Is My CFD Mesh Adequate? A Quantitative Answer

Krzysztof J. Fidkowski

Gas Dynamics Research Colloquium
Aerospace Engineering Department
University of Michigan

January 26, 2011
1. Introduction and Motivation
2. Outputs and Adjoint
3. Output Error Estimation
4. Mesh Adaptation
5. A Steady Result
6. Unsteady Extension and Result
7. Conclusions and Ongoing Work
Meshes for Computational Fluid Dynamics

- Various types supporting different discretizations.
- Resolution (mesh size, order) affects accuracy of flowfield approximation.
- In unsteady simulations, time step size is part of the “mesh.”

Unstructured surface mesh

Cartesian cut-cells

Multiblock volume mesh
Unstructured meshes can be generated with less user intervention (still not fully automated for complex geometries).

Multi-block meshes are of highest quality for high $Re$ viscous calculations.

- Wing-body geometry, $M = 0.75$, $C_L = 0.5$, $Re = 5 \times 10^6$.
- Drag computed with various state of the art CFD codes.

Differences in:
- Physical models
- Discretization
- Mesh size distribution

1 drag count ($0.0001 C_D$) $\approx$ 4-8 passengers for a large transport aircraft
Sources of Error

**Reality**
- Observation errors
- Noise, calibration...

**Validation**
- Comparison of CFD and experiment

**Verification**
- CFD solution/data
  - Convergence errors
  - Discretization errors
  - Modeling errors

**Experimental data**
- Compounding of errors

K.J. Fidkowski (UM)  GDRC 2011  January 26, 2011  7 / 38
Verification: Control of Numerical Error

- Dominant source is *discretization error*
- Controlling error means answering
  1. How much error is present? (*error estimation*)
  2. How do I get rid of it? (*mesh adaptation*)

- Possible strategies:

<table>
<thead>
<tr>
<th></th>
<th>Error estimation?</th>
<th>Effective adaptation?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource exhaustion</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Expert assessment</td>
<td>Maybe</td>
<td>Maybe</td>
</tr>
<tr>
<td>Convergence studies</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Comparison to experiments</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Feature–based adaptation</td>
<td>No</td>
<td>Maybe</td>
</tr>
<tr>
<td>Output–based methods</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Why Outputs?

Output = scalar quantity computed from the CFD solution.

- A CFD solution may contain millions of degrees of freedom.
- Often of interest are only a few scalars (forces, moments, etc.)
- It is mathematically easier to speak of “error in an output” than “error in a CFD solution.”

Output error = difference between an output computed with the discrete system solution and that computed with the exact solution to the PDE.

Output error estimation

- Identifies all areas of the domain that are important for the accurate prediction of an output.
- Accounts for error propagation effects.
- Requires solution of an adjoint equation.
Consider \( N_H \) algebraic equations and an output,

\[
R_H(u_H) = 0, \quad J_H = J_H(u_H)
\]

- \( u_H \in \mathbb{R}^{N_H} \) is the vector of unknowns
- \( R_H \in \mathbb{R}^{N_H} \) is the vector of residuals (LHS of the equations)
- \( J_H(u_H) \) is a *scalar* output of interest

**Adjoint definition**

The discrete output adjoint vector, \( \psi_H \in \mathbb{R}^{N_H} \), is the sensitivity of \( J_H \) to an infinitesimal residual perturbation, \( \delta R_H \in \mathbb{R}^{N_H} \),

\[
\delta J_H \equiv \psi_H^T \delta R_H
\]
Discrete Adjoint Equation

The perturbed state, \( u_H + \delta u_H \), must satisfy

\[
R_H(u_H + \delta u_H) + \delta R_H = 0 \quad \Rightarrow \quad \frac{\partial R_H}{\partial u_H} \delta u_H + \delta R_H = 0,
\]

Linearizing the output we have,

\[
\delta J_H = \frac{\partial J_H}{\partial u_H} \delta u_H = \psi^T_H \delta R_H = -\psi^T_H \frac{\partial R_H}{\partial u_H} \delta u_H
\]

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

\[
\left( \frac{\partial R_H}{\partial u_H} \right)^T \psi_H + \left( \frac{\partial J_H}{\partial u_H} \right)^T = 0
\]
If the following hold:

1. the algebraic equations came from a consistent discretization of a continuous PDE, and
2. the residual and output combination are *adjoint consistent*,

then the discrete vector $\psi_H$ approximates the *continuous adjoint* $\psi$.

