INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **53**:1541–1569
Published online 10 October 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1361

Comparison of adjoint-based and feature-based grid adaptation for functional outputs

R. Balasubramanian[1,*,†] and J. C. Newman III[2]

[1]*Computational Simulation and Design Center, ERC, Mississippi State University, Box 9627, Mississippi State, MS 39762, U.S.A.*
[2]*Department of Aerospace Engineering, Mississippi State University, Mississippi State, MS 39762, U.S.A.*

SUMMARY

Adjoint-based and feature-based grid adaptive strategies are compared for their robustness and effectiveness in improving the accuracy of functional outputs such as lift and drag coefficients. The output-based adjoint approach strives to improve the adjoint error estimates that relate the local residual errors to the global error in an output function via adjoint variables as weight functions. A conservative adaptive indicator that takes into account the residual errors in both the primal (flow) and dual (adjoint) solutions is implemented for the adjoint approach. The physics-based feature approach strives to identify and resolve significant features of the flow to improve functional accuracy. Adaptive indicators that represent expansions and compressions in the flow direction and gradients normal to the flow direction are implemented for the feature approach. The adaptive approaches are compared for functional outputs of three-dimensional arbitrary Mach number flow simulations on mixed-element unstructured meshes. Grid adaptation is performed with h-refinement and results are presented for inviscid, laminar and turbulent flows. Copyright © 2006 John Wiley & Sons, Ltd.

Received 22 November 2005; Revised 23 July 2006; Accepted 6 August 2006

KEY WORDS: adjoint error estimation; grid adaptation; functional outputs; unstructured meshes

1. INTRODUCTION

In engineering analysis and design, computational fluid dynamics (CFD) is typically used to compute specific quantities that assess the performance of the apparatus under investigation. For example, in a system such as an aircraft wing, these quantities are usually the integral output functions such as the lift and drag coefficients. An accurate estimate of these functional outputs is

*Correspondence to: R. Balasubramanian, Computational Simulation and Design Center, ERC, Mississippi State University, Box 9627, Mississippi State, MS 39762, U.S.A.
†E-mail: raavi@erc.msstate.edu

essential for the design of wings. However, because of the approximations made to the governing partial differential equations, and the compromise between the choice of discretization and available computational resources, there is often a degree of uncertainty in CFD simulations about the accuracy of these computed estimates. This reality forces the design engineer to include a large factor of safety in his design to accommodate for the lack of a reliable error estimator to guide his design process.

Error estimates of the computed outputs are an invaluable commodity to the designer and may be used to make informed decisions about the factor of safety bounds for improving existing design. Also, these error estimates are of immense help to the CFD engineer in providing a quantitative assessment of the functional error (which provide a global measure of the local residual/discretization errors) and may be used to develop an output-based adaptive approach. An output-based adaptive approach may be able to identify regions of the flow that have significant influence on the output functional and will also provide a better understanding and insight into the relevancy of resolving physical features of the flow such as shock waves, stagnation points and separation lines to improve functional accuracy.

The numerical solution of PDEs governing the flow requires discretization of the continuous flow domain into a finite number of elements or volumes. Two approaches, *structured* and *unstructured*, have evolved over the years to discretize the domain [1, 2]. Unstructured grids provide an alternative to structured grid methods because of its inherent flexibility, and its ability to resolve highly complex geometries efficiently. Another major advantage of unstructured grids is grid adaptation [3–6], as the mesh can be locally enriched where needed, without affecting other regions of the mesh. Grid adaptive techniques typically employ local error indicators to identify regions that affect the solution accuracy and locally enrich those regions. A common approach is to use error indicators based on the flow gradients or flow discontinuities. Flow regions that have large solution gradients are resolved with more points and regions of minimal significance are coarsened. This will typically lead to refinement of regions that are of physical significance such as boundary layer, shocks, separation lines, stagnation points, etc. [6–13]. This physics-based feature approach sometimes leads to requests to the grid generator, which result in focused local refinement of certain regions, whereas, the adapted grid may not produce the desired increase in accuracy for a global output functional. Moreover, these adaptive indicators may not be rigorous from an engineering context, where the main concern will be the accuracy of the output functions.

