Progress in Mesh-Adaptive Discontinuous Galerkin Methods for CFD

German Aerospace Center Seminar

Krzysztof Fidkowski
Department of Aerospace Engineering
The University of Michigan

May 4, 2009

Outline

1. Introduction
2. Output Error Estimation
   - Discrete Adjoint Solutions
   - Entropy Adjoint Connection
3. Mesh Generation and Adaptation
   - Hanging Node Refinement
   - Simplex Cut Cells
4. Results

Outline

1. Introduction
2. Output Error Estimation
   - Discrete Adjoint Solutions
   - Entropy Adjoint Connection
3. Mesh Generation and Adaptation
   - Hanging Node Refinement
   - Simplex Cut Cells
4. Results

Introduction

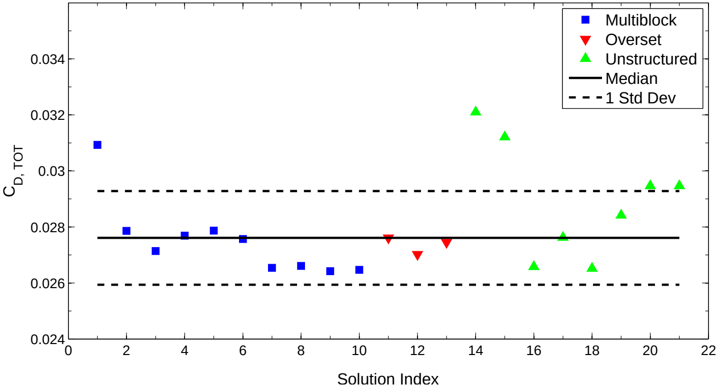
Complex CFD simulations made possible by

- Increasing computational power
- Improvements in numerical algorithms

New liability: ensuring accuracy of computations

- Management by expert practitioners is not feasible for increasingly-complex flowfields
- Reliance on best-practice guidelines is an open-loop solution: numerical error unchecked for novel configurations
- Output calculations are not yet sufficiently robust, even on relatively standard simulations

AIAA Drag Prediction Workshop III



Drag coefficient predictions for the DLR-F6 wing-body at $M = 0.75$, $C_L = 0.5$, $Re = 5 \times 10^6$.

- Variation of 25 drag counts: 1 drag count $\approx$ 4 passengers for a large transport aircraft (ADIGMA goal: 10 counts)
- Only slight improvement over results from previous two workshops

Improving CFD Robustness

Error estimation

- "Error bars" on outputs of interest are necessary for confidence in CFD results
- Mathematical theory exists for obtaining such error bars
- 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

Outline

Outline

1. Introduction
2. Output Error Estimation
   - Discrete Adjoint Solutions
   - Entropy Adjoint Connection
3. Mesh Generation and Adaptation
   - Hanging Node Refinement
   - Simplex Cut Cells
4. Results

Discrete Adjoint Definition

Consider N_h algebraic equations and an output,

$$\mathbf{R}_h(\mathbf{u}_h) = 0, \qquad J_h = J_h(\mathbf{u}_h)$$

- $\mathbf{u}_h \in \mathbb{R}^{N_h}$ is the vector of unknowns
- $\mathbf{R}_h \in \mathbb{R}^{N_h}$ is the vector of residuals
- $J_h(\mathbf{u}_h)$ is a *scalar* output of interest

The discrete output adjoint vector, $\psi_h \in \mathbb{R}^{N_h}$, is the sensitivity of J_h to an infinitesimal residual perturbation, $\delta\mathbf{R}_h \in \mathbb{R}^{N_h}$,

$$\delta J_h \equiv \psi_h^T \delta\mathbf{R}_h$$

Discrete Adjoint Equation

The linearized perturbed equations are:

$$\frac{\partial \mathbf{R}_h}{\partial \mathbf{u}_h}\delta \mathbf{u}_h + \delta \mathbf{R}_h = 0,$$

Also linearizing the output we have,

$$\delta J_h = \underbrace{\frac{\partial J_h}{\partial \mathbf{u}_h}\delta \mathbf{u}_h = \psi_h^T \delta \mathbf{R}_h}_{\text{adjoint definition}} \overbrace{= -\psi_h^T \frac{\partial \mathbf{R}_h}{\partial \mathbf{u}_h}\delta \mathbf{u}_h}^{\text{linearized equations}}$$

Requiring the above to hold for arbitrary perturbations yields the linear *discrete adjoint equation*

$$\left(\frac{\partial \mathbf{R}_h}{\partial \mathbf{u}_h}\right)^T \psi_h + \left(\frac{\partial J_h}{\partial \mathbf{u}_h}\right)^T = 0$$

Variational Adjoint Definition

Galerkin weighted residual statement: determine $\mathbf{u}_h \in \mathcal{V}_h$ such that

$$\mathcal{R}_h(\mathbf{u}_h, \mathbf{v}_h) = 0, \qquad \forall \mathbf{v}_h \in \mathcal{V}_h$$

- $\mathcal{V}_h$ is a finite-dimensional space of functions
- $\mathcal{R}_h(\cdot,\cdot) : \mathcal{V}_h \times \mathcal{V}_h \to \mathbb{R}$ is a semilinear form
- $\mathcal{J}_h(\mathbf{u}_h) : \mathcal{V}_h \to \mathbb{R}$ is a scalar output

The output adjoint is a function, $\psi_h \in \mathcal{V}_h$, that is the sensitivity of $\mathcal{J}_h$ to a residual perturbation, $\delta\mathbf{r}$:

$$\delta\mathcal{J}_h \equiv (\delta\mathbf{r}_h, \psi_h)$$

where $(\cdot,\cdot) : \mathcal{V}_h \times \mathcal{V}_h \to \mathbb{R}$ is a suitable inner product

Variational Adjoint Statement

The Fréchét-linearized equations are:

$$\mathcal{R}'_h[\mathbf{u}_h](\delta\mathbf{u}_h, \mathbf{v}_h) + (\delta\mathbf{r}_h, \mathbf{v}_h) = 0, \qquad \forall \mathbf{v}_h \in \mathcal{V}_h,$$

