18th AIAA Computational Fluid Dynamics Conference, June 25–28, 2007, Miami, FL

Adjoint Error Estimation and Adaptive Refinement for Embedded-Boundary Cartesian Meshes

Marian Nemec*
ELORET Corp., Moffett Field, CA 94035

Michael J. Aftosmis†
NASA Ames Research Center, Moffett Field, CA 94035

We present an approach for the computation of error estimates in output functionals such as lift or drag for an embedded-boundary Cartesian mesh method. The approach relies on the solution of an adjoint equation and provides error estimates that can be used to both improve the accuracy of the functional and guide a mesh refinement procedure. This is a significant step in our research toward automating the simulation process for flows in complex geometries. The accuracy of the approach is verified on an analytic model problem and validated against common results in the literature. The robustness of the approach is examined for two test cases in three dimensions, namely, an isolated wing in transonic flow and a canard-controlled missile in supersonic flow. The results demonstrate that the approach is tolerant of coarse initial meshes. A practical advantage of the approach is that the adaptive mesh refinement may be performed with a fixed surface triangulation. In all cases considered, the approach provided reliable estimates of the output functional on computationally affordable meshes.

I. Introduction

Adjoint methods are becoming increasingly important in the CFD analysis of aerodynamic performance. These methods offer reliable estimates of the sensitivity of output functionals, such as lift or drag, to the many parameters involved in numerical simulations. In optimal shape design, for example, the sensitivity of an objective function can be computed at a cost that is essentially independent of the number of design variables used to define the shape. Similarly in error analysis, the adjoint equation provides the sensitivity of the functional to residual errors in the governing flow equations. Both examples can be thought of as problems in optimal control. In design problems we adjust shape parameters to control the flow and minimize a given objective function, while in error analysis we adjust mesh refinement to control discretization errors that influence the functional of interest. Becker and Rannacher,[1] Giles and Pierce,[2] and Barth[3] provide detailed overviews on the use of adjoint equations in error analysis of CFD simulations.

In this work, we present the development of adjoint-based error estimates and mesh refinement for a Cartesian method with embedded, cut-cell boundaries.[4] Cartesian methods are very attractive for problems involving complex geometry since they enable automated mesh generation.[5–8] While these methods have largely eliminated the meshing bottleneck for inviscid-flow simulations, the issue of constructing an appropriately refined mesh that captures all relevant flow features for a given geometry and flow conditions is still largely a heuristic procedure that requires close supervision and user expertise. Our ultimate goal is to remove this dependence on user expertise, bringing a new level of automation and error control to the flow simulation process. The main challenge is in the formulation of an accurate error estimate that provides a bound on the uncertainty in the simulation and triggers local mesh refinement to reduce the predicted errors. Adjoint error estimates offer an approach that localizes mesh enrichment to only those flow features that influence the accuracy of target functionals, promising exceptional computational efficiency and automation.

In earlier work, Aftosmis and Berger[9] presented an incremental strategy for h-refinement of nested Cartesian meshes. Adaptation criteria in that work were based on either a feature detection scheme (scaled

*Research Scientist, Advanced Supercomputing Division, MS T27B; mnemec@mail.arc.nasa.gov. Member AIAA.
†Aerospace Engineer, Advanced Supercomputing Division, MS T27B; maftosmis@mail.arc.nasa.gov. Senior Member AIAA.
Copyright © 2007 by the American Institute of Aeronautics and Astronautics, Inc. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Governmental purposes. All other rights are reserved by the copyright owner.

first-differences of local flow quantities) or Richardson-type local truncation-error estimates. The current work enhances these capabilities with an adjoint-based adaptation criterion using the discrete adjoint solver of Nemec *et al.*[10] Our approach most closely follows the work of Venditti and Darmofal,[11] who proposed a discrete error approach based on an embedded "truth" mesh to estimate functional errors for finite-volume methods. The use of an embedded mesh and the discrete form of the governing equations greatly simplifies the implementation of the error estimation procedure. This approach has been extended to three-dimensional problems on tetrahedral meshes by Park,[12] and has been validated for both surface and volume functionals.[13] Furthermore, Lu[14] has refined and extended this approach to the discontinuous Galerkin (DG) method, and recently, Fidkowski[15] has demonstrated the effectiveness of this approach for an embedded-boundary DG method using simplex cut-cells. Examples of adjoint error analysis and mesh refinement for body-conforming discretizations can be found in Müller and Giles,[16] Barthet *et al.*,[17] Kim and Nakahashi,[18] Balasubramanian and Newman,[19] and Richter.[20]

The purpose of this paper is to evaluate the effectiveness of adjoint error estimates in the context of Cartesian-mesh methods with cut-cell boundaries and refinement interfaces. We evaluate the accuracy of two solution reconstruction schemes used in the computation of the adjoint correction term and the functional error bound. The first uses the linear reconstruction scheme of the flow solver. The second uses trilinear and triquadratic interpolation functions for the reconstruction of the adjoint solution. We verify the accuracy of the adjoint error estimates by constructing a model problem with an analytic solution. Unlike many approaches, nested refinement of Cartesian cells is inherently a discrete operation. This gives us the opportunity to directly examine the effect of local stretching errors on the accuracy of the adjoint error estimates. Our focus is on the irregular cells of the domain, where numerical artifacts may cause large residual errors, and consequently, poor error estimates. Two representative three-dimensional test problems are used to demonstrate the performance of the proposed approach, namely, the ONERA M6 wing in transonic flow and a missile geometry in supersonic flow. Factors under consideration include the accuracy of the error estimate, efficiency of the adaptation strategy, and overall robustness of the approach for problems with complex geometry and non-smooth flow features.

II. Governing Equations and Numerical Method

A. Flow Solver

Our goal is to compute a reliable approximation of a functional $\mathcal{J}(Q)$, such as lift, derived from a flow solution $Q = [\rho, \rho u, \rho v, \rho w, \rho E]^{\mathrm{T}}$ satisfying the steady-state three-dimensional Euler equations of a perfect gas

$$\mathcal{F}(Q) = 0 \tag{1}$$

Let $J(\mathbf{Q}_H)$ denote an approximation of the functional computed on an affordable Cartesian mesh with an average cell size H, where $\mathbf{Q} = [\bar{Q}_1, \bar{Q}_2, \ldots, \bar{Q}_N]^{\mathrm{T}}$ is the discrete solution vector of the cell-averaged values for all N cells of the mesh and J is the discrete operator used to evaluate the functional. The governing equations are discretized on a multilevel Cartesian mesh with embedded boundaries. The mesh consists of regular Cartesian hexahedra everywhere, except for a layer of body-intersecting cells, or *cut-cells*, adjacent to the boundaries as illustrated in Fig. 1. The spatial discretization of Eq. 1 uses a cell-centered, second-order accurate finite volume method with a weak imposition of boundary conditions, resulting in a system of equations

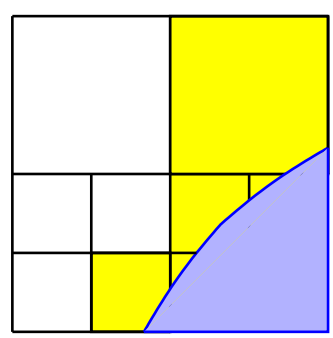


Figure 1. Multilevel Cartesian mesh in two-dimensions with a cut-cell boundary

$$\mathbf{R}(\mathbf{Q}_H) = 0 \tag{2}$$

The flux-vector splitting approach of van Leer[21] is used. Steady-state flow solutions are obtained using a five-stage Runge–Kutta scheme with local time stepping, multigrid, and a domain decomposition scheme for parallel computing. For further details on the flow solution algorithm, see Aftosmis *et al.*[4,22] and Berger *et al.*[23]

