Adjoint-based adaptation for high-order discretizations of unsteady turbulent flow simulations

Krzysztof J. Fidkowski

AEROSPACE ENGINEERING

UNIVERSITY of MICHIGAN ■ COLLEGE of ENGINEERING

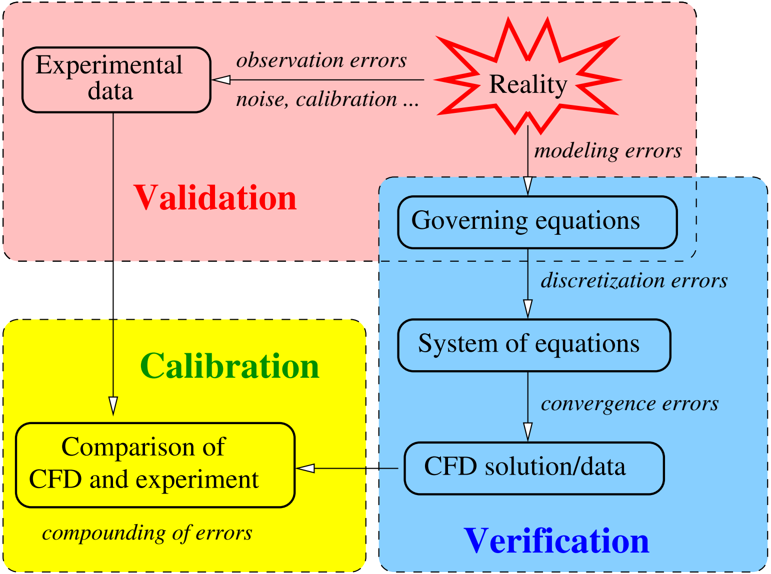
Workshop on New Trends in Numerical Methods for Hyperbolic Conservation Laws

Center for Computational & Applied Mathematics (CCAM)
Department of Mathematics, Purdue University

May 9, 2022

Outline

Many errors affect comparisons between CFD and experiments



We focus on controlling discretization errors

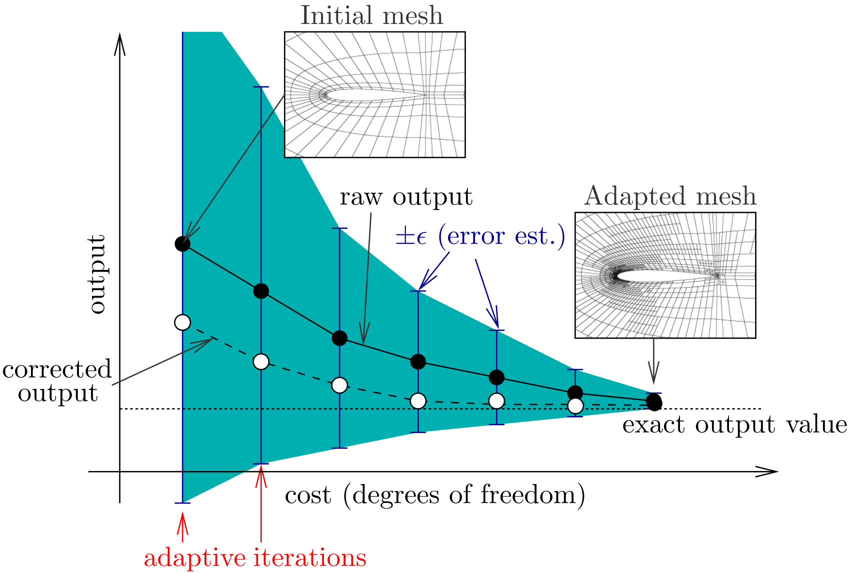
Error estimation

- Error estimates on outputs of interest are necessary for confidence in CFD results
- Mathematical theory exists for obtaining such estimates
- Recent works demonstrate the success of this theory for aerospace applications

Mesh adaptation

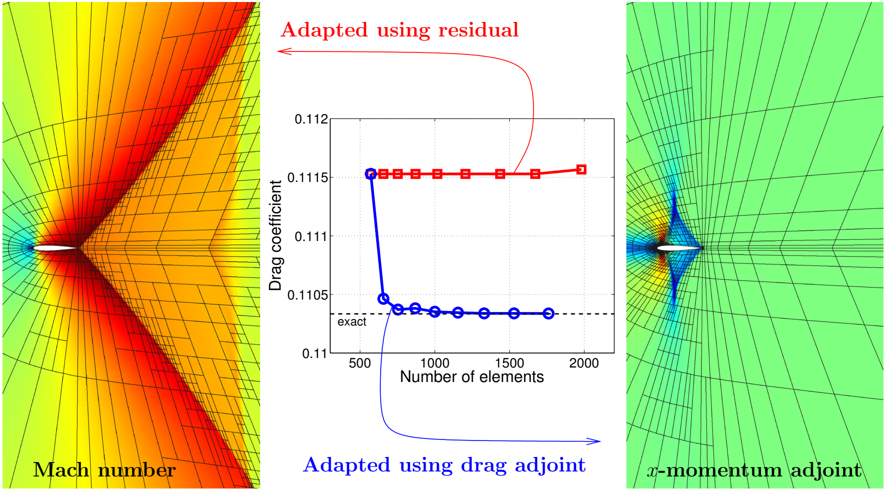
- Error estimation alone is not enough
- Engineering accuracy for complex aerospace simulations demands mesh adaptation to control numerical error
- Automated adaptation improves robustness by closing the loop in CFD analysis

A typical output-adaptive result



Why not just adapt "obvious" regions?

Fishtail shock in $M_\infty = 0.95$ inviscid flow over a NACA 0012 airfoil



Outline

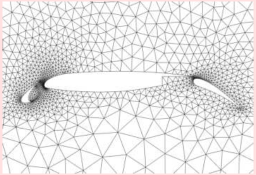
1. Introduction
2. Discretization
3. Error estimation and adaptation
4. Adaptation of unsteady turbulent flow

Computational fluid dynamics (CFD)

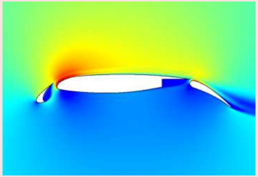
PDEs

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0$$
$$\frac{\partial (\rho \vec{v})}{\partial t} + \nabla \cdot (\rho \vec{v} \vec{v} + p \underline{\delta}) = 0$$
$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\rho \vec{v} H) = 0$$

discretization



solution



potential flow
Euler equations
Navier-Stokes
RANS
steady/unsteady
2D/3D

finite volume
finite difference
finite element
meshless methods
structured/unstructured
low/high order

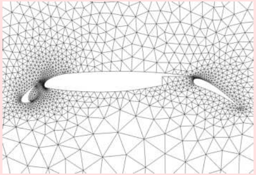
solver choices
HPC support
visualization
verification
validation
adaptation

Computational fluid dynamics (CFD)

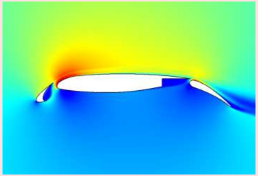
PDEs

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0$$
$$\frac{\partial (\rho \vec{v})}{\partial t} + \nabla \cdot (\rho \vec{v}\vec{v} + p\underline{\delta}) = 0$$
$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot (\rho \vec{v} H) = 0$$

discretization



solution



PDEs	discretization	solution
potential flow	finite volume	solver choices
Euler equations	finite difference	HPC support
Navier-Stokes	finite element	visualization
RANS	meshless methods	verification
steady/unsteady	structured/unstructured	validation
2D/3D	low/high order	adaptation