Also linearizing the output we have,

$$\delta\mathcal{J}_h = \underbrace{\mathcal{J}'_h[\mathbf{u}_h](\delta\mathbf{u}_h) = (\delta\mathbf{r}_h, \psi_h)}_{\text{adjoint definition}} = -\mathcal{R}'_h[\mathbf{u}_h](\delta\mathbf{u}_h, \psi_h)$$

with an overbrace labelled "linearized equations" spanning $\overbrace{(\delta\mathbf{r}_h, \psi_h) = -\mathcal{R}'_h[\mathbf{u}_h](\delta\mathbf{u}_h, \psi_h)}^{\text{linearized equations}}$

Requiring the above to hold for arbitrary perturbations yields the linear *variational adjoint statement*: find $\psi_h \in \mathcal{V}_h$ such that

$$\mathcal{R}'_h[\mathbf{u}_h](\mathbf{v}_h, \psi_h) + \mathcal{J}'_h[\mathbf{u}_h](\mathbf{v}_h) = 0, \qquad \forall \mathbf{v}_h \in \mathcal{V}_h$$

Continuous Adjoint

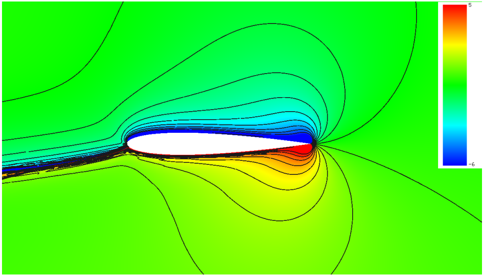
The continuous primal solution, $\mathbf{u} \in \mathcal{V}$, satisfies

$$\mathcal{R}(\mathbf{u}, \mathbf{v}) = 0, \qquad \forall \mathbf{v} \in \mathcal{V},$$

The continuous adjoint solution, $\psi \in \mathcal{V}$, satisfies

$$\mathcal{R}'[\mathbf{u}](\mathbf{v}, \psi) + \mathcal{J}'[\mathbf{u}](\mathbf{v}) = 0, \qquad \forall \mathbf{v} \in \mathcal{V}$$

- $\mathcal{V}$ is an infinite-dimensional space
- ψ is a Green's function relating source residuals to output perturbations [Giles and Pierce, 1997]



x-momentum lift adjoint, $M_\infty = 0.4$, $\alpha = 5^o$

Output Error Estimation

Output error: difference between an output computed with the discrete system solution and that computed with the exact solution

Output error estimation techniques

- Identify all areas of the domain that are important for the accurate prediction of an output
- Account for propagation effects inherent to hyperbolic problems
- Require solution of an adjoint equation

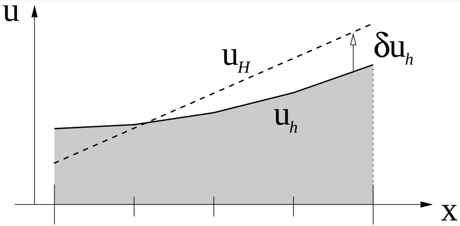
Output error estimates can be used to:

- Ascribe confidence levels to engineering outputs in the presence of numerical errors
- Drive an adaptive method to reduce the output error below a user-specified tolerance

Adjoint-Weighted Residual Method

Goal: Calculate $\mathcal{J}_H(\mathbf{u}_H) - \mathcal{J}_h(\mathbf{u}_h)$ = output error *estimate*

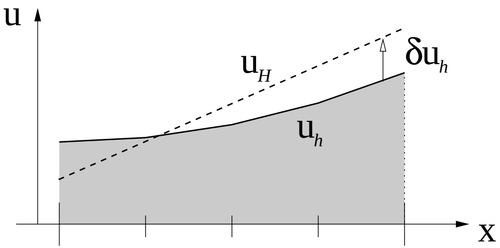
- $\mathbf{u}_H \in \mathcal{V}_H$ = *coarse* solution
- $\mathbf{u}_h \in \mathcal{V}_h$ = *fine* solution



- Could solve for $\mathbf{u}_h$ and recompute the output – expensive and not directly useful for adaptation
- Idea: $\mathbf{u}_H$ generally does not satisfy the fine-level equations. That is, $\mathcal{R}_h(\mathbf{u}_H, \mathbf{v}_h) \neq 0$. Instead, $\mathbf{u}_H$ solves: find $\mathbf{u}_h' \in \mathcal{V}_h$ such that

$$\mathcal{R}_h(\mathbf{u}_h', \mathbf{v}_h) - \mathcal{R}_h(\mathbf{u}_H, \mathbf{v}_h) = 0 \quad \forall \mathbf{v}_h \in \mathcal{V}_h$$

Adjoint-Weighted Residual Method (ctd.)



- $-\mathcal{R}_h(\mathbf{u}_H, \mathbf{v}_h)$ is a residual perturbation on the fine discretization
- Suppose we have an adjoint solution on the fine mesh: $\psi_h \in \mathcal{V}_h$
- The adjoint lets us calculate the output perturbation from the point of view of the fine discretization:

$$\delta \mathcal{J}_h = \mathcal{J}_h(\mathbf{u}_H) - \mathcal{J}_h(\mathbf{u}_h) \approx -\mathcal{R}_h(\mathbf{u}_H, \psi_h)$$

[Becker and Rannacher, 1996; Giles *et al* , 1997]

Approximating ψ_h

How do we calculate ψ_h = the adjoint on the fine discretization?

Options:

1. Solve for $\mathbf{u}_h$ and then ψ_h – expensive! Potentially still useful to drive adaptation. [Solín and Demkowicz, 2004; Hartmann *et al*]
2. Solve for $\psi_H \in \mathcal{V}_H$ = the adjoint on the coarse discretization:

$$\mathcal{R}'_H[\mathbf{u}_H](\mathbf{v}_H, \psi_H) + \mathcal{J}'_H[\mathbf{u}_H](\mathbf{v}_H) = 0, \qquad \forall \mathbf{v}_H \in \mathcal{V}_H,$$

   1. Reconstruct ψ_H on the fine discretization using a higher-accuracy stencil. Smoothness assumption on adjoint.
      [Rannacher, 2001; Barth and Larson, 2002; Venditti and Darmofal 2002; Lu, 2005; Fidkowski and Darmofal, 2007]
   2. Initialize ψ_h with ψ_H and take a few iterative solution steps on the fine discretization.
      [Barter and Darmofal, 2008; Oliver and Darmofal, 2008]