B. Adjoint Error Correction

To approximate the functional error $|\mathcal{J}(Q) - J(\mathbf{Q}_H)|$, we consider isotropic refinement of an initial Cartesian mesh to obtain a finer mesh with average cell size h containing approximately $8N$ cells (in three dimensions), and seek to compute the discrete error $|J(\mathbf{Q}_h) - J(\mathbf{Q}_H)|$ without solving the problem on the embedded mesh. As in Ref. 11, the derivation begins by considering a Taylor series expansion of the functional on the embedded mesh about the coarse mesh solution

$$J(\mathbf{Q}_h) \approx J(\mathbf{Q}_h^H) + \frac{\partial J(\mathbf{Q}_h^H)}{\partial \mathbf{Q}_h} \left(\mathbf{Q}_h - \mathbf{Q}_h^H\right) \tag{3}$$

where the algebraic vector $\mathbf{Q}_h^H$ denotes a reconstruction of the flow solution from the coarse mesh to the embedded mesh. Similarly, we write the residual equation as

$$\mathbf{R}(\mathbf{Q}_h) = 0 \approx \mathbf{R}(\mathbf{Q}_h^H) + \frac{\partial \mathbf{R}(\mathbf{Q}_h^H)}{\partial \mathbf{Q}_h} \left(\mathbf{Q}_h - \mathbf{Q}_h^H\right) \tag{4}$$

and by combining Eqs. 3 and 4, we obtain the following expression for the functional on the embedded mesh

$$J(\mathbf{Q}_h) \approx J(\mathbf{Q}_h^H) - \frac{\partial J(\mathbf{Q}_h^H)}{\partial \mathbf{Q}_h} \left[\frac{\partial \mathbf{R}(\mathbf{Q}_h^H)}{\partial \mathbf{Q}_h}\right]^{-1} \mathbf{R}(\mathbf{Q}_h^H) \tag{5}$$

The adjoint equation is obtained from Eq. 5 by defining the following intermediate product

$$\left[\frac{\partial \mathbf{R}(\mathbf{Q}_h^H)}{\partial \mathbf{Q}_h}\right]^{\mathrm{T}} \psi_h = \frac{\partial J(\mathbf{Q}_h^H)}{\partial \mathbf{Q}_h}^{\mathrm{T}} \tag{6}$$

where the vector ψ denotes the adjoint variables. Details of the adjoint solution method are given in Ref. 10. The adjoint variables provide a correction term that systematically scales the residual errors to improve the accuracy of the functional on the embedded mesh

$$J(\mathbf{Q}_h) \approx J(\mathbf{Q}_h^H) - \psi_h^{\mathrm{T}}\ \mathbf{R}(\mathbf{Q}_h^H) \tag{7}$$

Note that Eq. 7 does not require an explicit evaluation of the flow solution $\mathbf{Q}_h$, but we do require the solution of the adjoint equation, Eq. 6, on the embedded mesh. To circumvent this difficulty, the adjoint equation is solved on the coarse mesh

$$\left[\frac{\partial \mathbf{R}(\mathbf{Q}_H)}{\partial \mathbf{Q}_H}\right]^{\mathrm{T}} \psi_H = \frac{\partial J(\mathbf{Q}_H)}{\partial \mathbf{Q}_H}^{\mathrm{T}} \tag{8}$$

and the adjoint values are interpolated to the embedded mesh. Similar to the flow solution variable $\mathbf{Q}_h^H$, let ψ_h^H represent a reconstructed solution of the coarse mesh adjoint solution on the embedded mesh. Substituting the reconstructed adjoint solution into Eq. 7 yields the following expression for an estimate of the functional on the embedded mesh

$$J(\mathbf{Q}_h) \approx J(\mathbf{Q}_h^H) - \underbrace{(\psi_h^H)^{\mathrm{T}}\, \mathbf{R}(\mathbf{Q}_h^H)}_{\text{Adjoint Correction}} - \underbrace{(\psi_h - \psi_h^H)^{\mathrm{T}}\, \mathbf{R}(\mathbf{Q}_h^H)}_{\text{Remaining Error}} \tag{9}$$

The adjoint solution from the coarse mesh provides a computable correction term and a remaining error term that primarily represents the error made in solving the adjoint equation on the coarse mesh. Equation 9 is exact for linear functionals and residual equations. Unfortunately, the remaining error term cannot be computed directly, because it requires a solution of the adjoint equation on the embedded mesh. Various strategies have been proposed in the literature to address this issue and among the most promising approaches is the use of higher-order reconstruction schemes to approximate the embedded adjoint solution.[24,25] The higher-order reconstruction operator, as well as the reconstruction operators for the flow and adjoint solutions ($\mathbf{Q}_h^H$ and ψ_h^H), are described in the next section.

C. Data Reconstruction

Nested subdivision of a regular hexahedron on the coarse mesh creates eight embedded hexahedra. The purpose of the reconstruction operator is to prolong the flow and adjoint solutions from the coarse parent cell to all its embedded cells, such that we obtain a smoother representation of the solution while maintaining its order of accuracy. Venditti and Darmofal[11] used linear and quadratic reconstruction operators, where the quadratic operator was constructed to minimize an $\mathcal{H}_1$ norm using a patch-wise least-squares procedure. Park[12] proposed a simpler reconstruction scheme using one-dimensional cubic splines and similar approaches have been studied in Refs. 17, 19.

We present two formulations of the reconstruction operator, both based on local polynomial interpolation. The first formulation relies on the linear reconstruction operator of the flow solver. The flow solution value at each of the embedded cell centroids is given by

$$Q_h^H = \bar{Q}_H + \mathbf{d}\phi\nabla Q_H \tag{10}$$

where $\bar{Q}_H$ is the cell average value on the coarse mesh, $\mathbf{d}$ is the distance vector from the coarse cell centroid to the centroid of the embedded cell, ϕ is the slope limiter, and ∇Q is the coarse cell solution gradient determined via linear least-squares. A similar procedure is used for the adjoint variables, and we use the Barth-Jespersen limiter to enforce monotonic reconstruction. Substituting the linearly reconstructed solutions into Eq. 9 gives the following expression for the functional on the embedded mesh

$$J(\mathbf{Q}_h) \approx J(\mathbf{Q}_\mathrm{L}) - (\psi_\mathrm{L})^\mathrm{T}\,\mathbf{R}(\mathbf{Q}_\mathrm{L}) - (\psi_\mathrm{L} - \psi_\mathrm{C})^\mathrm{T}\,\mathbf{R}(\mathbf{Q}_\mathrm{L}) \tag{11}$$

where the subscript $(\cdot)_\mathrm{L}$ denotes linear reconstruction (Eq. 10) and subscript $(\cdot)_\mathrm{C}$ denotes that a constant solution value obtained from the centroid of the coarse cell is used in each embedded cell. Note that we use the higher-order interpolant in both the adjoint correction and remaining error terms. The main advantages of this approach are robustness, especially in regions where the flow and adjoint solutions exhibit steep gradients and discontinuities, and simplicity of implementation. The primary disadvantage is a potential reduction in accuracy of the adjoint correction and remaining error terms. We investigate these issues in the first test case of the Results section.

The second formulation relies on the construction of trilinear and triquadratic interpolation functions. The approach is similar to the standard shape functions for "brick" elements used in the finite element method. The polynomial used for the trilinear interpolant in each coarse cell is given by

$$\psi_\mathrm{TL} = c_0 + c_1x + c_2y + c_3z + c_4xy + c_5xz + c_6yz + c_7xyz \tag{12}$$

To construct this polynomial, we use a normalized local coordinate system and reconstruct the solution from the cell centroid to the vertices of the cell using Eq. 10, as illustrated in Fig. 2. Hence, each vertex receives contributions from all cells common to this vertex, not just the face neighbors of the cell. The average of all contributions determines the solution value at each vertex, and together the values at the eight vertices determine a unique trilinear interpolant.

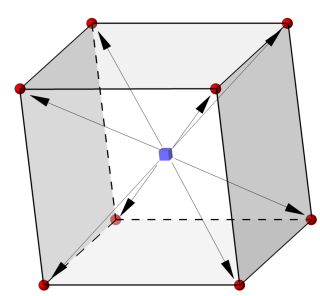


Figure 2. Reconstruction of solution values and gradients from cell centroid to vertices

For the higher-order reconstruction operator, we use the following restricted triquadratic polynomial

$$\begin{aligned}\psi_\mathrm{TQ} = {} & c_0 + c_1x + c_2y + c_3z + c_4xy + c_5xz + c_6yz + c_7xyz + c_8x^2 + c_9y^2 + c_{10}z^2 \\ & + c_{11}x^2y + c_{12}x^2z + c_{13}xy^2 + c_{14}xz^2 + c_{15}y^2z + c_{16}yz^2 + c_{17}x^2yz + c_{18}xy^2z + c_{19}xyz^2\end{aligned} \tag{13}$$

To determine the 20 unknown coefficients of this polynomial, we use the eight solution values (from the trilinear case) in conjunction with the solution gradient at the vertices. The gradient value at a vertex is determined by the arithmetic average of all gradients from cells common to the vertex. We solve the resulting over-determined system of 32 equations using a least-squares procedure from the LAPACK linear algebra package. Substituting the trilinear and triquadratic interpolants into Eq. 9 gives a new expression for the functional on the embedded mesh

$$J(\mathbf{Q}_h) \approx J(\mathbf{Q}_\mathrm{L}) - (\psi_\mathrm{TQ})^\mathrm{T}\,\mathbf{R}(\mathbf{Q}_\mathrm{L}) - (\psi_\mathrm{TQ} - \psi_\mathrm{TL})^\mathrm{T}\,\mathbf{R}(\mathbf{Q}_\mathrm{L}) \tag{14}$$

Note that the trilinear and triquadratic interpolants are presently used only for the reconstruction of the adjoint variable.

