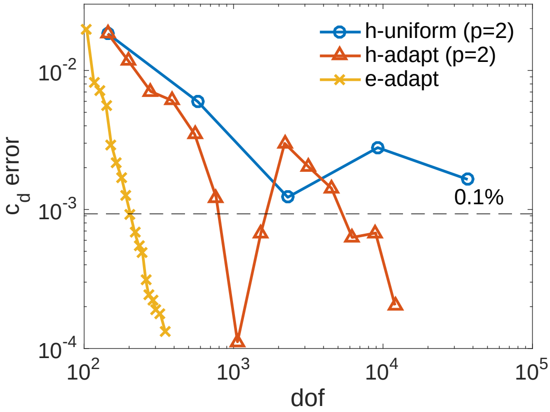
Case 2. Convergence

✔ rapid convergence



Case 2. Convergence

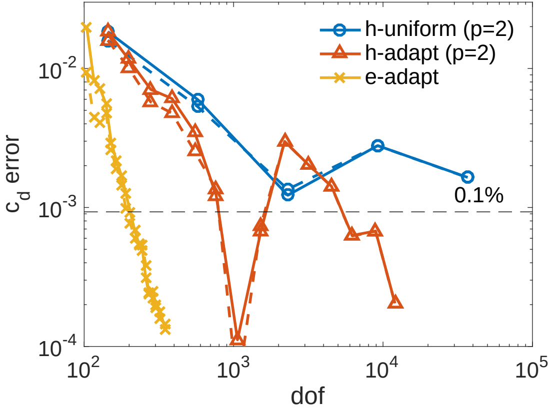
✔ rapid convergence



Case 2. Convergence & error estimate effectivity

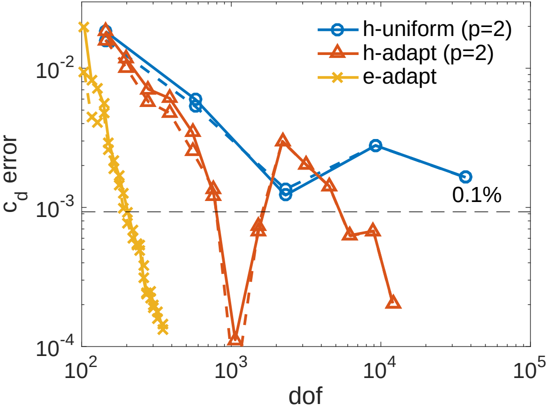
✔ rapid convergence

✔ reliable error estimate



Case 2. Convergence & error estimate effectivity

- ✔ rapid convergence
- ✔ reliable error estimate
- ✔ geometric flexibility (non-parametric & topology)



Summary

Summary

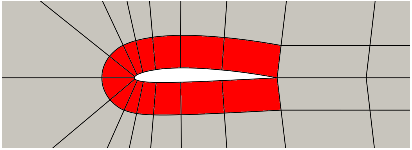
Idea: *forgo reduction of a specific case,*
and focus on reducible & reusable features (e.g. BLs)

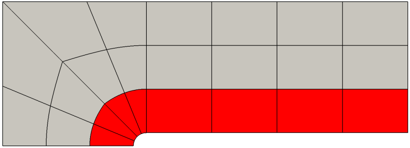
Ingredients:

DG + RBE + error estimate + adaptivity.

Result: reliable and flexible model reduction method
(though not as fast as case-specific ROMs)

Ongoing work: optimal basis rep. & quadrature;
RBE library & automated selection;
multi-parameter/nonlinear RBEs;





Backup

Implementation: basis functions and quadrature rules

Evaluation of $R_n(w_n, v_n)$ requires

1. means to evaluate basis functions
2. quadrature rule.

Current:

Basis: represented by high-order polynomials ($p \approx 15$)

Quadrature: high-order Gauss quadrature ($q \approx 50$)

Ongoing:

Basis: optimal h/p representation

Quadrature: specialized integration rules (e.g. magic points)