We use a high-order finite-element method

s conservation equations

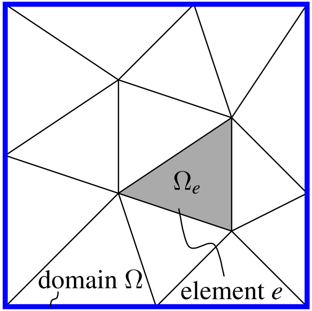
$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \vec{\mathbf{F}} + \mathbf{S} = \mathbf{0}$$

state approximation

$$\mathbf{u}_h(\vec{x}) \approx \sum_{e=1}^{N_e} \sum_{n=1}^{N_{pe}} \mathbf{U}_{en} \phi_{en}(\vec{x})$$

discrete primal problem

$$\frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = \mathbf{0}$$



N_e = # of elements

N_{pe} = # of basis functions on element e

$\phi_{en}(\vec{x})$ = n^{th} basis function of order p_e on e

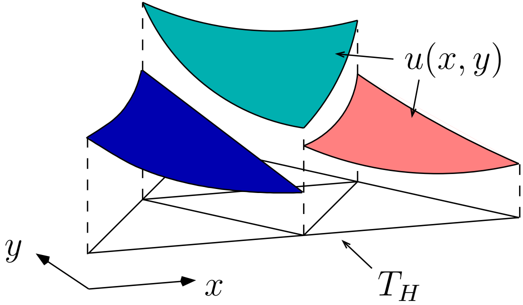
p_e = approximation order on element e

$\mathbf{U}_{en}$ = s coefficients on n^{th} basis function on element e

$\mathbf{U}$ = vector of all N primal unknowns

$\mathbf{R}$ = vector N discrete residuals

Discontinuous Galerkin solution approximation and weak form



$$\mathbf{u} \in \boldsymbol{\mathcal{V}}_h = [\mathcal{V}_h]^s$$
$$\mathcal{V}_h = \left\{ u \in L^2(\Omega) : u|_{\Omega_e} \in \mathcal{P}^p(\Omega_e)\ \forall\, \Omega_e \in T_h \right\}$$

Weak form:

$$\mathcal{R}_h(\mathbf{u}_h, \mathbf{v}_h) = \sum_{e=1}^{N_e} \mathcal{R}_h(\mathbf{u}_h, \mathbf{v}_h|_{\Omega_e}) = 0,\ \forall \mathbf{v}_h \in \boldsymbol{\mathcal{V}}_h,$$

The elemental contribution to the weak form is:

$$\begin{aligned}\mathcal{R}_h(\mathbf{u}_h, \mathbf{v}_h|_{\Omega_e}) = &\int_{\Omega_e} \mathbf{v}_h^T \partial_t \mathbf{u}_h \, d\Omega - \int_{\Omega_e} \partial_i \mathbf{v}_h^T \mathbf{F}_i \, d\Omega + \int_{\partial\Omega_e} \mathbf{v}_h^{+T} \widehat{\mathbf{F}} \, ds \\ &- \int_{\partial\Omega_e} \partial_i \mathbf{v}_h^{+T} \mathbf{K}_{ij}^{+} \left(\mathbf{u}_h^{+} - \widehat{\mathbf{u}}_h \right) n_j \, ds + \int_{\Omega_e} \mathbf{v}_h^T \mathbf{S} \, d\Omega\end{aligned}$$

Adjoint solutions let us calculate sensitivities efficiently

- Suppose N_μ parameters affect our PDE, but we only have one scalar output, $J(\mathbf{U})$:

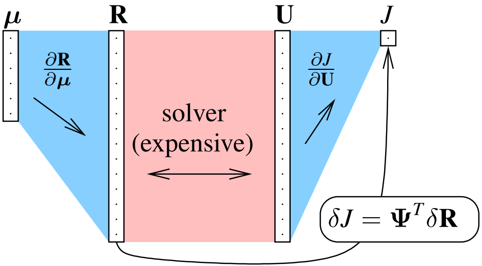
$$\underbrace{\boldsymbol{\mu}}_{\text{inputs} \in \mathbb{R}^{N_\mu}} \rightarrow \underbrace{\mathbf{R}(\mathbf{U},\boldsymbol{\mu}) = \mathbf{0}}_{N \text{ equations}} \rightarrow \underbrace{\mathbf{U}}_{\text{state} \in \mathbb{R}^N} \rightarrow \underbrace{J(\mathbf{U})}_{\text{output (scalar)}}$$

- We can efficiently compute sensitivities using a discrete adjoint vector, $\boldsymbol{\Psi} \in \mathbb{R}^N$,

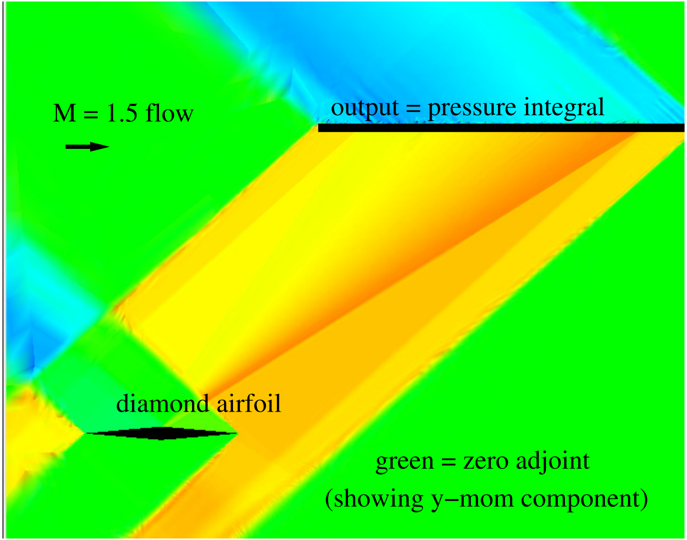
$$\frac{dJ}{d\boldsymbol{\mu}} = \boldsymbol{\Psi}^T \frac{\partial \mathbf{R}}{\partial \boldsymbol{\mu}}$$

- $\boldsymbol{\Psi}$ solves the linear *discrete adjoint equation*

$$\boxed{\left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right)^T \boldsymbol{\Psi} + \left(\frac{\partial J}{\partial \mathbf{U}}\right)^T = \mathbf{0}}$$

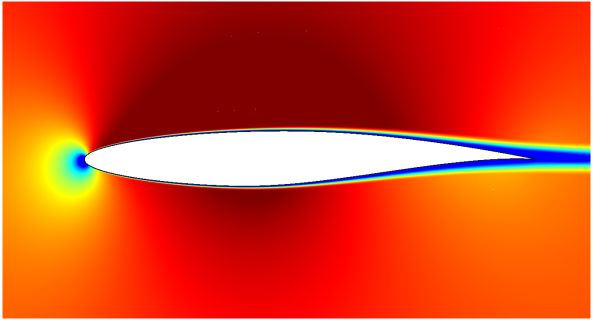


Sample steady adjoint solution

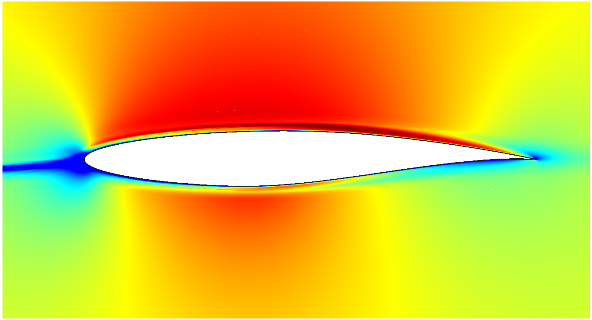


Another steady adjoint solution

RAE 2822, $M_\infty = 0.5$, $Re = 10^5$, $\alpha = 1^\circ$



x-momentum primal state

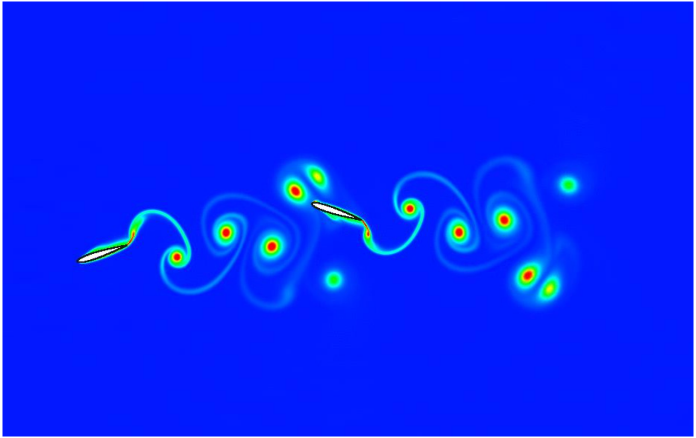


conservation of x-momentum drag adjoint

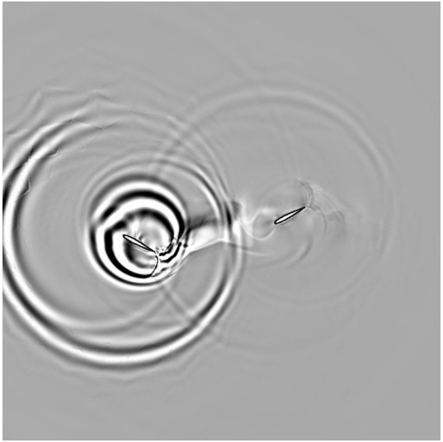
- The adjoint shares similar qualitative features with the primal
- Note a wake “reversal” in the adjoint solution