D. Error Estimates

The remaining error term in Eq. 9 can be used to determine a bound on the local error in each cell of the embedded mesh

$$e = \left| \left(\psi_h - \psi_h^H\right)^{\mathrm{T}} \mathbf{R}_h(\mathbf{Q}_h^H) \right| \tag{15}$$

We examine two alternate forms of this error estimate using the computable expressions for the remaining error terms in Eqs. 11 and 14, respectively, given by

$$e_{\mathrm{L}} = \left| (\psi_{\mathrm{L}} - \psi_{\mathrm{C}})^{\mathrm{T}} \mathbf{R}(Q_{\mathrm{L}}) \right| \tag{16}$$

$$e_{\mathrm{Q}} = \left| (\psi_{\mathrm{TQ}} - \psi_{\mathrm{TL}})^{\mathrm{T}} \mathbf{R}(Q_{\mathrm{L}}) \right| \tag{17}$$

The error estimate given by Eq. 16 uses the linear reconstruction scheme of the flow solver, while Eq. 17 relies on the trilinear and triquadratic interpolation schemes. These error estimates are a simpler version of the estimate suggested by Venditti and Darmofal,[11] who include an additional term based on the residual of the adjoint equation. The sum of the cell-wise error contributions to the functional gives an estimate of the net functional error E

$$E = \sum_{k=0}^{N} e_k \tag{18}$$

E. Adaptation Strategy

The mesh refinement parameter used in the adaptation procedure is based on the error estimates given by Eqs. 16 and 17. Note that these equations are defined on the embedded mesh. The refinement parameter for each cell of the working mesh is determined by the sum of the contributions from its embedded cells. Given a user specified global tolerance `TOL` for the functional of interest, we specify a target error level for each cell $t = \texttt{TOL}/N$. A cell is flagged for refinement if

$$e > \lambda t \tag{19}$$

where $\lambda \geq 1$ is an optional global threshold factor used to scale the refinement parameter (discussed below). Termination criterion for the adaptation is satisfied when the global error E is less than the user specified tolerance `TOL`.

The refinement parameter is used to drive an incremental refinement strategy described in Ref. 9. Starting from a coarse initial mesh, cells are flagged for refinement as indicated by Eq. 19. The solution is computed on the refined mesh and this adaptation cycle continues until the termination criterion is satisfied. Only one level of refinement is allowed at each adaptation cycle. The refinement region is typically enlarged by one or two coarse cells to ensure that the interface boundary of the refined region of the mesh is in a region of low error. In most cases, this is sufficient to prevent the propagation of interface boundaries and ensure mesh smoothness during the adaptation procedure.

Equation 19 includes the global threshold factor λ to permit control over the mesh refinement strategy. Efficiency arguments suggest that one should add relatively few cells on the coarse meshes early in the refinement procedure, and ideally, add the most cells on the last cycle to satisfy the required error tolerance. This strategy not only reduces the number of expensive solutions performed on relatively fine meshes, but also prevents excessive refinement of coarse meshes when the accuracy of the flow and adjoint solutions is poor. It focuses early efforts of the adaptation on the worst regions of the mesh, seeking to get the maximum improvement in the functional for each refined cell. This is accomplished by starting the adaptation with a large threshold value and gradually reducing the threshold as the adaptation progresses. For example, $\lambda = 16$ represents a typical starting value and we reduce the threshold by a factor of two or four on subsequent refinement cycles.

III. Results

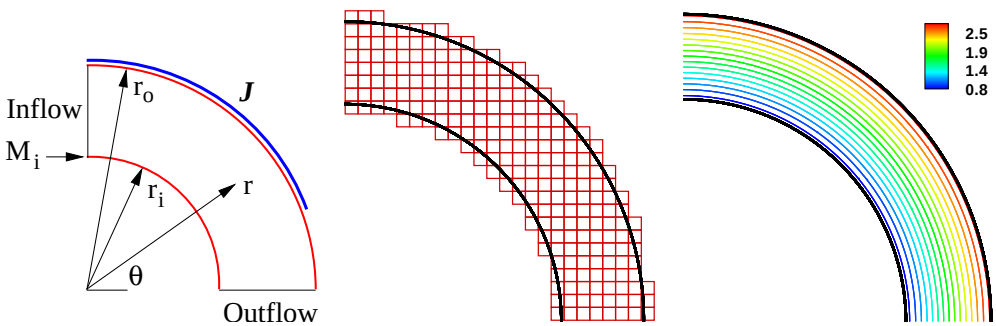
A. Supersonic Vortex Problem

The first test case considers an analytic problem to verify the accuracy of the adjoint correction term and its error bound using the linear and triquadratic reconstruction schemes introduced in Sec. IIC for the flow and

adjoint variables. In addition, this problem also examines the accuracy of the mesh refinement parameter and provides a simple demonstration of the strengths of the adjoint approach over traditional feature-based and local trucation-error approaches. The problem involves isentropic flow between concentric circular arcs at supersonic conditions as shown in Fig. 3. This problem was used in Refs. 9 and 26 to verify the order of accuracy of the flow and adjoint solvers, respectively. The goal of the computation is the evaluation of the integral of pressure over a portion of the outer arc, as sketched in Fig. 3(a),

$$\mathcal{J} = \int_0^{0.7158\frac{\pi r_o}{2}} p_{r_o}\ \mathrm{d}l = \frac{0.7158\pi r_o}{2} p_{r_o} \tag{20}$$

which is similar in form to the lift and drag boundary integrals commonly used in aerodynamic design. The seemingly odd choice of 71.58% of arc length is made to coincide with a Cartesian face of the initial mesh, thereby fixing the domain of integration such that it remains invariant with mesh refinement. The functional value based on the analytic solution is $J \approx 4.4019$. We use second-order spatial discretization without limiters to solve the problem.



(a) Problem setup (b) Initial mesh (c) Nondimensional pressure

Figure 3. Supersonic vortex model problem ($M_i = 2.25$, $r_i = 1$, and $r_o = 1.382$). The functional, J, is defined over the indicated portion of the outer arc

1. *Uniform Mesh Refinement*

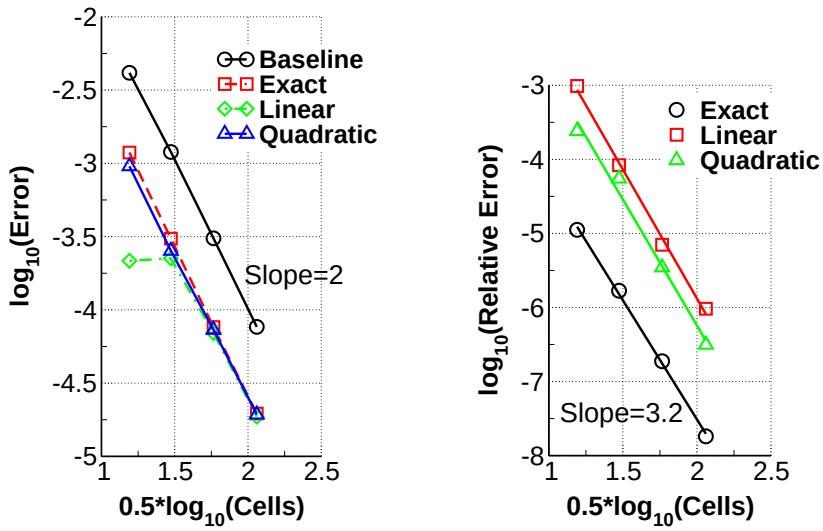
A uniform mesh refinement study is considered starting from the initial mesh shown in Fig. 3(b) to examine the error convergence rate of the functional and the adjoint correction terms. Figure 4(a) shows that the functional error convergence rate, labeled as "Baseline", is second-order. This is consistent with the spatial discretization of the flow equations. The error is measured with respect to the analytic solution. We also show the accuracy of the adjoint correction term computed via three approaches: linear (Eq. 11) and triquadratic (Eq. 14) reconstruction of the adjoint variables, labeled as "Linear" and "Quadratic", respectively, as well as a direct solution for the adjoint variables on the embedded mesh (Eq. 7), labeled as "Exact". The exact adjoint solution is only used to establish a benchmark for the two reconstruction schemes since its computational cost is roughly equivalent to a flow solution on the embedded mesh.