Outline

Entropy Adjoint Connection

Collaborative work with P. L. Roe

- Adjoint-based output error estimation is "state of the art" but it
  - requires solution of an adjoint problem for each output
  - targets only requested outputs
- Currently investigating connection between **entropy variables** and adjoint solutions in order to derive an adaptive indicator that
  - does not require solution of an adjoint problem
  - produces an "overall good" solution

For a conservation law of the form

$$\mathbf{A}_i \partial_i \mathbf{u} = 0$$

the entropy variables, **v**, satisfy an adjoint equation:

$$\mathbf{A}_i^T \partial_i \mathbf{v} = 0$$

Entropy Adjoint Connection

- The entropy variables are readily computable from the state **u**:

$$\mathbf{v} = U_{\mathbf{u}}^T = \left[\frac{\gamma - S}{\gamma - 1} - \frac{1}{2}\frac{\rho V^2}{p}, \; \frac{\rho u_i}{p}, \; -\frac{\rho}{p} \right]^T,$$

- The output associated with the entropy variable adjoint is

$$J = \int_{\Omega} F_i n_i ds$$

where $F_i(U)$ is the entropy flux, and U is the entropy function:

$$U = -\rho S/(\gamma - 1), \quad S = \ln p - \gamma \ln \rho,$$

- J measures the net entropy generation in the domain
- The analysis extends to Navier-Stokes
- Idea: use **v** as an adjoint solution in output error estimation
  - Targets areas where entropy generation is not predicted well
  - Does not require solution of an adjoint problem

Outline

Mesh Generation and Adaptation

- Mesh generation is often the most challenging and time-consuming aspect of CFD
- Curved boundary representation required by discontinuous Galerkin (DG) makes mesh generation even more difficult
- Adaptation generally requires robust mesh generation

Pursuing two approaches:

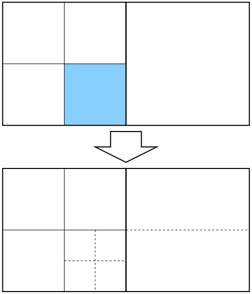
1. Hanging node adaptation of quadrilateral and hexahedral meshes – primarily for error indicator testing purposes
2. Cut-cell mesh generation – as a long term solution

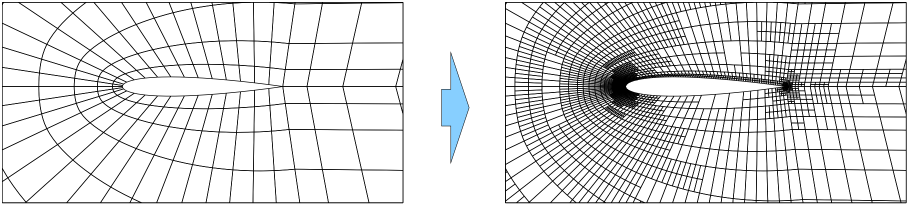
Outline

1. Introduction
2. Output Error Estimation
   - Discrete Adjoint Solutions
   - Entropy Adjoint Connection
3. Mesh Generation and Adaptation
   - Hanging Node Refinement
   - Simplex Cut Cells
4. Results

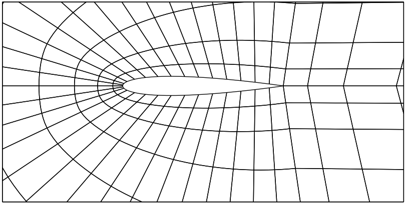
Hanging Node Refinement

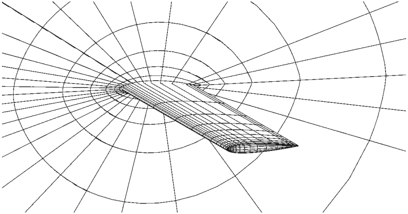
- Straightforward implementation
- No change to DG solution space
- No re-meshing or geometry callback is necessary with suitable initial curved mesh
- Available in 2D and 3D





Quad/Hex Mesh Generation

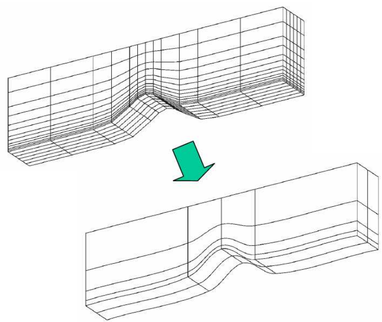


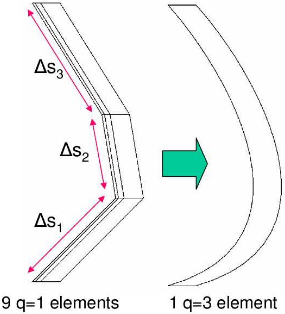


- An initial quad or hex mesh is required for hanging-node adaptation
- For accurate high-order computation, need a curved boundary representation
- Approach
    - Leverage existing structured multiblock capability to generate high-order geometry meshes
    - Goal: Provide initial meshes for testing adaptive indicators
    - Not a long-term solution: ultimately complement with cut-cell capability

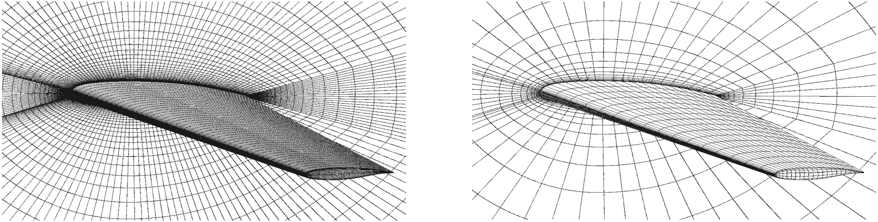
Linear Multiblock to Curved Meshes

- Idea: generate a finer mesh than required and agglomerate elements
- Fineness of linear mesh depends on the desired order, q
- Non-corner linear nodes provide high-order geometry information
- Implemented a conversion utility with automated agglomeration



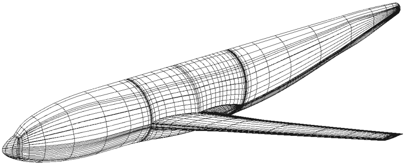


Have hook into ANSYS ICEM CFD



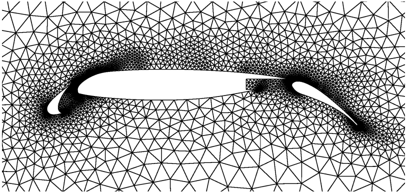
Top: DPW-W1 medium
$q = 1$ and $q = 3$ meshes

Right: DLR-F6 with fairing,
coarse mesh, $q = 3$ mesh

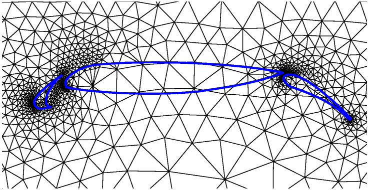


Outline

What Are Cut Cells?