An unsteady adjoint solution

- Two NACA 0012 airfoils in pitch-plunge motion at $M_\infty = 0.3$, $Re = 1200$
- Output = lift on the aft airfoil at the end of the simulation



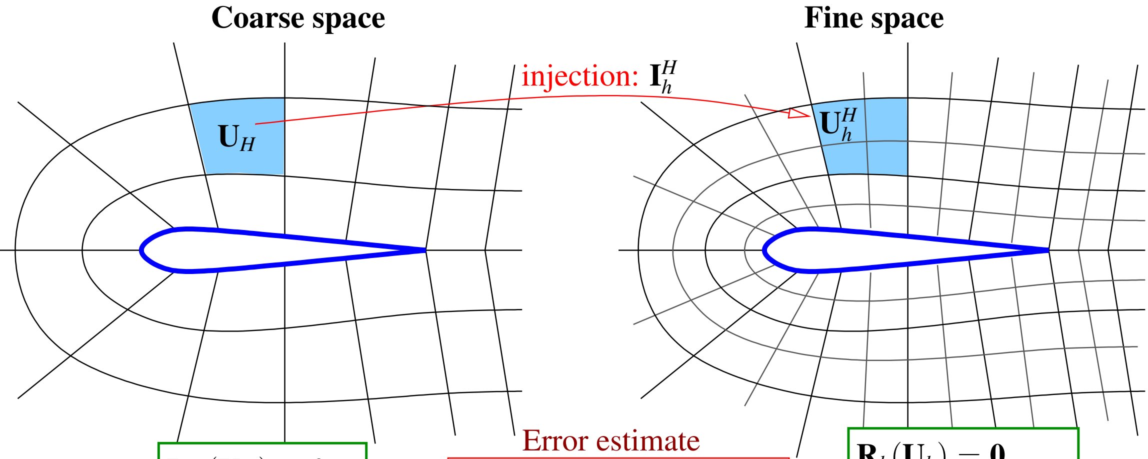
Primal solution (entropy)



Adjoint solution (GCL)

Outline

We estimate output errors relative to a fine space



$\mathbf{R}_H(\mathbf{U}_H) = \mathbf{0}$

output: $J_H(\mathbf{U}_H)$

Error estimate

$$\delta J = J_H - J_h \approx \mathbf{\Psi}_h^T \delta \mathbf{R}_h$$

$$\delta \mathbf{R}_h \equiv -\mathbf{R}_h(\mathbf{U}_h^H)$$

$\mathbf{R}_h(\mathbf{U}_h) = \mathbf{0}$

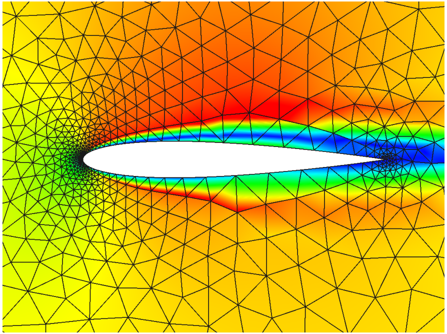
output: $J_h(\mathbf{U}_h)$

$\mathbf{R}_h(\mathbf{U}_h^H) \neq \mathbf{0}$

Example of output error estimation for an airfoil

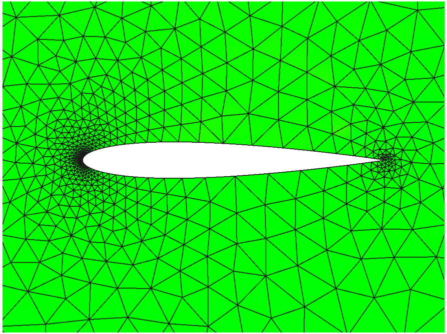
- A finer space (e.g. order enrichment) can uncover residuals in a converged solution
- Example: NACA 0012 at $\alpha = 2^\circ$ in $Re = 5000$, $M_\infty = 0.5$ flow

Coarse space state, $\mathbf{U}_H$



$p_H = 1$

Coarse space residual, $\mathbf{R}_H(\mathbf{U}_H)$

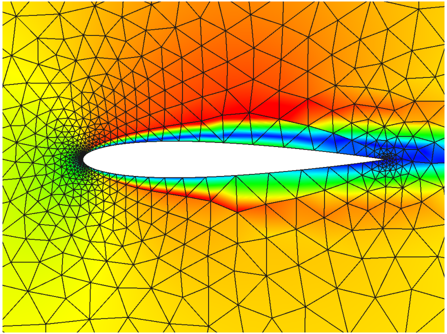


Zero as expected

Example of output error estimation for an airfoil

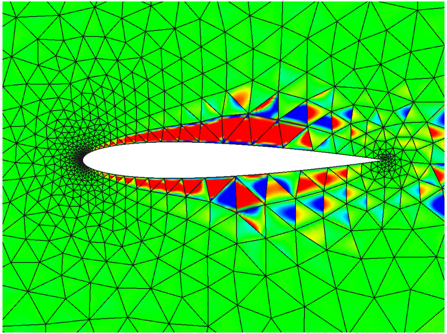
- A finer space (e.g. order enrichment) can uncover residuals in a converged solution
- Example: NACA 0012 at $\alpha = 2^\circ$ in $Re = 5000$, $M_\infty = 0.5$ flow

Injected state, $\mathbf{U}_h^H$



$p_h = 2$

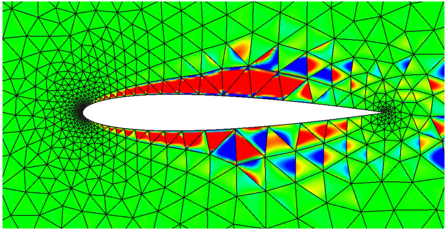
Fine space residual, $\mathbf{R}_h(\mathbf{U}_h^H)$



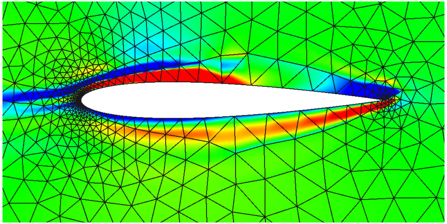
Nonzero: new info

Example of output error estimation for an airfoil

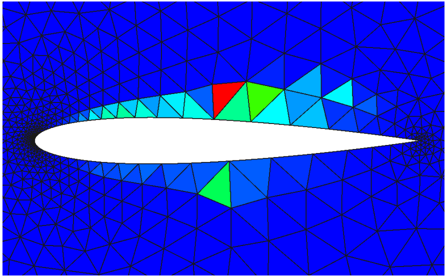
Fine space residual, $\mathbf{R}_h(\mathbf{U}_h^H)$



Fine space adjoint, $\mathbf{\Psi}_h$



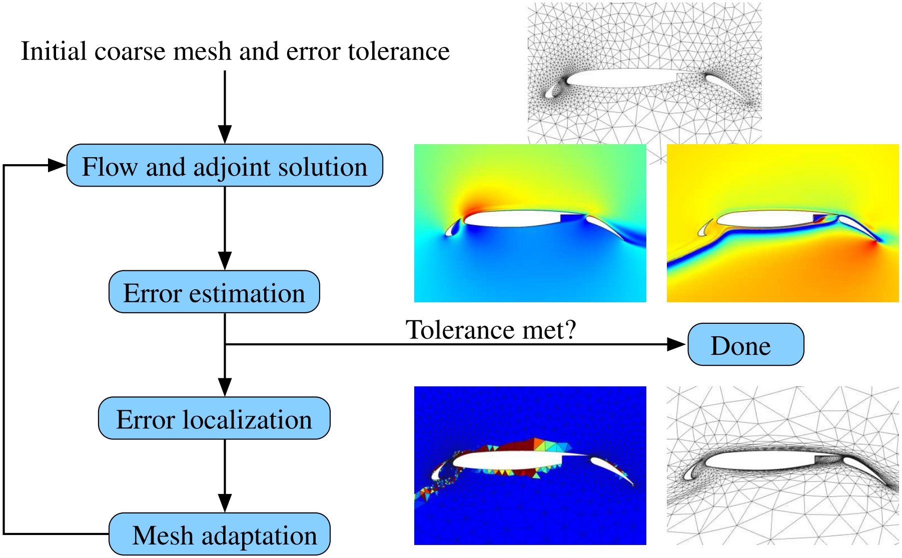
Error indicator, $\epsilon_e = |\mathbf{\Psi}_{h,e}^T \mathbf{R}_{h,e}(\mathbf{U}_h^H)|$