Returning to Fig. 4(a), the adjoint correction terms reduce the error in the functional by roughly a half-order of magnitude, but do not change the convergence rate of the functional. A naive interpretation of Fig. 4(a) is that the linear correction provides a better estimate of the functional than the exact correction. However, the goal of Eq. 7 is to predict the value of the functional on the embedded mesh from the coarse mesh solution. As seen in Fig. 4(a), for each baseline solution the exact correction predicts the functional value on the next mesh exceptionally well, even on the coarsest mesh the prediction is within plotting accuracy. With this understanding, the accuracy of the linear correction term is poor on the coarsest mesh, but improves substantially over the next three meshes. The triquadratic correction term performs nearly as well as the exact correction on all meshes.

In Fig. 4(b), we quantify the differences between the three adjoint correction formulations. In this plot, the error is measured with respect to the functional value on the embedded mesh after the adjoint correction. This is referred to as the relative error and is given by

$$\text{Relative Error} = |J(\mathbf{Q}_h) - [J(\mathbf{Q}_h^H) - (\psi_h^H)^{\mathrm{T}}\,\mathbf{R}(\mathbf{Q}_h^H)]| \tag{21}$$

As expected, the exact correction term provides the lowest relative error in Fig. 4(b). We observe that for all correction terms, the relative error is superconvergent - the convergence rates exceed third-order. Formally, the expected convergence rate of this term is fourth-order; however, there is some reduction due to non-smoothness of the adjoint solution (as discussed below) and a reduction in accuracy in the evaluation of flow residuals. Overall, triquadratic interpolation provides a better estimate of the functional than the linear reconstruction scheme.



(a) Error is measured with respect to analytic solution

(b) Error is measured with respect to embedded mesh value

Figure 4. Functional error convergence for uniform mesh refinement, supersonic vortex problem

Figure 5 shows the convergence of error estimates based on Eqs. 16 and 17 labeled as "Linear Bound" and "Quadratic Bound", respectively. We also include the convergence of the relative error to show that the triquadratic interpolation scheme provides a tighter bound on the estimate of the remaining error. The use of the absolute value function in Eqs. 16 and 17 prevents error cancellation from neighboring cells and results in a conservative estimate of the bound on the remaining error.

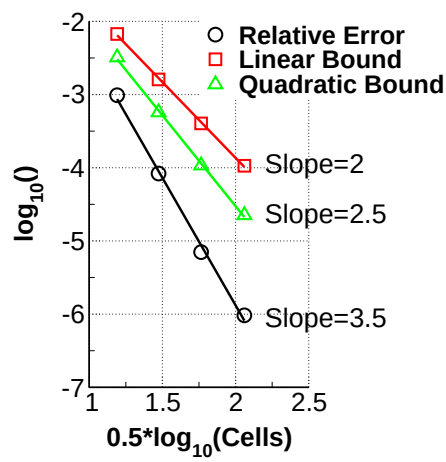


Figure 5. Comparison of error-bound estimates using linear and quadratic reconstruction schemes with the relative error for the supersonic vortex problem

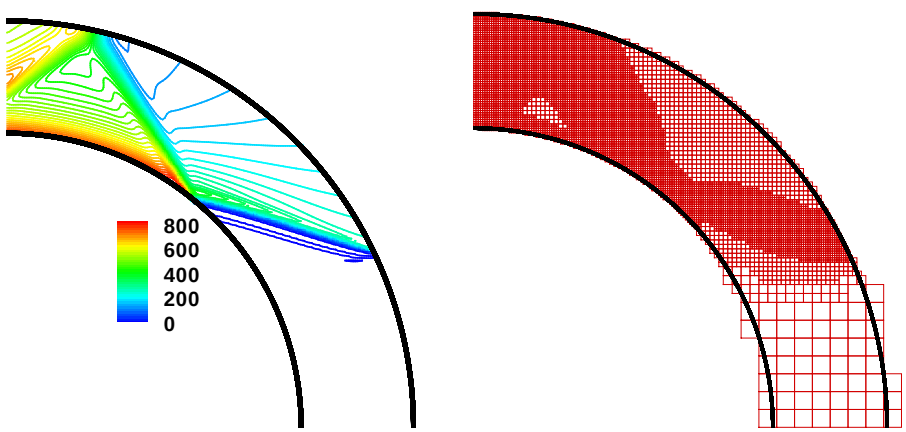
2. Adaptive Mesh Refinement

Before presenting adaptive mesh refinement results, it is insightful to examine the adjoint solution for this simple flow. Figure 6(a) shows the adjoint solution corresponding to the density variable. Given the smoothness

of the pressure field in Fig. 3(c), the nonlinear variations displayed in the adjoint field are initially striking. They remind us that the smooth shockless turning of a supersonic flow is the result of a delicate balance between expansion and compression. The nonlinear variations reveal the influence of point-source mass perturbations, which include interactions with the inner arc, on the functional being computed in Eq. 20. The adjoint variable vanishes 71.58% along the outer arc, precisely because any perturbation downstream from this location cannot influence the functional in this hyperbolic problem. The largest sensitivity of the functional to residual errors occurs along the lower arc, which is due to the nonuniform inlet Mach number distribution and the number of local Mach wave reflections that can reach the functional from this region of the domain.

To test the performance of the mesh refinement procedure, we specify an error tolerance of $\texttt{TOL} = 0.00002$ and use the error estimate based on triquadratic interpolation (Eq. 17) as the refinement parameter. The refinement threshold, λ in Eq. 19, is set to one. Figure 6(b) shows the mesh after three adaptation cycles starting from the uniform mesh shown in Fig. 3(b). The refinement pattern is clearly driven by the features of the adjoint solution. No mesh refinement occurs near the exit of the duct and a significant portion of the cut-cells along the outer arc is not highly refined. This indicates that despite their irregularity, the error contribution from these cells remains low.

Figure 7 shows the error convergence of the functional and the convergence of the refinement parameter. In Fig. 7(a), the error is measured with respect to the analytic solution and we compare the error convergence rate of the functional on a sequence of uniformly refined meshes with the adjoint adapted meshes. Note that the error in the adjoint corrected functional, Eq. 14, is significantly smaller than the baseline functional value. In fact, with the adjoint correction, the functional error on the final adapted mesh is over an order of magnitude smaller relative to the functional value on finest mesh of the uniform refinement study. These two meshes contain roughly the same number of cells. As shown in Fig. 7(b), the refinement parameter reaches the requested error tolerance on the fourth adaptation cycle. As a final comment, we have also evaluated refinement parameters based on the linear reconstruction scheme, as well as a direct solution of the adjoint problem on the embedded mesh and found that with an appropriate scaling of the error tolerance, the performance of the mesh refinement procedure is essentially the same for all cases.

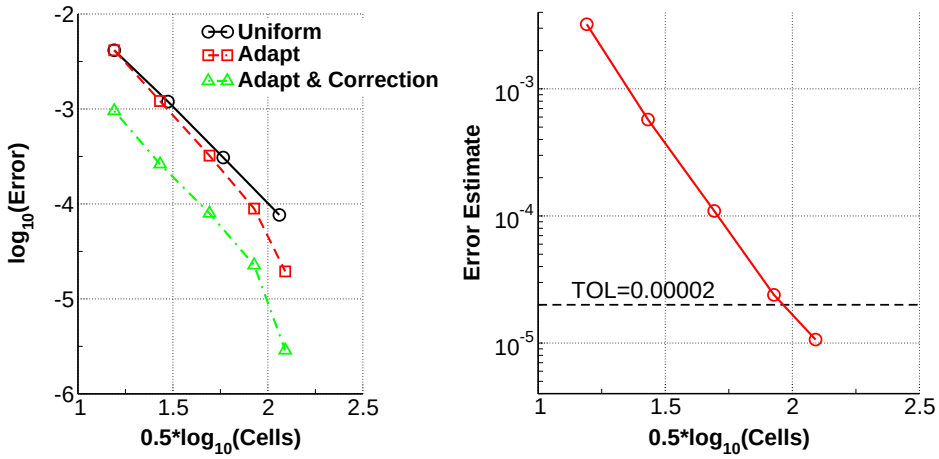


(a) Density adjoint contours (b) Mesh after three adaptation cycles

Figure 6. Adjoint solution and adaptive refinement for the supersonic vortex problem

B. ONERA M6 Wing

The second test case considers the familiar ONERA M6 wing[27] at transonic conditions, $M_\infty = 0.84$ and $\alpha = 3.06°$. Drag is selected as the functional of interest and we specify an error tolerance for the adaptation of 0.002. The refinement parameter is based on the error estimate given in Eq. 16 for the linear reconstruction scheme. The distance to the outer boundary is 30 chords and the cells have an aspect ratio of two in the spanwise direction. The surface triangulation contains roughly 500,000 triangles, which provides a high resolution model of the wing. The starting value of the threshold parameter λ in Eq. 19 is set to 64 and



(a) Functional (b) Error Bound

Figure 7. Convergence of functional and remaining error estimate for adaptive refinement of the supersonic vortex problem

then reduced by a factor of four on subsequent adaptation cycles. The Barth–Jespersen limiter is used for the computation.