Boundary-conforming mesh



Simplex cut-cell mesh

Features

- Cut-cell meshes do not conform to geometry boundary
- Solution only exists inside the computational domain
- Premise: metric-driven meshing of a simple convex volume (e.g. box) is straightforward
- Simplex cut cell meshes can be adapted anisotropically in any direction

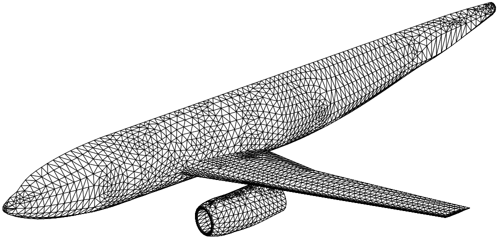
Geometry Representation

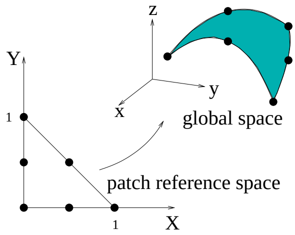
2D: Cubic splines

- Efficient treatment of curved boundaries; slope & curvature continuity

3D: Quadratic patches

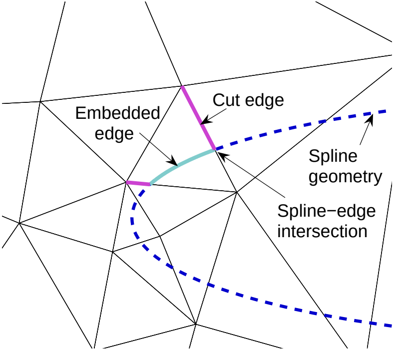
- Patch surface ($\mathbf{x}$) given analytically: $\mathbf{x} = \sum_j \phi(\mathbf{X})_j \mathbf{x}_j$, where $\mathbf{X} = [X, Y]$ are patch ref space coords, and $\mathbf{x} = [x, y, z]$ are global coords
- Water-tight representation (no holes)

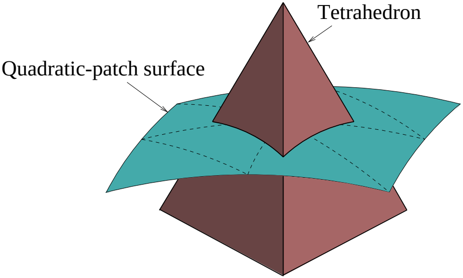




Intersection Problem

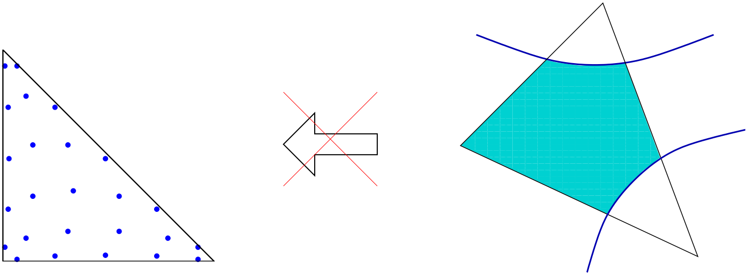
- 2D: cubic-equation for spline/edge intersection
- 3D: conic-section algorithm for patch/plane intersection
- Multiply-cut elements treated as separate cut cells
- Elements completely inside geometry removed from mesh structure





Integration

- High-order finite element method requires integration over:
  - **Element boundaries** (edges in 2D, faces in 3D)
  - **Element interiors** (areas in 2D, volumes in 3D)
- Regular triangles and tetrahedra can be mapped to reference elements, where optimal integration rules exist
- These rules do not (in general) apply to cut cells, where areas and volumes are of irregular shape

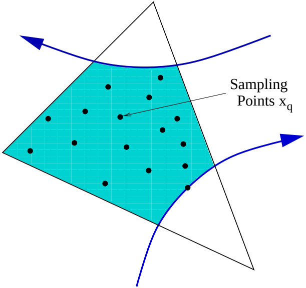


Area Integration

Goal

Sampling points, $\mathbf{x}_q$, and weights, w_q for integrating arbitrary $f(\mathbf{x})$ to a desired order:

$$\int_\kappa f(\mathbf{x})d\mathbf{x} \approx \sum_q w_q f(\mathbf{x}_q)$$



Key Idea

Project $f(\mathbf{x})$ onto space of high-order basis functions, $\zeta_{\mathbf{i}}(\mathbf{x})$:

$$f(\mathbf{x}) \approx \sum_{\mathbf{i}} F_{\mathbf{i}}\zeta_{\mathbf{i}}(\mathbf{x})$$

Choose $\zeta_{\mathbf{i}}(\mathbf{x})$ to allow for simple computation of $\int_\kappa \zeta_{\mathbf{i}}(\mathbf{x})d\mathbf{x}$.

Area Integration (ctd.)

Set $\zeta_{\mathbf{i}} \equiv \nabla \cdot \mathbf{G}_{\mathbf{i}}$ and use the divergence theorem:

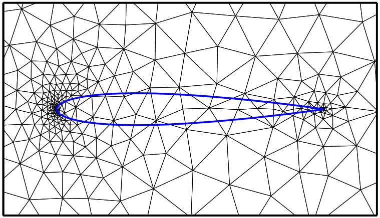
$$\int_{\kappa} \zeta_{\mathbf{i}} d\mathbf{x} = \int_{\kappa} \nabla \cdot \mathbf{G}_{\mathbf{i}} d\mathbf{x} = \int_{\partial\kappa} \mathbf{G}_{\mathbf{i}} \cdot \mathbf{n} ds$$

- $\mathbf{G}_{\mathbf{i}}$ = a standard high-order basis (e.g. tensor product)
- Line integrals over $\partial\kappa$ using 1D edge formulas