Output error: $\delta J \approx -\mathbf{\Psi}_h^T \mathbf{R}_h(\mathbf{U}_h^H)$

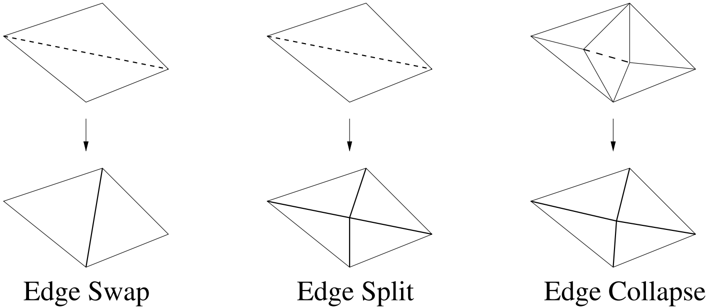
Idea: adapt where ϵ_e is high, to reduce the residual there

Mesh adaptation is an iterative process

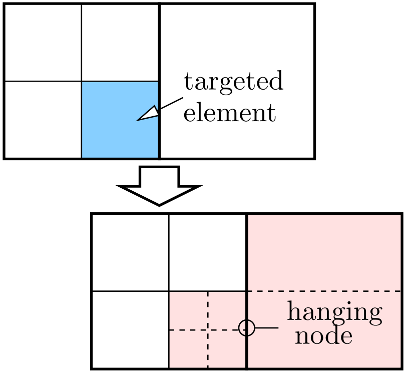


Adaptation can consist of local mesh modification

- As generating meshes is hard, we can modify the mesh incrementally
- Often more robust than global re-meshing
- Node movement can improve quality
- Hanging nodes easily supported in DG



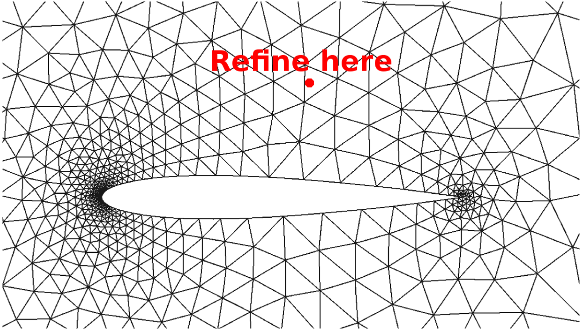
Unstructured local mesh operators



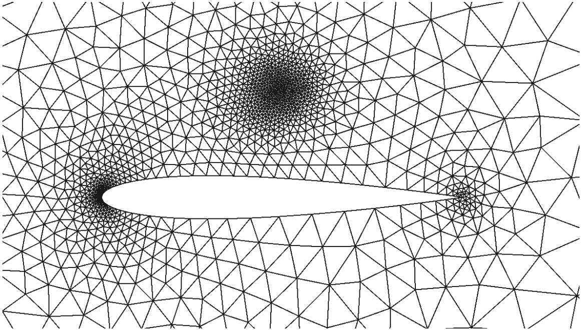
Hanging-node refinement

Or the entire mesh can be globally regenerated

- Make use of automated mesh generation software/scripts
- Current mesh is used to define a *Riemannian metric*
- Anisotropy can be incorporated/optimized (e.g. MOESS)
- Example of refinement near a single point:



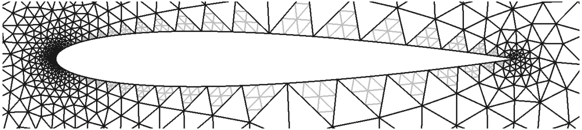
Original mesh



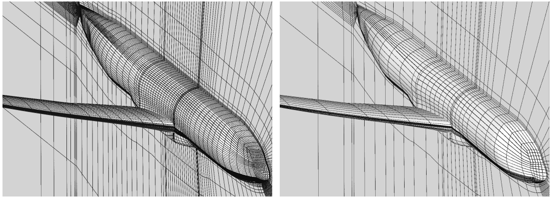
After refinement

Curved boundaries require special treatment at high order

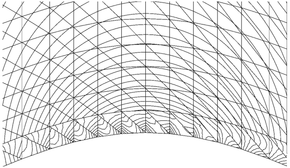
- DG needs an accurate representation of curved boundaries
- Curving elements is not easy
- Tangling is hard to avoid, especially in 3D anisotropic elements



First-layer curving (extend via elasticity)

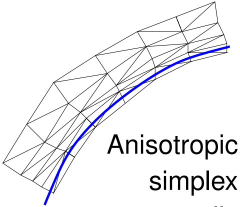


Agglomeration: linear $\rightarrow$ cubic elements



pressure contours

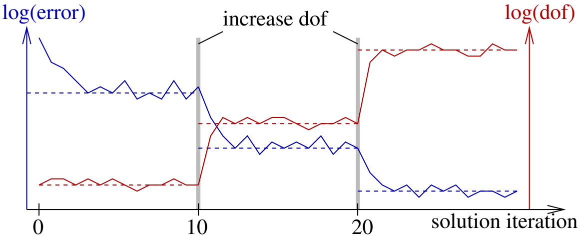
$p = 2$ Euler flow over a linear-element bump representation



Anisotropic simplex cut cells

A mesh optimization algorithm [Yano, 2012]

- Given: mesh, primal and adjoint
- Determine: mesh metric that minimizes output error at a fixed solution cost
- Key ingredients
  1. Error convergence model: metric $\rightarrow$ output error
  2. Cost model: metric $\rightarrow$ solution cost
  3. Iterative algorithm that equidistributes the marginal error-to-cost ratio
- Expect multiple iterations of optimization until error "bottoms out" at a fixed cost; can then increase allowable cost to further reduce error

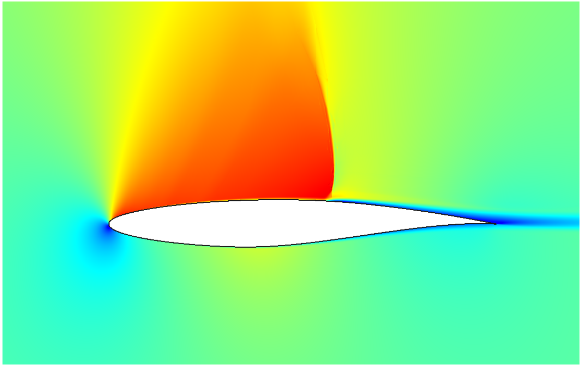


- Each iteration requires primal and adjoint solutions
- These are quick since starting from good initial guesses
- Use results from final run or average of last few runs

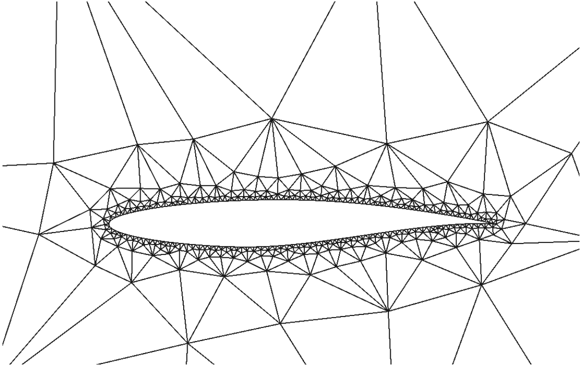
h-Adaptation Example: RAE 2822 in transonic flow