$\psi$ is a Green’s function relating source residual perturbations in the PDE to output perturbations.
The discrete adjoint, $\psi_H$, is obtained by solving a linear system.

This system involves linearizations about the primal solution, $u_H$, which is generally obtained first.

When the full Jacobian matrix, $\frac{\partial R_H}{\partial u_H}$, and an associated linear solver are available, the transpose linear solve is straightforward.

When the Jacobian matrix is not stored, the discrete adjoint solve is more involved: all operations in the primal solve must be linearized, transposed, and applied in reverse order.

In unsteady discretizations, the adjoint must be marched backward in time from the final to the initial state.
1. Introduction and Motivation
2. Outputs and Adjoints
3. Output Error Estimation
4. Mesh Adaptation
5. A Steady Result
6. Unsteady Extension and Result
7. Conclusions and Ongoing Work
Consider two discretization spaces:

1. A **coarse** space with $N_H$ degrees of freedom
2. A **fine** one with $N_h > N_H$ degrees of freedom

The fine discretization is usually obtained from the coarse one by refining the mesh or increasing the approximation order.

The coarse state $u_H$ will generally not satisfy the fine-level equations: $R_h(I_h^H u_H) \neq 0$, where $I_h^H$ is a coarse-to-fine prolongation operator.

The fine-level adjoint, $\psi_h$, translates the residual perturbation $\delta R_h \equiv -R_h(I_h^H u_H)$ to an output perturbation:

$$\delta J \approx -(\psi_h)^T R_h (I_h^H u_H)$$

\[ \text{adjoint-weighted residual} \]

*Approximation sign is present because $\delta R_h$ is not infinitesimal.*
NACA 0012, $M_\infty = 0.5$, $\alpha = 5^\circ$

*Interested in lift error in a $p = 1$ (second-order accurate) finite element solution. Using $p = 2$ for the fine space in error estimation.*

- Adjoint-based error estimate: $- (\psi_h)\ ^T \ R_h \ (I_h ^H u_H) = -0.001097$
- Actual difference: $\delta J = -0.001099$
Adaptive Solution Flowchart

Initial coarse mesh & error tolerance

Flow and adjoint solution

Output error estimate

Error localization

Mesh adaptation

Tolerance met?

Done
Goal: need to identify problematic areas of the mesh

The output error estimate,

\[ \delta J \approx - (\psi_h)^T R_h (l_H^H u_H) \]

is a sum over mesh elements (for finite volume/element methods)

Error indicator on element \( k \)

\[ \epsilon_k = \left| - (\psi_{h,k})^T R_{h,k} (l_H^H u_H) \right| \]

Refinement in areas where \( \epsilon_{k,H} \) is large will reduce the residual there and hence improve the output accuracy.
### Adaptation Mechanics

1. **$h$-adaptation:** only triangulation is varied
2. **$p$-adaptation:** only approximation order is varied
3. **$hp$-adaptation:** both triangulation and approximation order are varied

Given an error indicator, how should the mesh be adapted?

- Refine some/all elements?
- Incorporate anisotropy (stretching)?
- How to handle elements on the geometry?

Since mesh generation is difficult in the first place, adaptation needs to be automated to enable multiple iterations.
Meshing and Adaptation Strategies

Metric-based anisotropic mesh regeneration (e.g. BAMG software)

Riemannian ellipse

Local mesh operators, and direct optimization

Cut-cell meshes: Cartesian and simplex
1 Introduction and Motivation
2 Outputs and Adjoint
3 Output Error Estimation
4 Mesh Adaptation
5 A Steady Result
6 Unsteady Extension and Result
7 Conclusions and Ongoing Work
NACA Wing, $M = 0.4$, $\alpha = 3^\circ$

- Hanging-node adaptation
- Cubic curved geometry representation
- Hexahedral meshes
- $p = 2$ (third order) DG solution approximation
- Interested in lift and drag

**Indicators**

1. Drag and lift adjoints
2. Entropy adjoint
3. Residual
4. Entropy

Initial mesh

Mach number contours
NACA Wing, \( M = 0.4, \alpha = 3^\circ \)

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

![Graphs showing Drag and Lift coefficient versus Degrees of freedom for NACA Wing, M = 0.4, \( \alpha = 3^\circ \)]
NACA Wing, $M = 0.4, \alpha = 3^\circ$, Final Meshes

Drag Adjoint

Entropy Adjoint

Lift Adjoint

Residual
Visualization of entropy isosurface and transverse cut contours
Outline

1. Introduction and Motivation
2. Outputs and Ad joints
3. Output Error Estimation
4. Mesh Adaptation
5. A Steady Result
6. Unsteady Extension and Result
7. Conclusions and Ongoing Work
The error estimation equations hold for unsteady problems.