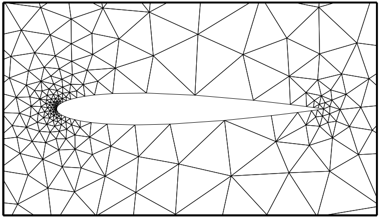
- Projection $f(\mathbf{x}) \approx \sum_{\mathbf{i}} F_{\mathbf{i}} \zeta_{\mathbf{i}}(\mathbf{x})$ minimizes the least-squares error at randomly-chosen sampling points, $\mathbf{x}_q$, inside the cut cell
- QR factorization, $\zeta_{\mathbf{i}}(\mathbf{x}_q) = Q_{q\mathbf{j}} R_{\mathbf{ji}}$, and integration over κ leads to an expression for the quadrature weights:

$$\int_{\kappa} f(\mathbf{x}) d\mathbf{x} \approx \sum_{\mathbf{i}} F_{\mathbf{i}} \int_{\kappa} \zeta_{\mathbf{i}}(\mathbf{x}) d\mathbf{x} = \sum_{q} f(\mathbf{x}_q) \underbrace{Q_{q\mathbf{j}} (R^{-T})_{\mathbf{ji}} \int_{\kappa} \zeta_{\mathbf{i}}(\mathbf{x}) d\mathbf{x}}_{w_q}$$

Example: 2D Flow Solution



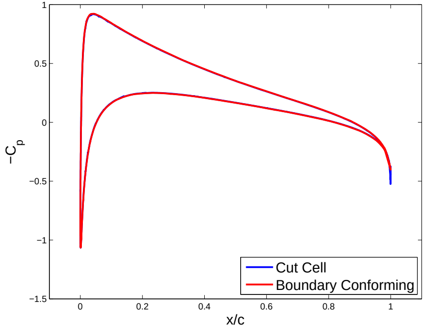
Cut-cell mesh



Boundary-conforming mesh

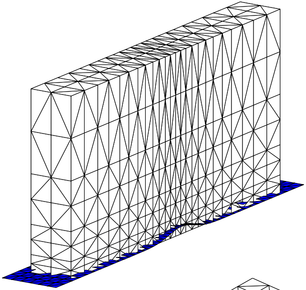
- $M = 0.5, \alpha = 3^o$
- $p = 2$ interpolation

C_p comparison

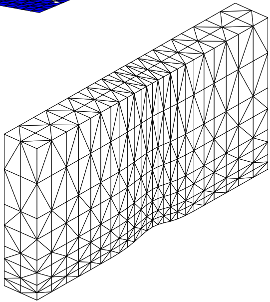


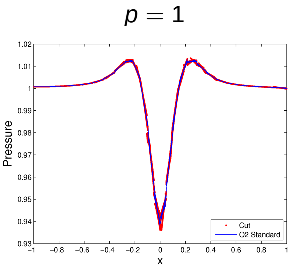
Example: 3D Flow Solution

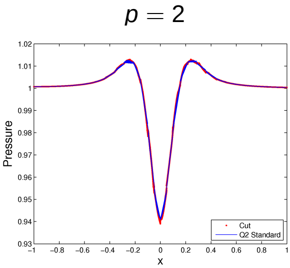
Cut-cell mesh:



Boundary-conforming mesh:



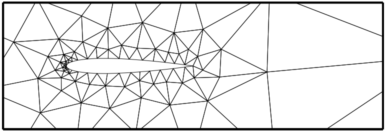




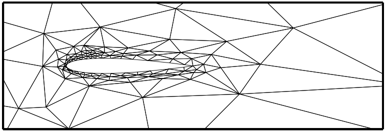
Metric-Driven Adaptation

Idea: refine elements with high error; coarsen elements with low error

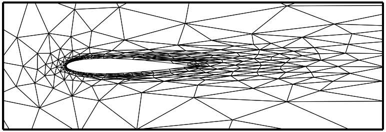
Iteration 0



Iteration 2



Iteration 4



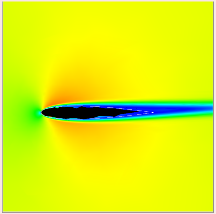
- Use *a priori* output error estimate to relate element error to size request: $\epsilon_\kappa \sim h_\kappa^r$
- Detect anisotropy by measuring $p+1$st order derivatives of a scalar quantity (Mach number)
- Optimize mesh size to meet requested tolerance and to satisfy error equidistribution
- Meshing: BAMG in 2D, TetGen in 3D
- *Left:* NACA 0012, $M = 0.5$, $Re = 5000$, $p = 2$ adapted on drag

Example: Adaptation + Cut Cells

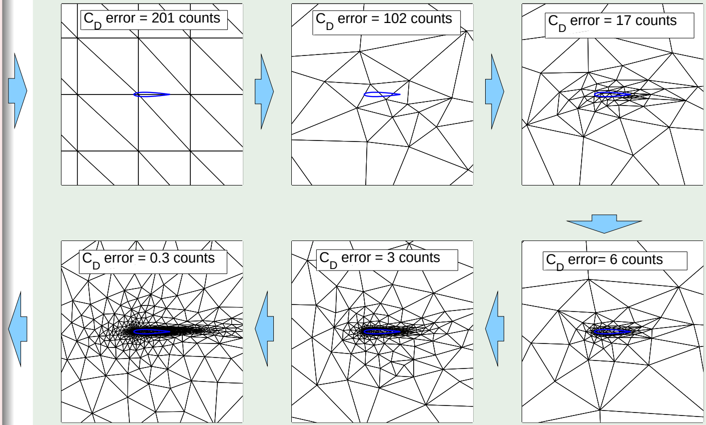
User

- $M_\infty = 0.5$, Re = 5000
- C_D to within 1 count

NACA 0012 geometry



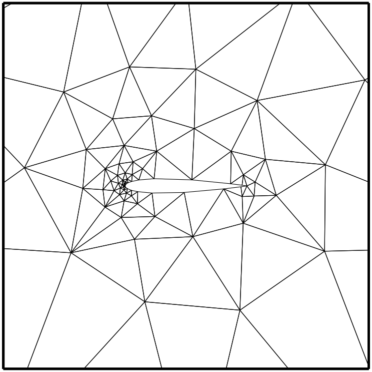
Automated



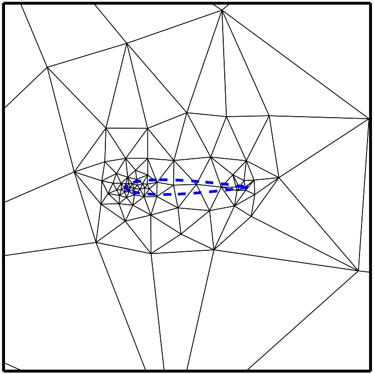
Outline

Cut-Cell Drag Adaptation in a Viscous Case

NACA 0012, $M = 0.5$, $Re = 5000$, $\alpha = 2^o$: drag adjoint adaptation
(Discontinuous Galerkin FEM discretization for all results)