Figure 8 shows near-body cut-planes of the initial and final meshes with Mach number contours on the final mesh. The initial mesh contains less than 3,800 cells with only 99 cut-cells. The 20 count error tolerance in drag is satisfied after ten levels of refinement and the final mesh contains approximately 1.8 million cells. The mesh refinement is primarily focused on the leading- and trailing-edge regions with only a moderate refinement of the shock-wave system and the leading-edge stagnation line. Overall, the regularity of the final mesh is good without any obvious smoothness issues at refinement interfaces or wall boundaries.

Figure 9 examines the convergence of the drag functional with mesh refinement. The final drag value is 0.0119 without the adjoint correction and 0.0117 with the correction. The correction term consistently reduces the baseline drag and the two values are convergent. Furthermore, the correction predicts the drag value on the next mesh of the adaptation with improving accuracy as the adaptation progresses. The final drag values are in good agreement with the survey of results presented by Holst,[28] the previous work of Aftosmis and Berger,[9] and are slightly larger than the values reported by Park;[12] however, limiters were not used in that work. Lastly, a close look at results presented in Fig. 9 reveals a mild change of slope in the convergence of drag at the 7th adaptation cycle. This change in slope coincides with resolution of the base-flow at the wing's blunt trailing edge, as this geometric feature remains "hidden" until the 7th cycle. A close-up of the trailing-edge region is shown in Fig. 10, where a small region of separated flow is observed.

C. Canard-Controlled Missile

A canard-controlled missile is considered to demonstrate the automation capabilities of the proposed approach on a complex geometry problem with significant off-body flow features. The missile geometry is shown in Fig. 11 and consists of two canard surfaces with shafts and body cut-outs, four tail fins in an "X" configuration and a narrow groove upstream of the tail. The surface discretization contains roughly 68,000 triangles, which is representative of a typical engineering model. As in previous studies,[9,29,30] we consider supersonic flow conditions at $M_\infty = 1.6$ and $\alpha = 3°$, with canards deflected 15° (nose up). Separate adaptations are performed for three functionals, namely, axial and normal forces, and the pitching moment. The refinement parameter is given by Eq. 16 and is based on the linear reconstruction scheme. The moment center is located approximately a half-body length from the nose. The cell aspect ratio is unity and we use the van Leer limiter in all computations. The value of the threshold parameter λ in Eq. 19 is unity in all adaptations.

Figure 12 shows the evolution of the mesh from the initial mesh in the upper left corner to a fine mesh in the lower right corner for the pitching-moment functional. The initial mesh is very coarse, containing 8,079 cells with only 337 cut-cells, and clearly does not require any expert knowledge to setup. Note the emergence of a "diamond-shaped" adaptation region defined by the bow-shock angle and the farthest downstream extent

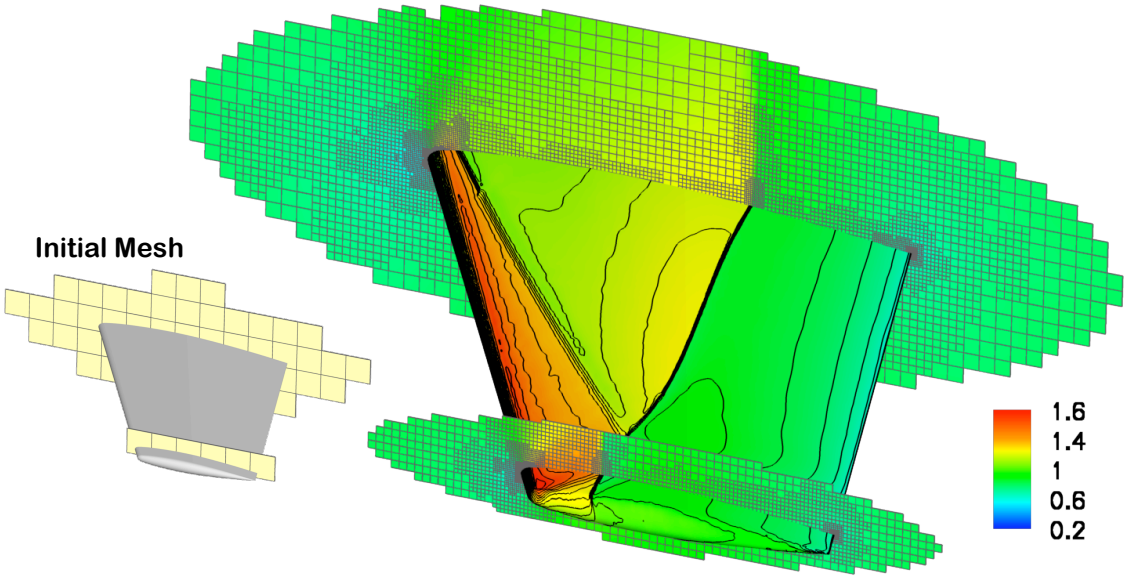


Figure 8. Near-body initial and final meshes for the ONERA M6 wing, shown with Mach number contours. Location of cutting planes is at 20% and 93% of span ($M_\infty = 0.84$ and $\alpha = 3.06°$)

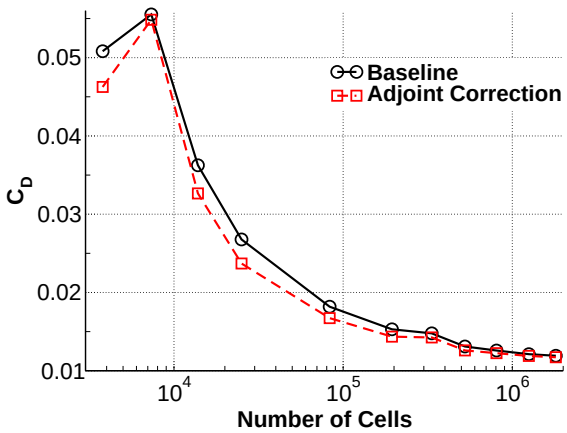


Figure 9. Drag convergence using adjoint-based mesh refinement for the ONERA M6 wing ($M_\infty = 0.84$ and $\alpha = 3.06°$)

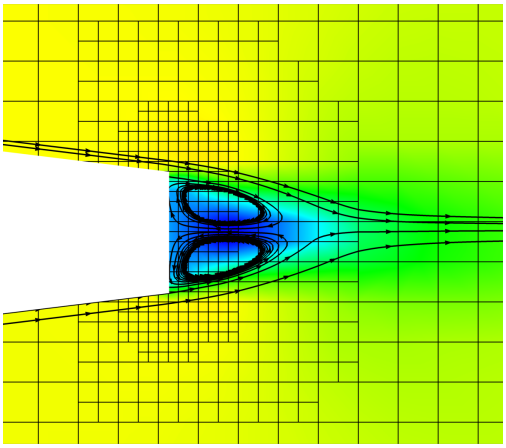


Figure 10. Trailing-edge region near the root of the ONERA M6 wing, shown with x-velocity contours (blue < 0)