Spalart-Allmaras RANS, $M_\infty = 0.73, \alpha = 2.79^\circ, Re = 6.5M$

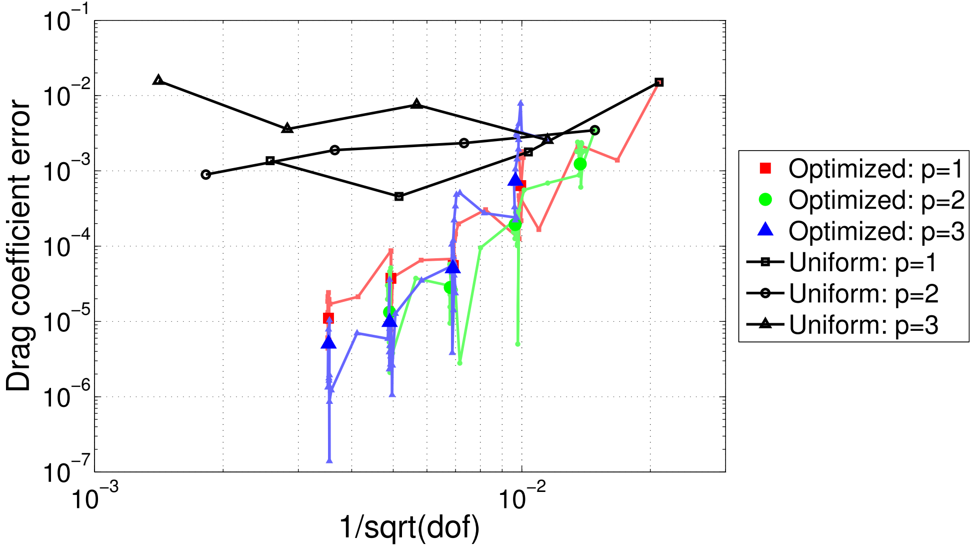
Mach number contours



Initial mesh: 758 triangles, farfield @$2000c$

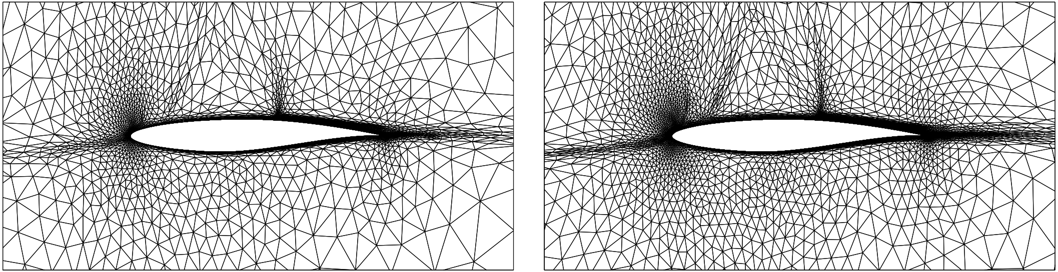


RAE 2822 in transonic flow: output convergence

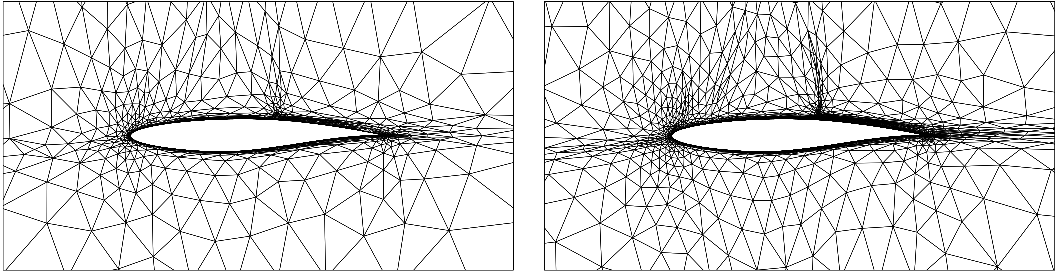


RAE 2822 in transonic flow: optimized meshes

$\mathbf{p = 1}$



$\mathbf{p = 2}$

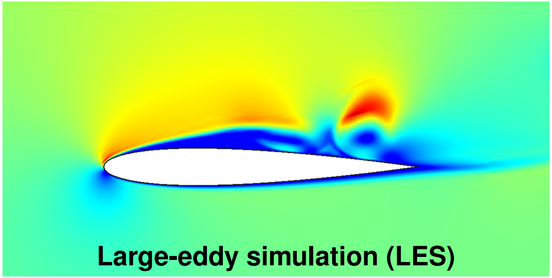


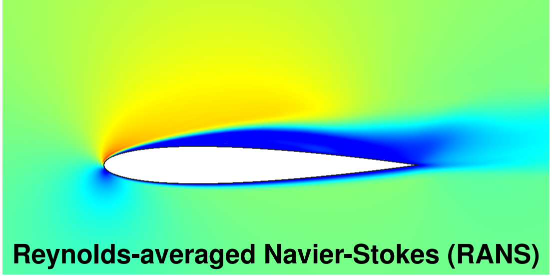
Outline

Simulations of turbulent flows

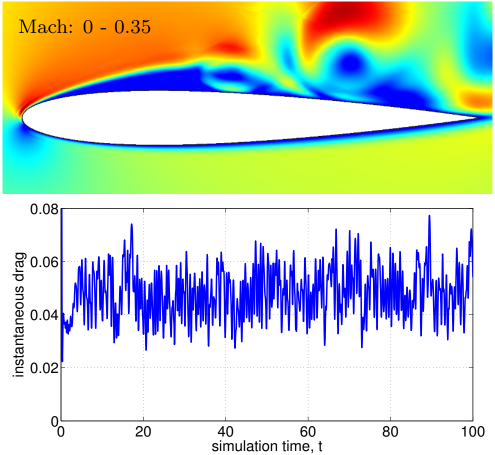
- **Turbulence** = random state fluctuations in space and time, caused by flow instabilities at high Reynolds numbers
- Simulating **unsteady turbulence** (DNS, LES, ...) is accurate but expensive
- **Modeling turbulence** via averaged quantities (RANS) is cheap but less accurate

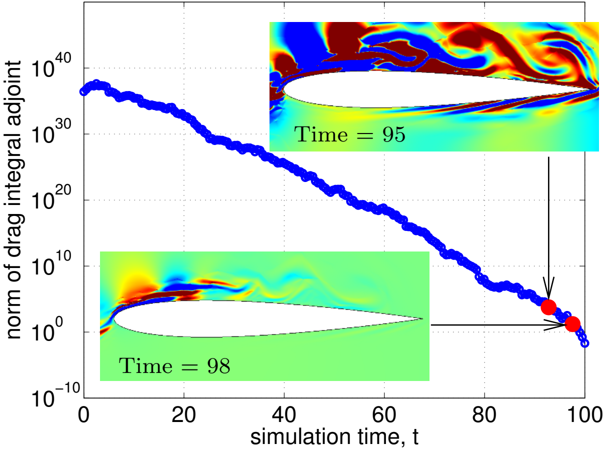
$$\text{e.g. Spalart-Allmaras 1-eqn. model: } \frac{D(\rho\tilde{\nu})}{Dt} = \underbrace{\mathcal{P}(\mathbf{u})}_{\text{production}} \underbrace{+\,\mathcal{T}(\mathbf{u})}_{\text{transport}} \underbrace{-\,\mathcal{D}(\mathbf{u})}_{\text{destruction}}$$





Unsteady adjoint-based gradient calculations are unstable





- Time-averaged quantities are still well-defined and interesting outputs
- However, we cannot directly use unsteady adjoints to compute their derivatives

Use a RANS model instead . . . with a correction

[Parish & Duraisamy, 2016]

- RANS by itself may not accurately predict the average state, $\bar{\mathbf{U}}$
- We can apply a correction field, $\beta(\vec{x})$, to the production term in the turbulence model,

$$\frac{D(\rho\tilde{\nu})}{Dt} = \underbrace{\boxed{\beta(\vec{x})}\mathcal{P}(\mathbf{u})}_{\text{production}} \underbrace{+\mathcal{T}(\mathbf{u})}_{\text{transport}} \underbrace{-\mathcal{D}(\mathbf{u})}_{\text{destruction}}$$

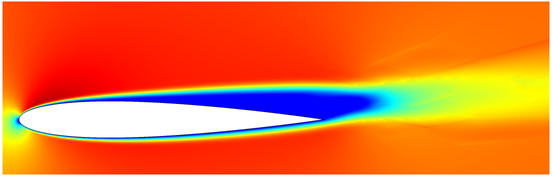
- Goal: minimize discrepancy between RANS and the unsteady solution, e.g.

$$\beta(\vec{x}) = \arg\min \mathcal{E}^2 = w_u \|\mathbf{U}_{\text{RANS}}(\beta(\vec{x})) - \bar{\mathbf{U}}\|^2 + \sum_{\text{outputs}} w_J (J(\mathbf{U}_{\text{RANS}}(\beta(\vec{x}))) - \bar{J})^2$$

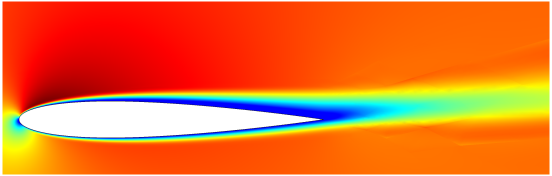
- Solve this optimization problem with a gradient-based method using RANS adjoints