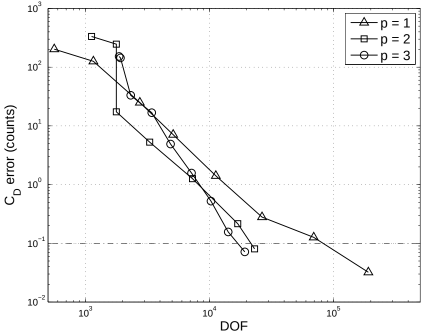


Initial boundary-conforming mesh

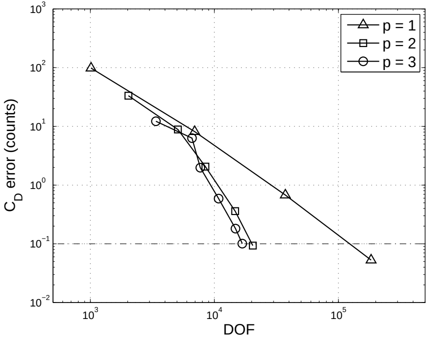


Initial cut-cell mesh

- Degree of freedom (DOF) vs. drag output error for $p = 1, 2, 3$
- Requested tolerance is 0.1 drag counts (horizontal line)
- Cut-cell and boundary-conforming results are similar

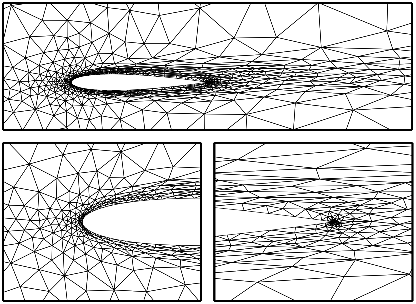


Boundary-conforming

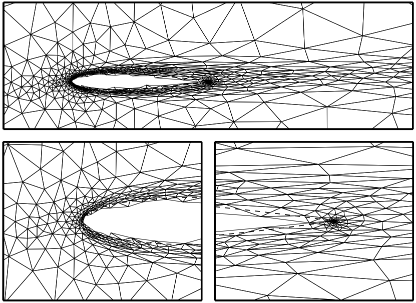


Cut-cell

Viscous Case: Final Meshes



$p = 3$ adapted boundary-conforming mesh

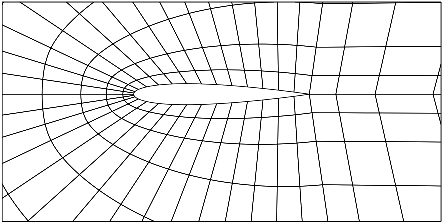


$p = 3$ adapted cut-cell mesh

- Meshes generated with BAMG (INRIA)
- $p = 1$ meshes have approximately 50 times more elements

Viscous Case: Indicator Comparison

- Hanging-node adaptation
- fixed fraction: 10%
- $q = 3$ geometry representation
- Quad boundary conforming meshes
- $p = 2$ solution interpolation
- Measured lift and drag



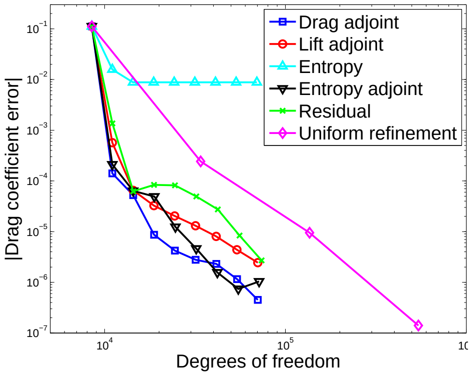
Initial mesh

Indicators

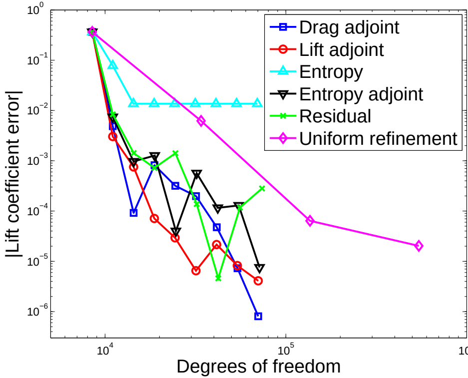
1. Drag adjoint
2. Lift adjoint
3. Entropy adjoint
4. Residual
5. Entropy

Viscous Case: Indicator Comparison (ctd.)

- Degree of freedom (DOF) versus output error for $p = 2$
- Entropy adjoint performance is comparable to output adjoints



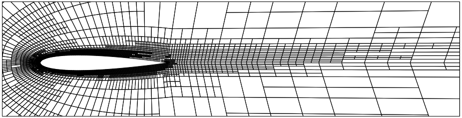
Drag Error



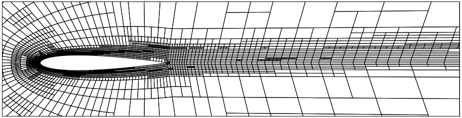
Lift Error

Viscous Case: Indicator Comparison, Final Meshes

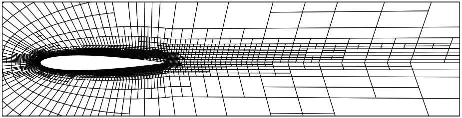
- Entropy adjoint refinement similar to output adjoints
- Leading edge, boundary layer, and initial wake targeted



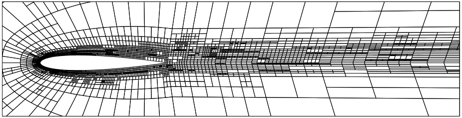
Drag Adjoint



Entropy Adjoint



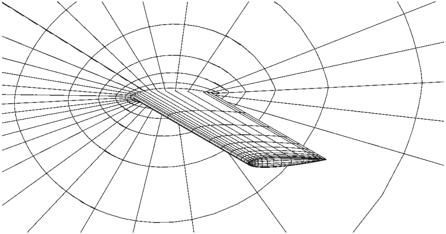
Lift Adjoint



Residual

NACA Wing: Indicator Comparison

- Hanging-node adaptation
- fixed fraction: 10%
- $q = 3$ geometry representation
- Hex boundary conforming meshes
- $p = 2$ solution interpolation
- Measured lift and drag