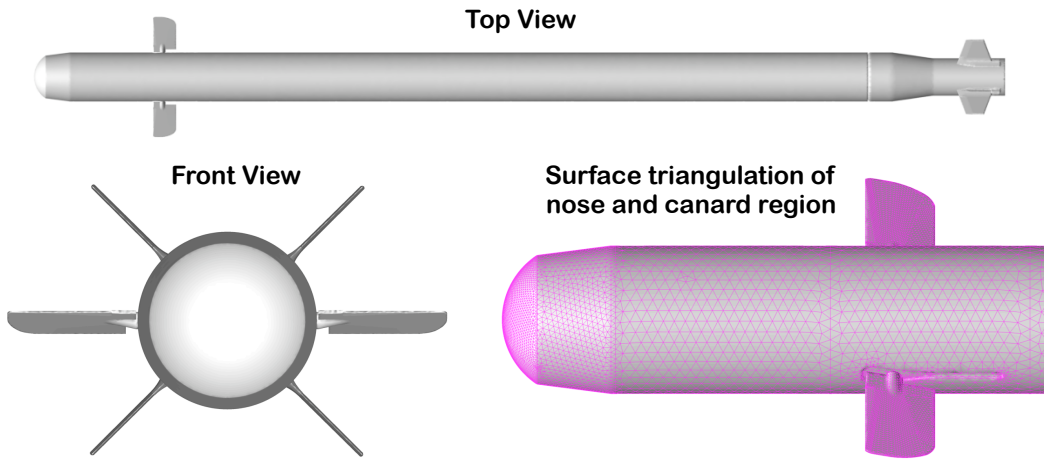


Figure 11. Canard-controlled missile surface geometry

of interactions of the flow with the tail geometry. The mild skewness of the diamond region is due to the non-zero angle of attack. The reason for this adaptation pattern is explained in Fig. 13, which shows the adjoint solution for the x-momentum variable. This plot reveals the influence of perturbations in the residual of the x-momentum equation on the pitching-moment functional. Disturbances originating downstream of the intersection of the Mach angle and the tail cannot influence the value of the pitching moment due to the supersonic flow conditions. Therefore, in this region of the flow, the value of the refinement parameter (Eq. 16) is small because the interpolation error in the adjoint variables vanishes. The value of the refinement parameter is also small in the region upstream of the bow-shock, but this is due to small residual errors in the flow equations. Since the refinement parameter is an inner product of the adjoint interpolation error with the flow residuals, the adaptation is localized to this diamond-shaped region. The front view of the missile in Fig. 13 emphasizes the fact that the adjoint variable is non-zero only in the near-body region close to the fins, which leads to very localized mesh refinement. Details of the mesh refinement near the nose and tail of the missile are shown in Fig. 14. Note that the error indicator limits the refinement of the shock waves, as well as expansions, to near-body regions. Similar results are obtained for the axial and normal force functionals.

A salient feature of this test case is the formation of canard vortices that convect nearly 40 canard-chord lengths to the tail fins and influence the force and moment coefficients. Figure 15 shows the near-body mesh refinement and the location of the vortices using contours of total pressure loss along several cutting planes. By the 5th adaptation cycle, the vortices are well resolved and travel the full length of the missile body. Close examination of the refinement parameter (not shown) indicates that the value of the functional is also sensitive to the resolution of the fins' leading-edge stagnation stream surfaces.

Figure 16 summarizes the mesh convergence of axial force, normal force and pitching moment with refinement. We specify an error tolerance of 0.05 for axial force and 0.3 for normal force. For each functional, we plot the convergence of the adaptation for that particular functional, as well as values obtained from the adaptations that target the other two functionals. Referring to Fig. 16(a), the error tolerance for drag is satisfied on the 4th adaptation cycle on a mesh with roughly one million cells. We perform two additional adaptation cycles to verify that the changes in the functional remain small. The adaptations that target the two other functionals reach a similar value of drag on meshes containing roughly two million cells. Similar results are obtained for the normal force, shown in Figs. 16(b), and an additional level of refinement is required for the pitching moment, shown in Fig. 16(c). The results indicate that for this case we obtain good predictions even for functionals that are not specifically targeted by the adaptation.

Figure 16 also shows a "Reference" value on each plot, which denotes previously published results.[30] The mesh used in Ref. 30 was hand-crafted without the use of adaptive mesh refinement and provides an anecdotal "expert mesh" data point. This mesh contains 3.4 million cells with a cell aspect ratio of 1.8 in the body normal directions. The present adjoint-based approach attains converged results on meshes with roughly half the number of cells, with a cell aspect ratio of unity. More importantly, the results are obtained

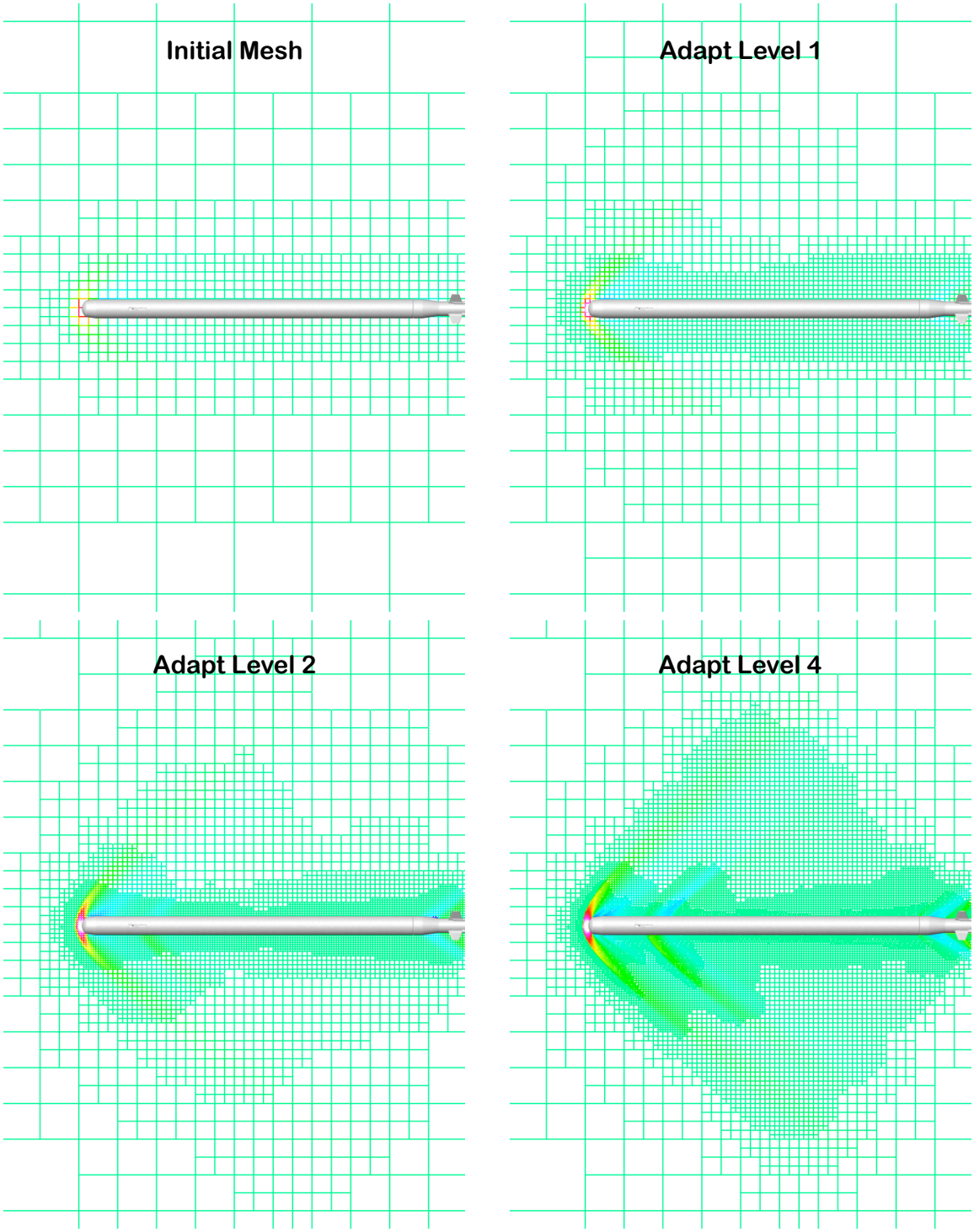


Figure 12. Sequence of mesh refinements colored by pressure for pitching moment. Symmetry plane, $M_\infty = 1.6$ and $\alpha = 3°$