The adjoint is more expensive for nonlinear problems:
- Adjoint solve proceeds backwards in time.
- State vector is required at each time for linearization.
- Must store or recompute state.

Adaptation is trickier with the additional dimension of time.

Current approach: **finite elements in space and time**.

\[ u_H(x, t) = \sum_n \sum_j u_{H,j}^n(x) \phi_{H,j}^n(t) \]

- \( \phi_{H,j}(x) = j^{\text{th}} \) spatial basis function
- \( \phi_{H,j}^n(t) = n^{\text{th}} \) temporal basis function
- Basis functions are discontinuous in space and time (DG).
Discretization: Space-Time Mesh

- Time is discretized in slabs (all elements advance the same $\Delta t$)
- Each space-time element is prismatic (tensor product: $T^H_e \otimes I^H_k$)
- The spatial mesh is assumed to be invariant in time

\[ \Omega_{\text{time}} \]
\[
\begin{align*}
  t_k & \quad \text{time slab } I_{H,k} \\
  t_{k-1} & \quad \text{element } T_{H,e}
\end{align*}
\]
Unsteady Adaptive Solution

Solution steps

1. Initial space–time mesh and error tolerance
2. Solve primal by marching forward in time
3. Solve adjoint, estimate the output error, and localize to adaptive indicators in a loop over time slabs backwards in time
4. Error tolerance met? (yes) → Done
   (no) → Identify elements and time slabs for refinement
5. Adapt space-time mesh

The adaptation consists of hanging-node refinement in space and slab bisection in time.
Impulsively-Started Airfoil in Viscous Flow

**Governing equations (Navier-Stokes)**

\[
\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial}{\partial x_i} \left[ \mathbf{F}_i^I(\mathbf{u}) - \mathbf{F}_i^V(\mathbf{u}, \nabla \mathbf{u}) \right] = 0
\]

- \( \mathbf{u} = [\rho, \rho u, \rho v, \rho E]^T \)
- \( \mathbf{F}_i^I(\mathbf{u}) \) is the inviscid flux
- \( \mathbf{F}_i^V(\mathbf{u}, \nabla \mathbf{u}) \) is the viscous flux

**Initial and boundary conditions**

- At \( t = 0 \) the velocity is blended smoothly to zero in a circular disk around the airfoil
- The freestream conditions are \( M_\infty = 0.25, \ \alpha = 8^\circ, \ \text{Re} = 5000 \)
The output of interest is the lift coefficient integral from $t = 9$ to $t = 10$

Time integral output definition.
A vortex-shedding pattern has been established by the time of the output measurement.

Convergence of output using various adaptive indicators. Shown on output-based results are:
- Error bars at $\pm \delta J$ (actual error est.)
- Whiskers at $\pm \epsilon$ (conservative error est.)
Lift coefficient time histories for adapted meshes with similar degrees of freedom. Values shown only at end of time slabs.

Convergence of the $L_2$ time history error for various adaptive indicators

**Output-based adaptation yields not only an accurate scalar output, but also an accurate lift coefficient time history.**
Impulsively-Started Airfoil: Adapted Spatial Meshes

- Meshes shown at iterations with similar total degrees of freedom.
- Spatially-marginalized output error indicator is shown on the elements of the output-adapted mesh.

Adapted on output error (5956 elements)
Adapted on approximation error (4585 elements)
Adapted on residual (7929 elements)
Output error indicator yields a fairly-uniform temporal refinement.

Approximation error focuses on the initial time (dynamics of the IC) and the latter 1/3 of the time, when the shed vortices develop.

Residual creates a mostly-uniform temporal mesh as it tracks acoustic waves.
Outline

1. Introduction and Motivation
2. Outputs and Adjoint
3. Output Error Estimation
4. Mesh Adaptation
5. A Steady Result
6. Unsteady Extension and Result
7. Conclusions and Ongoing Work
Conclusions

- An adequate CFD mesh is one that yields a sufficiently-low discretization error.
- The effect of discretization error on outputs can be quantified.
- Added cost: the solution of an adjoint problem.
- Benefit: error estimates and efficient meshes.
- Ideas apply to both steady and unsteady CFD problems.

What lies ahead

- Unsteady problems:
  - Dynamically-refined spatial meshes and grid motion
  - Forward solution checkpointing
  - Adjoint stability
- Entropy adjoint as a cheaper alternative
- Error bounds instead of estimates