Initial mesh

Indicators

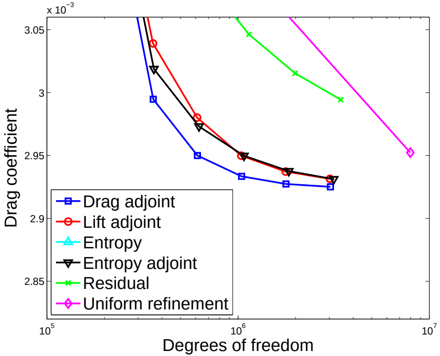
1. Drag adjoint
2. Lift adjoint
3. Entropy adjoint
4. Residual
5. Entropy

Wing:

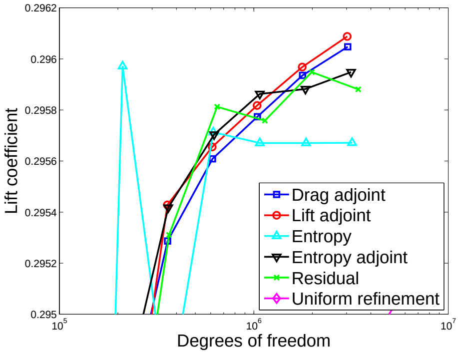
- Unswept, untapered
- Rounded wingtip
- Aspect ratio = 10

NACA Wing: Indicator Comparison (ctd.)

- Degree of freedom (DOF) versus output error for $p = 2$
- Entropy adjoint performance again comparable to output adjoints

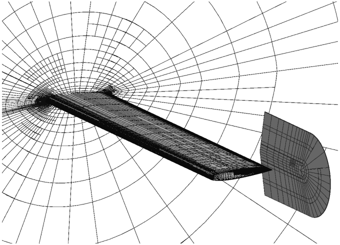


Drag

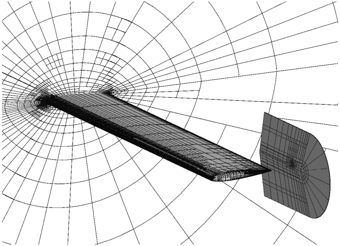


Lift

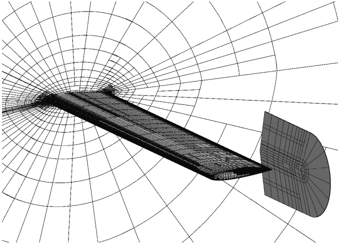
NACA Wing: Indicator Comparison, Final Meshes



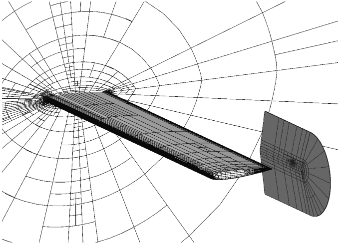
Drag Adjoint



Entropy Adjoint



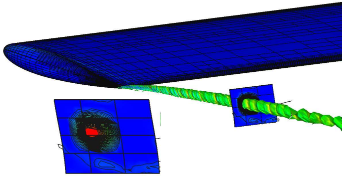
Lift Adjoint



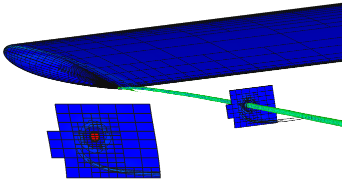
Residual

NACA Wing: Indicator Comparison, Tip Vortex

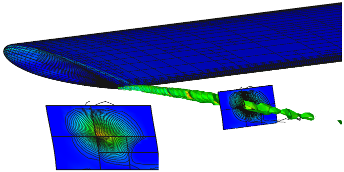
Visualization of entropy isosurface and transverse cut contours