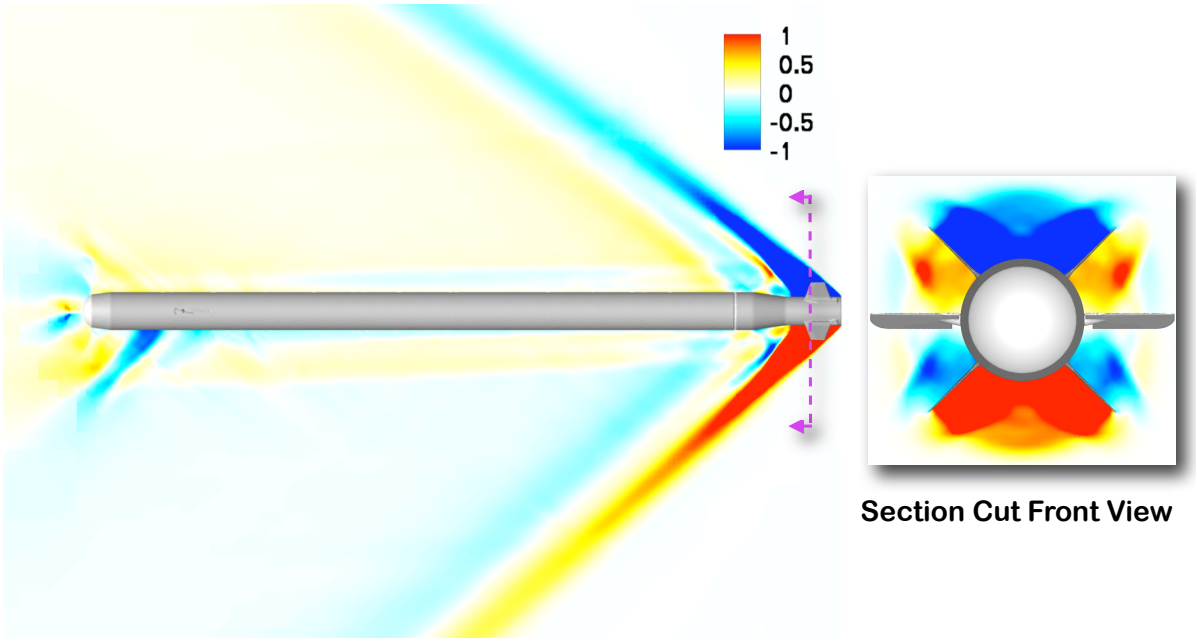
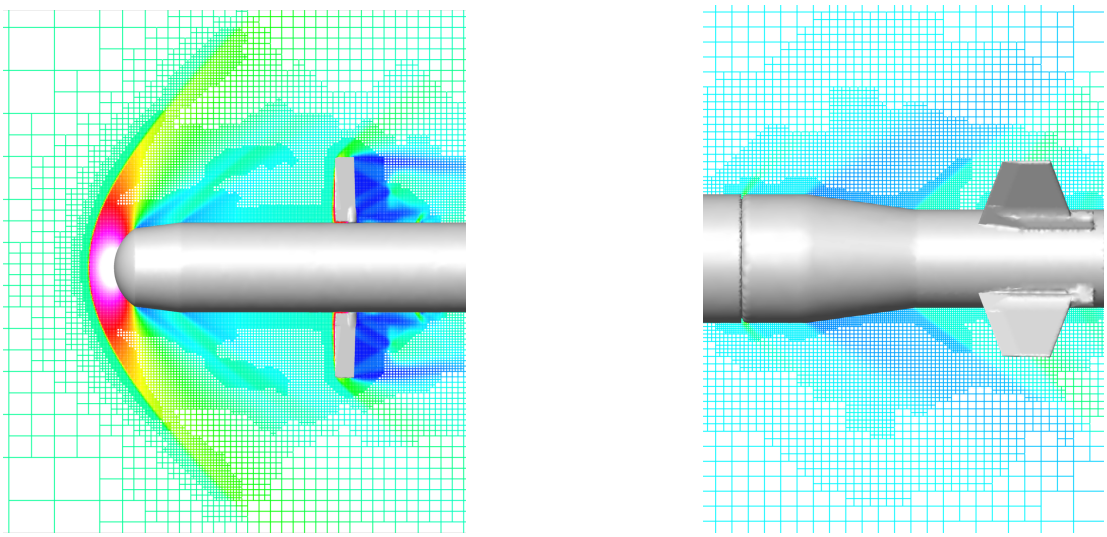


Figure 13. X**-momentum adjoint variable on symmetry plane and front view of a section cut near tail**



(a) Top view of nose

(b) Side view of tail

Figure 14. Close-up views of canard region and tail colored by pressure (4 adaptation cycles)

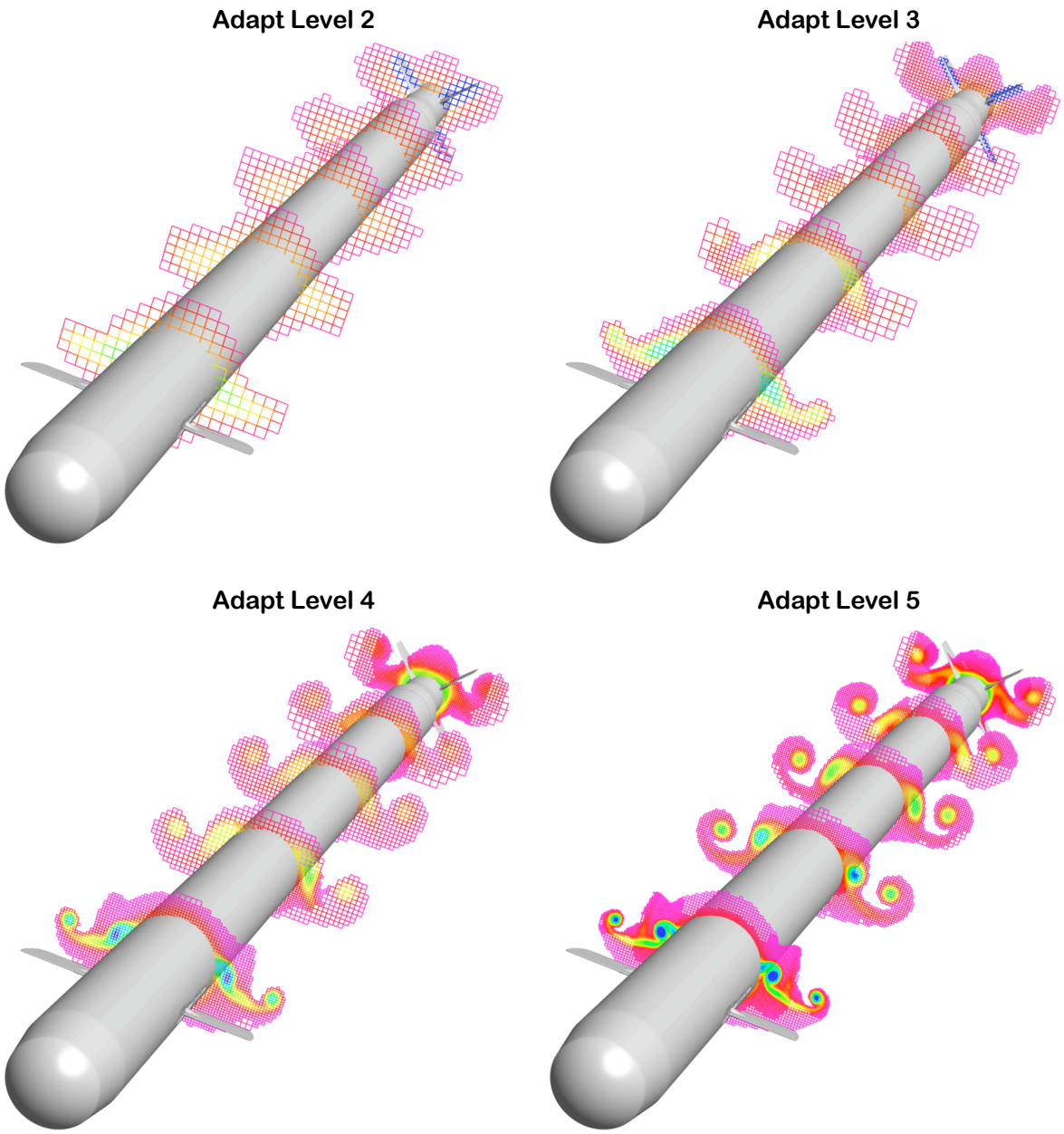
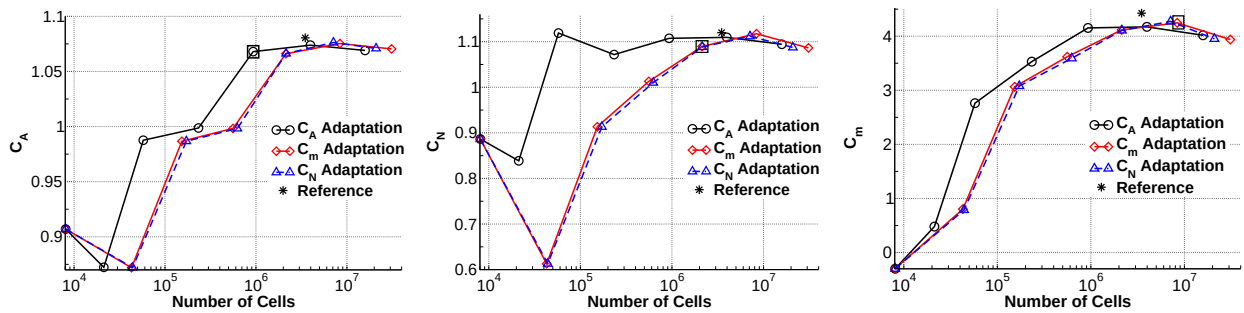


Figure 15. Formation of canard vortices showing near-body mesh refinement. Cutting planes are colored and truncated by contours of total pressure loss

in an automatic fashion starting from very coarse meshes. We attribute the small differences between the present results and the "expert mesh" to insufficient mesh resolution at the tail region of the flow in the "expert mesh".