Corrected RANS is much more accurate

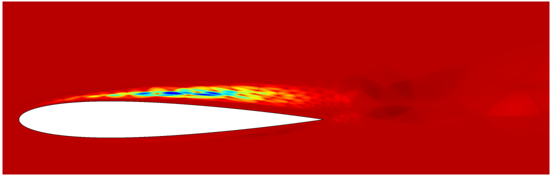
NACA 0012, $M = 0.2$, $\alpha = 5^\circ$, $Re = 10{,}000$



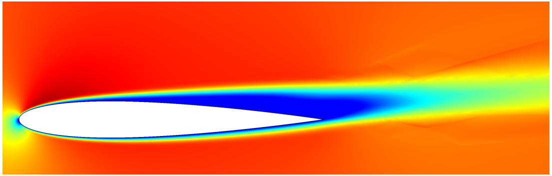
Time-averaged state, $\bar{\mathbf{U}}$



RANS without a correction, $\mathbf{U}_{\text{RANS}}$



Correction field, $\beta(\vec{x})$

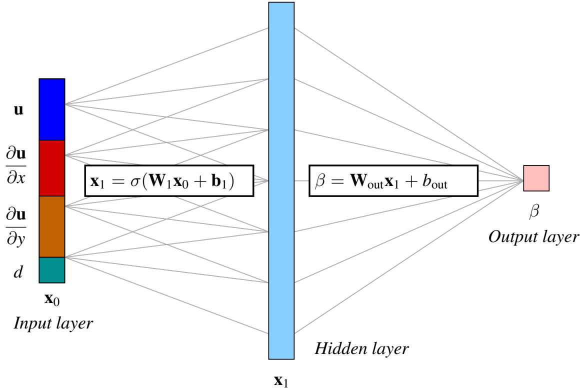


Corrected RANS, $\mathbf{U}_{\text{RANS}}(\beta(\vec{x}))$

We can obtain a model for β using machine learning

- $\beta(\vec{x})$ for one shape is not in a useful form for optimization
- We seek a model for β in terms of **local** flow quantities:

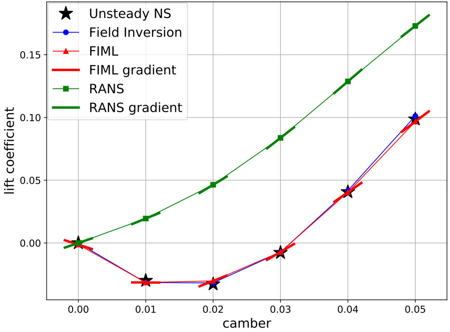
$$\beta \approx \underbrace{\beta_{ML}(\mathbf{u}, \nabla\mathbf{u}, d)}_{\text{NN model}} \quad (d = \text{ wall distance})$$

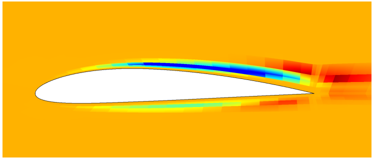


- Each quadrature point of each element yields one data point
- Small network: hidden layer $\mathbf{x}_1 \in \mathbb{R}^{30}$
- Adam optimizer, mini-batch size of $1{,}000$, and $500{,}000$ iterations
- Sigmoid activation function
- Once trained, deployed as part of the turbulence model
- Analytical linearization for derivatives

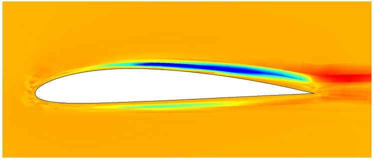
Example of FIML sensitivities: camber variation

- NACA X412 airfoils, $M = 0.2$, $Re = 10^4$
- FIML at each camber shape
- Interested in lift and its sensitivity with respect to camber variations
- Uncorrected RANS outputs severely off
- Correction with β: much more accurate outputs, and reasonable gradients

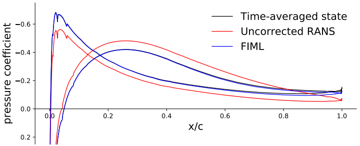




$\beta(\vec{x})$



$\beta(\mathbf{u}, \nabla \mathbf{u}, d)$



pressure coefficient

Error estimation for time-averaged outputs

Define a time-averaged output computed from the unsteady discrete solution, $\mathbf{U}(t)$,

$$\bar{J} \equiv \frac{1}{T_f - T_i} \int_{T_i}^{T_f} J(\mathbf{U}(t))\, dt$$

Suppose that we have an unsteady adjoint, $\mathbf{\Psi}(t)$, arising from the Lagrangian

$$\mathcal{L} \equiv \bar{J} + \frac{1}{T_f - T_i} \int_{T_i}^{T_f} \mathbf{\Psi}(t)^{\mathsf{T}} \mathbf{R}(\mathbf{U}(t))\, dt$$

The unsteady adjoint-weighted residual yields an estimate of the impact of a perturbation to the residual, $\delta\mathbf{R}(t)$, on the time-averaged output,

$$\delta\bar{J} \approx \frac{1}{T_f - T_i} \int_{T_i}^{T_f} \mathbf{\Psi}(t)^{\mathsf{T}} \delta\mathbf{R}(t)\, dt$$

Error estimation for time-averaged outputs (ctd.)

We decompose the unsteady adjoint and residual perturbation into time-averaged and time-varying components,

$$\boldsymbol{\Psi}(t) = \bar{\boldsymbol{\Psi}} + \boldsymbol{\Psi}'(t), \quad \delta\mathbf{R}(t) = \delta\bar{\mathbf{R}} + \delta\mathbf{R}'(t)$$

so that the error estimate becomes

$$\delta\bar{J} \approx \bar{\boldsymbol{\Psi}}^{\mathsf{T}}\delta\bar{\mathbf{R}} + \frac{1}{T_f - T_i}\int_{T_i}^{T_f} \boldsymbol{\Psi}'(t)^{\mathsf{T}}\delta\mathbf{R}'(t)\,dt$$

We assume that the first term dominates for chaotic flows, in which the time-varying components of the adjoint and residual perturbation are not strongly correlated

Augmented steady-state systems for unsteady simulations

Suppose that an augmented steady-state system models the unsteady problem,

$$\widetilde{\mathbf{R}}(\widetilde{\mathbf{U}}) = \mathbf{0},$$

where $\widetilde{\mathbf{U}} \in \mathbb{R}^{\widetilde{N}}$ is the augmented state vector, with $\widetilde{N} \geq N$, and $\bar{\mathbf{U}} \approx \mathbf{I}^r\widetilde{\mathbf{U}}$. The augmented residual (e.g. for RANS) is

$$\widetilde{\mathbf{R}}(\widetilde{\mathbf{U}}) = \begin{bmatrix} \mathbf{R}(\mathbf{I}^r\widetilde{\mathbf{U}}) + \mathbf{R}^{\mathrm{aug}}(\widetilde{\mathbf{U}}) \\ \widetilde{\mathbf{R}}^{\mathrm{aug}}(\widetilde{\mathbf{U}}), \end{bmatrix},$$

where $\mathbf{R}^{\mathrm{aug}}$ is an additive change to the original residuals, and $\widetilde{\mathbf{R}}^{\mathrm{aug}}$ is the set of new residuals (e.g. associated with the eddy-viscosity equation).

Error estimation using the augmented system

Denote by $\mathbf{U}_h^H(t) = \mathbf{I}_h^H \mathbf{U}_H(t)$ the prolongation of the coarse unsteady state into a fine space. Similarly, $\widetilde{\mathbf{U}}_h^H$ is the prolongation of $\widetilde{\mathbf{U}}_H$. Evaluating the fine-space residuals yields

$$\begin{aligned}
\delta \mathbf{R}_h(t) &= -\mathbf{R}_h(\mathbf{U}_h^H(t)) + \mathbf{M}_h \mathbf{I}_h^H \mathbf{M}_H^{-1} \mathbf{R}_H(\mathbf{U}_H(t)) \\
\delta \widetilde{\mathbf{R}}_h &= -\widetilde{\mathbf{R}}_h(\widetilde{\mathbf{U}}_h^H)
\end{aligned}$$

We use the augmented-system adjoint, $\widetilde{\boldsymbol{\Psi}}_h$, to define two error estimates:

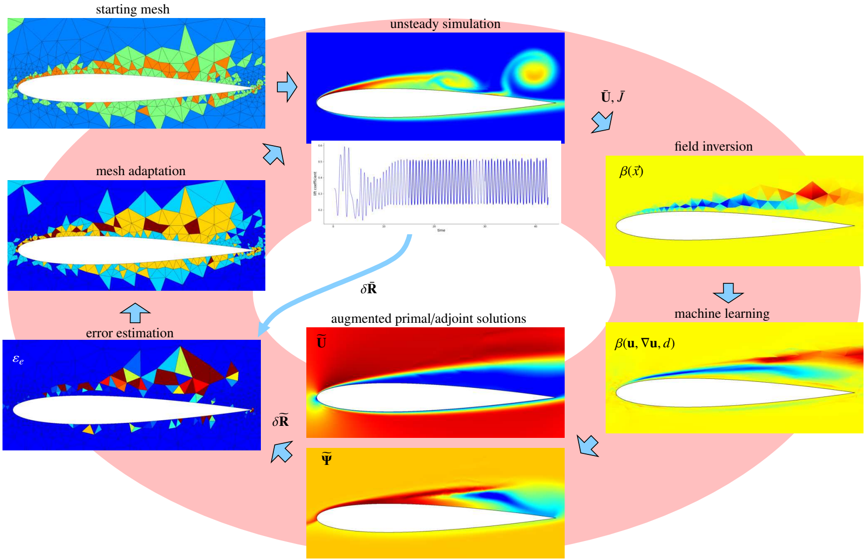
E1. Time-averaged unsteady residual weighted by the FIML adjoint:

$$\delta \bar{J}_h \approx \delta \widetilde{\boldsymbol{\Psi}}_h^{\mathsf{T}} \delta \bar{\mathbf{R}}_h, \qquad \delta \bar{\mathbf{R}}_h \equiv \frac{1}{T_f - T_i} \int_{T_i}^{T_f} \delta \mathbf{R}_h(t)\, dt$$

E2. FIML residual weighted by the FIML adjoint:

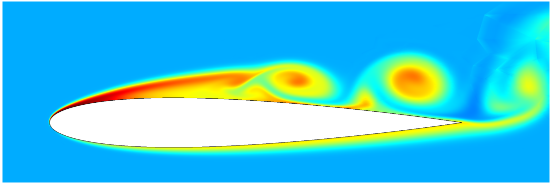
$$\delta \widetilde{J}_h \approx \delta \widetilde{\boldsymbol{\Psi}}_h^{\mathsf{T}} \delta \widetilde{\mathbf{R}}_h$$

Unsteady mesh adaptation iterations

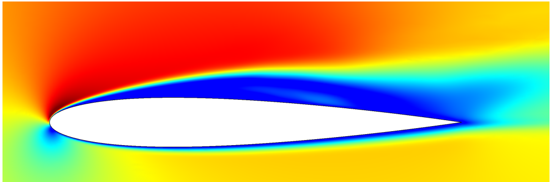


FIML adaptation example

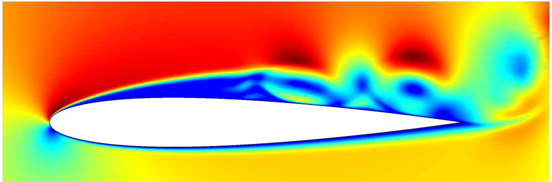
NACA 0012, $M_\infty = 0.2$, $Re = 10{,}000$, $\alpha = 7^\circ$



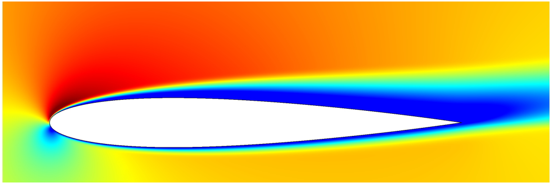
Instantaneous entropy



Time-averaged Mach



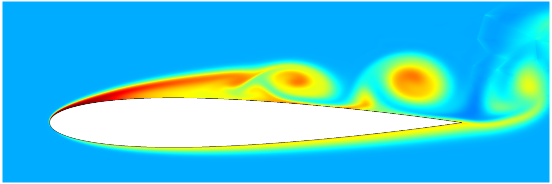
Instantaneous Mach



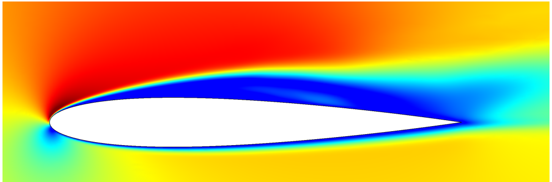
RANS Mach

FIML adaptation example

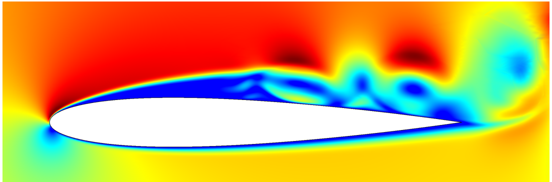
NACA 0012, $M_\infty = 0.2$, $Re = 10{,}000$, $\alpha = 7^\circ$