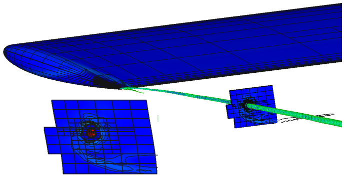
Drag Adjoint



Entropy Adjoint



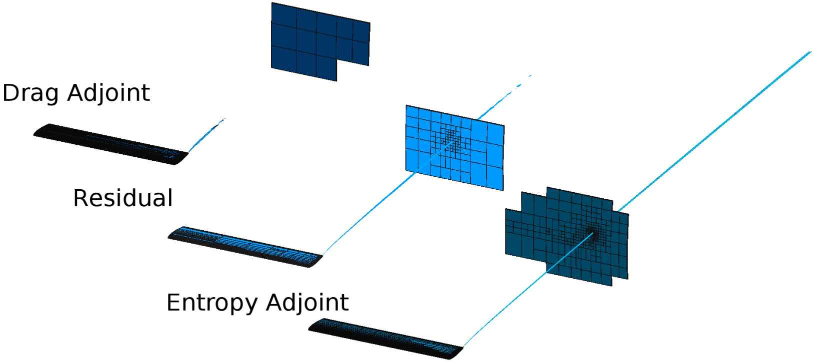
Lift Adjoint



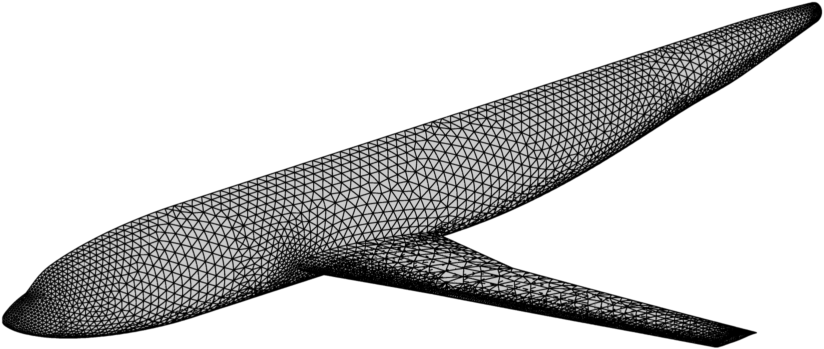
Residual

NACA Wing: Indicator Comparison, Tip Vortex

Entropy adjoint indicator targets tip vortex due to nonzero entropy residual

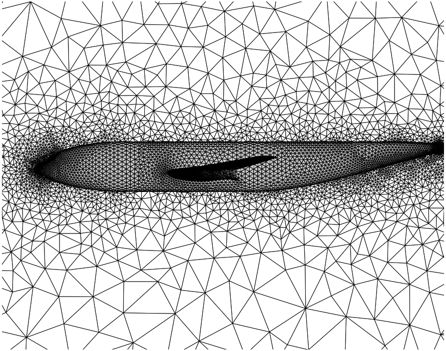


Wing-Body: 3D Cut Cell Example, Geometry

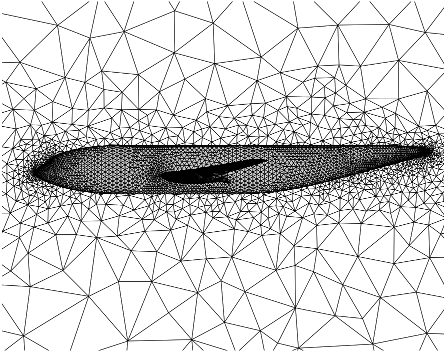


- Geometry from Drag Prediction Workshop
- 10,000 quadratic surface patches

Wing-Body: Adapted Meshes

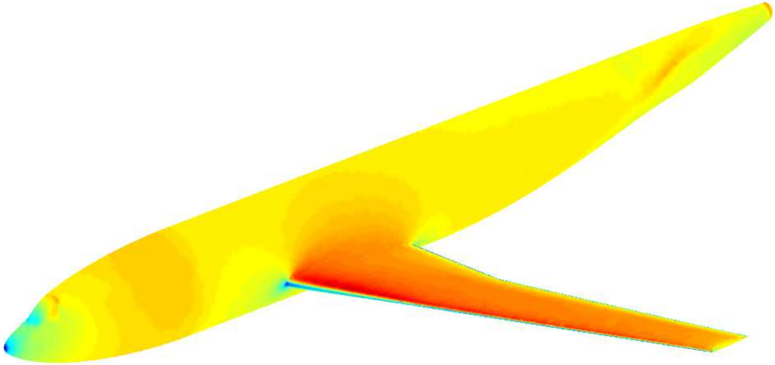


$p = 1$: 300,000 elements



$p = 2$: 85,000 elements

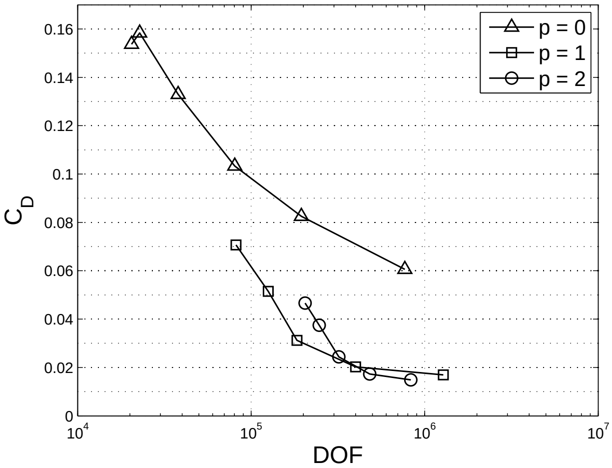
Wing-Body: Solution



- Inviscid $M_\infty = 0.1$ flow
- Surface Mach number contours shown for a $p = 2$ solution

Wing-Body Drag Comparison

Adaptation using drag adjoint with $p = 0, 1, 2$



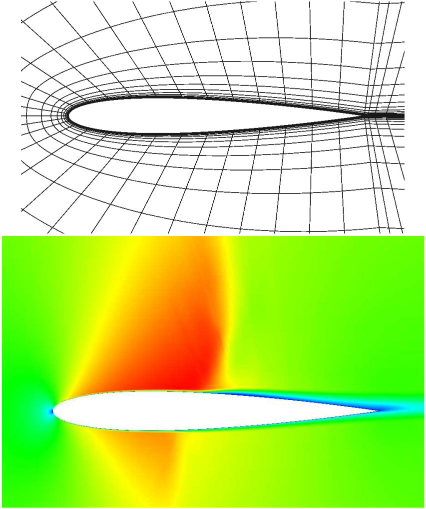
Ongoing Work

RANS turbulence modeling

- Spalart-Allmaras (SA) model
- Consistent discretization
- Scalable solvers

Shock stabilization

- Required to eliminate high-order oscillations at discontinuities
- Pursuing resolution-based artificial viscosity [Persson & Peraire, 2006]



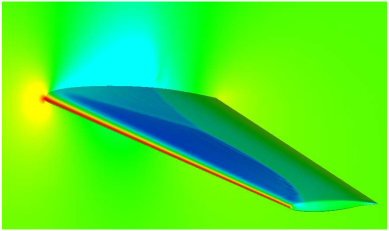
Right: NACA 0012, M=0.5, $\alpha = 1.25^o$, $Re = 100k$, $p = 3$. SA model

Drag Prediction Workshop

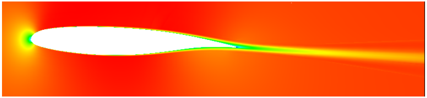
DPW II wing-alone test case

- $M = 0.76$
- $Re = 10^6$
- $\alpha = 0.5^o$
- $p = 2$
- 40,000 elements
- hexahedral curved mesh from available multiblock linear mesh

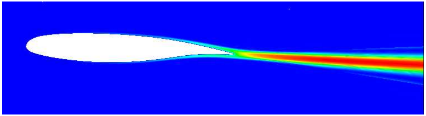
Next Step
Solution-based adaptation



Pressure



x-momentum



SA working variable

Concluding Remarks

- Robust CFD analysis of complex configurations requires error estimation and mesh adaptation
- Output error estimation based on adjoint solutions is a practical technique for accurately solving the hyperbolic problems common in aerospace applications
- The connection between entropy variables and adjoint solutions leads to a novel indicator – the potential and limitations of which are currently being investigated
- Robust mesh adaptation is one of the largest barriers for the effective implementation of these methods
- Investigating high-order cut-cells as a long-term research area for mesh generation

Acknowledgments

Collaborators

- David Darmofal
- Philip Roe
- Karen Willcox

Questions?