(a) Axial force (b) Normal force (c) Pitching moment

Figure 16. Convergence of functionals for the canard-controlled missile problem. The symbol □ denotes that the error tolerance is satisfied

IV. Conclusions

An adjoint approach for the computation of functional error estimates on embedded-boundary Cartesian meshes has been developed. A detailed verification study has been presented using an analytic model problem to systematically evaluate the accuracy of the error estimates and the resulting mesh-refinement parameter. The approach has been applied to three-dimensional test cases in complex geometry and non-smooth flow. Based on our results, we can draw the following conclusions:

1. The triquadratic interpolation scheme of the adjoint variables provides a more accurate estimate of both the adjoint correction term and the remaining error than the linear reconstruction scheme.

2. For the supersonic vortex problem, adjoint error correction improves the accuracy of the functional by roughly a half-order of magnitude. Furthermore, adaptive mesh refinement based on the remaining error term improves the accuracy of the functional by an additional half-order of magnitude, resulting in significant computational savings.

3. Mesh irregularity due to refinement interfaces and cut-cells does not significantly influence the accuracy of the error estimate.

4. The incremental mesh refinement strategy in conjunction with the adjoint refinement parameter is tolerant of coarse initial meshes and converges to affordable meshes that provide reliable estimates of the functional.

5. An important advantage of the present approach in a practical setting is that the surface triangulation remains fixed throughout the mesh refinement. The triangulation is generated in a pre-processing step, prior to the CFD analysis. This operation is inexpensive because it only involves the surface model and not the volume mesh.

Future work should focus on robustness improvements for the triquadratic interpolant and the examination of alternative formulations of the refinement parameter that include the adjoint residuals. It would be interesting to evaluate the efficiency of this approach for off-body functionals, such as those encountered in sonic boom applications. We also intend to revisit the mesh adaptation strategy to maximize both meshing efficiency and robustness of the procedure.

V. Acknowledgments

The authors gratefully acknowledge Marsha Berger (NYU) and Scott Murman (NASA Ames) for helpful discussions. This work was supported by the NASA Ames Research Center contract NNA06BC19C.

References

[1]Becker, R. and Rannacher, R., "An optimal control approach to a posteriori error estimation in finite element methods," *Acta Numerica 2000*, 2001, pp. 1–102.

[2]Giles, M. B. and Pierce, N. A., "Adjoint error correction for integral outputs," *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, edited by T. Barth and H. Deconinck, Vol. 25 of *Lecture Notes in Computational Science and Engineering*, Springer-Verlag, 2002.

[3]Barth, T., "Numerical Methods and Error Estimation for Conservation Laws on Structured and Unstructured Meshes," Lecture notes, von Karman Institute for Fluid Dynamics, Series: 2003-04, Brussels, Belgium, March 2003.

[4]Aftosmis, M. J., Berger, M. J., and Adomavicius, G., "A Parallel Multilevel Method for Adaptively Refined Cartesian Grids with Embedded Boundaries," AIAA Paper 2000–0808, Reno, NV, Jan. 2000.

[5]Aftosmis, M. J., Berger, M. J., and Melton, J. E., "Robust and Efficient Cartesian Mesh Generation for Componenet-Based Geometry," *AIAA Journal*, Vol. 36, No. 6, 1998, pp. 952–960.

[6]Hunt, J. D., *An Adaptive 3D Cartesian Approach for the Parallel Computation of Inviscid Flow about Static and Dynamic Configurations*, Ph.D. thesis, University of Michigan, 2004.

[7]Dawes, W. N., "Building Blocks Towards VR-Based Flow Sculpting," AIAA Paper 2005–1156, Reno, NV, Jan. 2005.

[8]Lahur, P. R., "Automatic Hexahedra Grid Generation Method for Component-based Surface Geometry," AIAA Paper 2005–5242, Toronto, ON, June 2005.

[9]Aftosmis, M. J. and Berger, M. J., "Multilevel Error Estimation and Adaptive h-Refinement for Cartesian Meshes with Embedded Boundaries," AIAA Paper 2002–0863, Reno, NV, Jan. 2002.

[10]Nemec, M., Aftosmis, M. J., Murman, S. M., and Pulliam, T. H., "Adjoint Formulation for an Embedded-Boundary Cartesian Method," To Appear, AIAA Paper 2005–0877, Reno, NV, Jan. 2005.

[11]Venditti, D. A. and Darmofal, D. L., "Grid Adaptation for Functional Outputs: Application to Two-Dimensional Inviscid Flow," *Journal of Computational Physics*, Vol. 176, 2002, pp. 40–69.

[12]Park, M. A., "Adjoint-Based, Three Dimensional Error Prediction and Grid Adaptation," *AIAA Journal*, Vol. 42, No. 9, 2004, pp. 1854–1862.

[13]Jones, W. T., Nielsen, E. J., and Park, M. A., "Validation of 3D Adjoint Based Error Estimation and Mesh Adaptation for Sonic Boom Prediction," AIAA Paper 2006–1150, Jan 2006.

[14]Lu, J., *An* a posteriori *Error Control Framework for Adaptive Precision Optimization using Discontinuous Galerkin Finite Element Method*, Ph.D. thesis, Massachusetts Institute of Technology, 2005.

[15]Fidkowski, K. J., *A Simplex Cut-Cell Adaptive Method for High-Order Disctretizations of the Compressible Navier–Stokes Equations*, Ph.D. thesis, Massachusetts Institute of Technology, 2007.

[16]Müller, J.-D. and Giles, M. B., "Solution Adaptive Mesh Refinement Using Adjoint Error Analysis," AIAA Paper 2001–2550, June 2001.

[17]Barthet, A., Airiau, C., Braza, M., and Tourrette, L., "Adjoint-based error correction applied to far-field drag breakdown on structured grid," AIAA Paper 2006–3315, 2006.

[18]Kim, H.-J. and Nakahashi, K., "Output-Based Error Estimation and Adaptive Mesh Refinement Using Viscous Adjoint Method," AIAA Paper 2006–1395, 2006.

[19]Balasubramanian, R. and Newman III, J. C., "Comparison of Adjoint-based and Feature-based Grid Adaptation for Functional Outputs," AIAA Paper 2006–3314, 2006.

[20]Richter, T., *Parallel Multigrid Method for Adaptive Finite Elements with Application to 3D Flow Problems*, Ph.D. thesis, University of Heildelberg, 2005.

[21]van Leer, B., "Flux-Vector Splitting for the Euler Equations," ICASE Report 82-30, Sept. 1982.

[22]Aftosmis, M. J., Berger, M. J., and Murman, S. M., "Applications of Space-Filling-Curves to Cartesian Methods for CFD," AIAA Paper 2004–1232, Reno, NV, Jan. 2004.

[23]Berger, M. J., Aftosmis, M. J., and Murman, S. M., "Analysis of Slope Limiters on Irregular Grids," AIAA Paper 2005–0490, Reno, NV, Jan. 2005.

[24]Giles, M. B. and Süli, E., "Adjoint methods for PDEs: *a posteriori* error analysis and postprocessing by duality," *Acta Numerica*, Vol. 11, 2002, pp. 145–236.

[25]Pierce, N. A. and Giles, M. B., "Adjoint and Defect Error Bounding and Correction for Functional Estimates," AIAA Paper 2003–3846, June 2003.

[26]Nemec, M. and Aftosmis, M. J., "Aerodynamic Shape Optimization Using a Cartesian Adjoint Method and CAD Geometry," AIAA Paper 2006–3456, San Francisco, CA, June 2006.

[27]Schmitt, V. and Charpin, F., "Pressure Distributions on the ONERA-M6-Wing at transonic Mach numbers," *Experimental Data Base for Computer Program Assessment*, AGARD-R-138, May 1979.

[28]Holst, T. L., "On Approximate Factorization Schemes for Solving the Full Potential Equation," NASA TM 110435, Feb. 1997.

[29]Nygaard, T. A. and Meakin, R. L., "An Aerodynamic Analysis of a Spinning Missile with Dithering Canards," AIAA Paper 2002–2799, St. Luis, Missouri, June 2002.

[30]Murman, S. M., Aftosmis, M. J., and Berger, M. J., "Numerical Simulation of Rolling Airframes Using a Multilevel Cartesian Method," *AIAA Journal of Spacecraft and Rockets*, Vol. 41, No. 3, 2004, pp. 426–435.