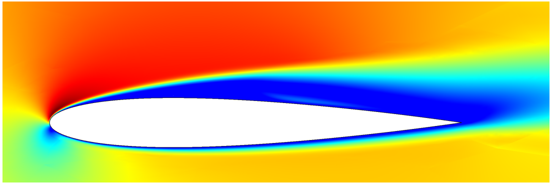
Instantaneous entropy



Time-averaged Mach

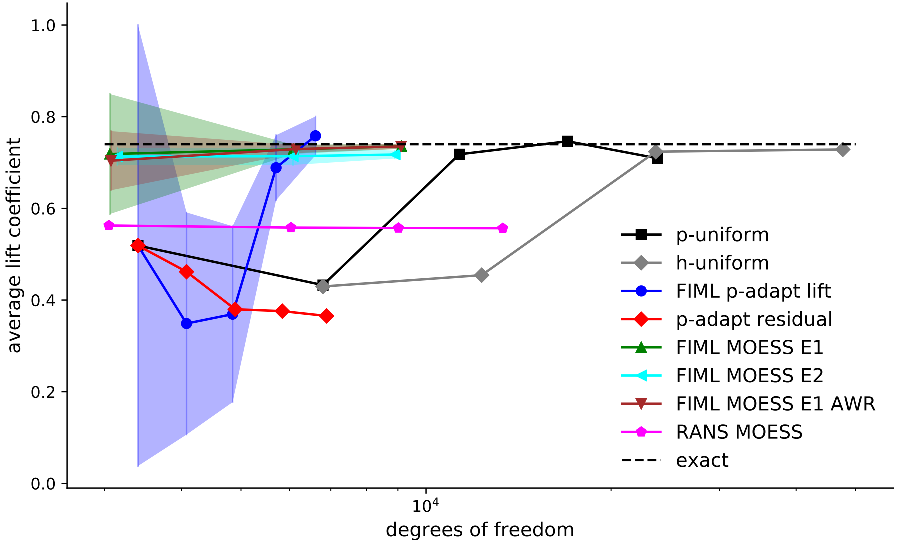


Instantaneous Mach

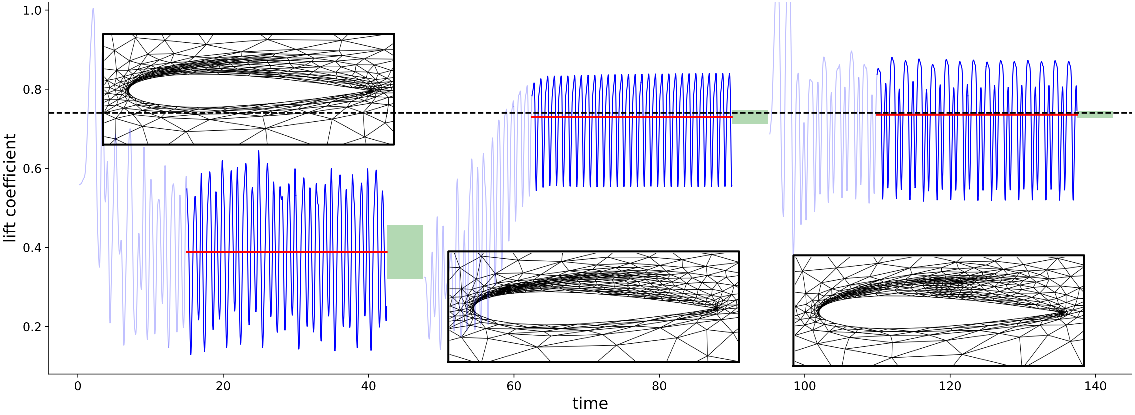


FIML Mach

Lift coefficient convergence

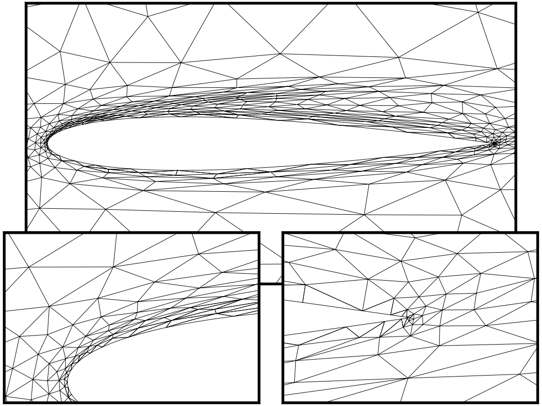


FIML MOESS unsteady adaptation history

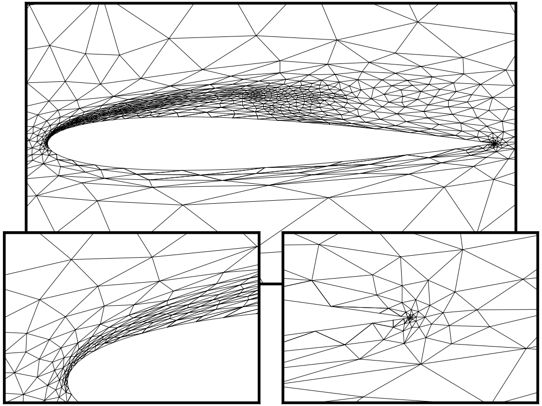


Red lines = time-averaged output **Green bars** = output-error estimates

Comparison of adapted meshes at 9000 dof



RANS MOESS

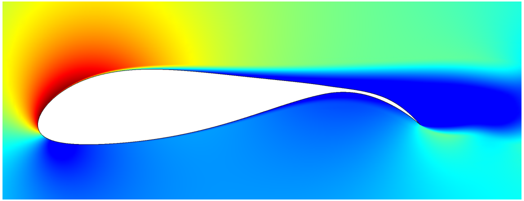


FIML MOESS

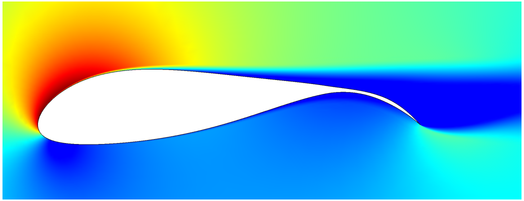
Different mesh size and anisotropy due to different residuals and adjoints in AWR

A higher Reynolds-number example

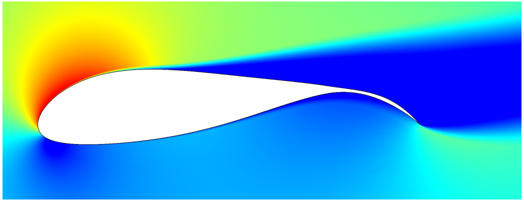
High-lift airfoil, $M_\infty = 0.2$, $Re = 1M$, $\alpha = 17^\circ$



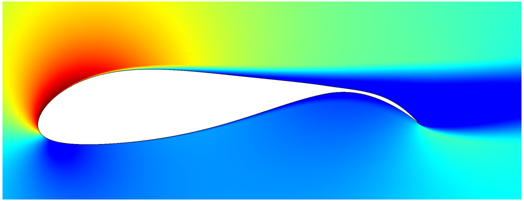
Instantaneous entropy



Time-averaged Mach

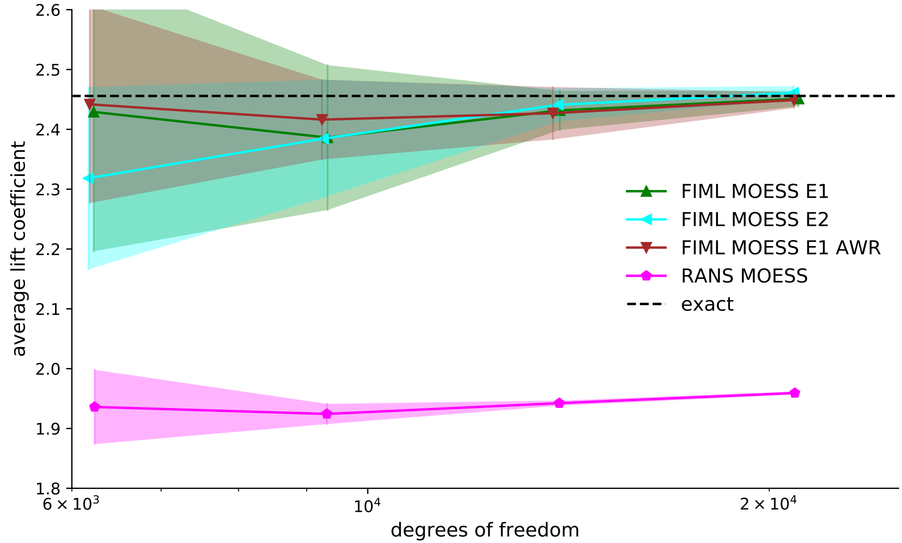


RANS Mach

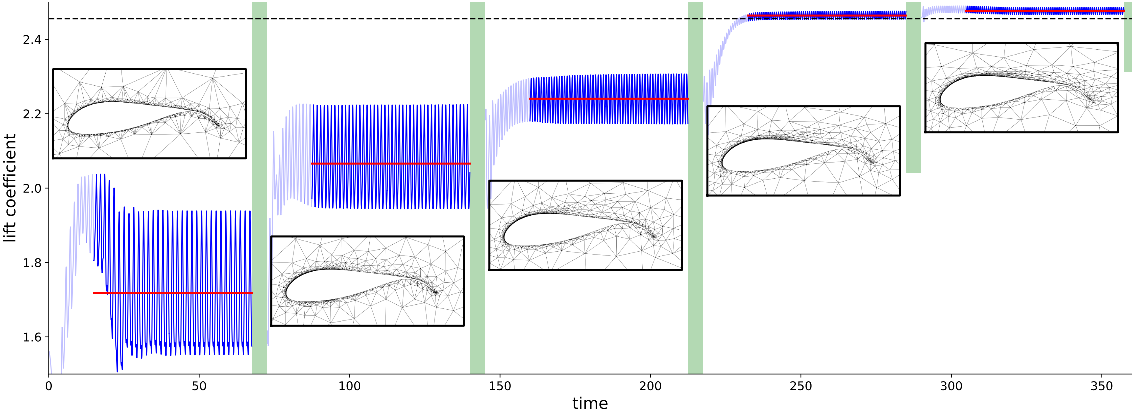


FIML Mach

Lift coefficient convergence

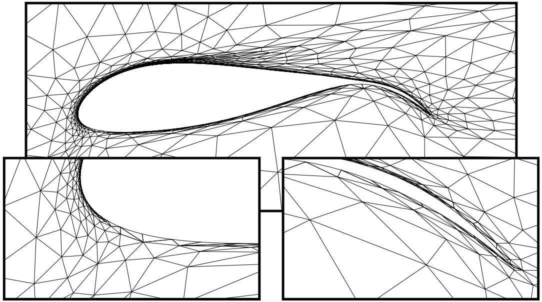


FIML MOESS unsteady adaptation history

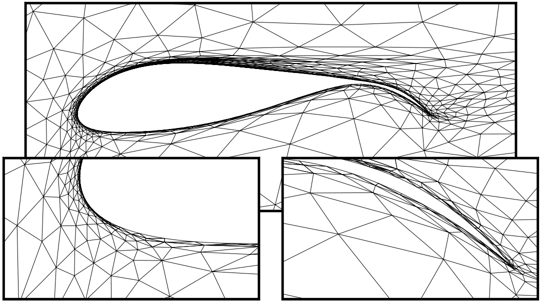


Red lines = time-averaged output **Green bars** = output-error estimates

Comparison of adapted meshes at 13500 dof



RANS MOESS



FIML MOESS

Different mesh size and anisotropy due to different residuals and adjoints in AWR

Summary and conclusions

- CFD results are polluted by various error sources
- Improving CFD accuracy requires attention to models, meshes, and solvers
- Adjoint-based methods work well for steady and deterministic unsteady problems
- Chaotic problems preclude direct adjoint solutions, and regularization techniques are expensive
- Steady augmented models offer an attractive alternative when interested in statistically-steady outputs
- We have demonstrated output-based error estimation and mesh adaptation of unsteady turbulent flow using tuned/trained steady augmented models