The adjoint (dual) solution describes the sensitivity of the output function to the linearized flow (primal) residuals. By invoking the dual problem, local residual errors resulting from approximation of the solution to the PDEs can be related to the global error in an output function via adjoint variables as weight functions. These error estimates can be used as a correction to produce improved functional estimates. Using the adjoint error correction procedure [14–28], a grid adaptive strategy may be developed to enhance the accuracy of the chosen output to a prescribed tolerance. The adaptive strategy strives to improve the computable error estimates by forming adaptation parameters/indicators based on the level of error in computable error correction. Based on this strategy, a grid adaptive scheme can be implemented that takes into account the error in the primal solution, or both the primal and dual solutions. Becker and Rannacher [26–28] have developed this output-based adaptive procedure by exploiting finite element orthogonality properties and duality concepts. Their adaptation parameter included only the error in primal solution. By invoking the dual (adjoint) problem, Süli [29] and Süli and Houston [30] have performed global error control for adaptive finite element approximations of hyperbolic problems. They found computable error bounds (based on error in primal solution) of linear functional to drive the adaptive algorithm.

DOI: 10.1002/fld

Peraire and collaborators [31, 32] have incorporated an adaptive procedure based on an implicit *a posteriori* procedure for computing upper and lower bounds on functional outputs of finite element solutions. Venditti and Darmofal [23, 33, 34] have enhanced this output-based adaptive procedure by including the error in both primal and dual solutions to form the adaptation parameters. They have compared this procedure with a curvature-based adaptive approach and demonstrated its robustness on finite element and finite volume discretizations. Müller and Giles [35] have employed an alternate output-based strategy based on the computable error estimates. In principle, this leads to an adaptive algorithm for minimizing the magnitude of the correction and hence, not a very robust approach. Balasubramanian [36] has compared the adaptive strategies by Venditti and Darmofal [23, 33, 34] and Müller and Giles [35]. Park [37, 38] has employed the adjoint-based adaptive procedure of Venditti and Darmofal [23, 33, 34] for three dimensional RANS simulations.

In the present study, adjoint-based [23, 33, 34] and feature-based [9–12] grid adaptive approaches are implemented and compared for functional outputs of arbitrary Mach number flow simulations on mixed-element unstructured meshes. The three-dimensional compressible Reynolds-averaged Navier–Stokes equations are solved as described in Reference [39], by introducing preconditioning that is uniformly applicable to Mach numbers ranging from essentially incompressible to supersonic. The one-equation turbulence model of Spalart and Allmaras [40, 41] is used for turbulent flows. The adjoint adaptation procedure requires a smooth reconstruction of the flow and adjoint solutions to a fine-mesh to compute the error indicators. The current study is based on a discrete adjoint formulation [42] and employs the meshless moving least squares (MLS) approximation [43–48] to reconstruct the coarse-mesh flow and adjoint solutions to fine-mesh. All the previous work on three-dimensional adjoint adaptation [37, 38] have been performed on unstructured tetrahedral meshes. The present study strives to support a mixed-element unstructured mesh, comprising of hexahedrals, prisms, pyramids and tetrahedrals. Also, the present study extends the adjoint error estimation procedure for both uniformly refined (obtained by h-refinement of each coarse-mesh element in a fixed ratio, say 1:8) and non-uniformly refined (obtained from a grid generator such as AFLR3 [49] by manually setting the point spacing) fine-meshes. The adaptive mesh library is based on References [50–53] and performs isotropic h-refinement of the elements. Results are presented for inviscid, laminar and turbulent flows.

2. GOVERNING EQUATIONS

2.1. Fluid analysis

The Reynolds-averaged Navier–Stokes equations for three-dimensional, time-dependent, variable Mach flows are presented here in a non-rotating Cartesian co-ordinate system for a finite-volume framework. In the absence of body forces, the non-dimensionalized equations in integral form for a bounded domain Ω, with boundary $\partial\Omega$, can be expressed in primitive variable form [54] as

$$\boldsymbol{M}\boldsymbol{\Gamma}_q^{-1}\frac{\partial}{\partial t}\int_\Omega \boldsymbol{q}\,\mathrm{d}V + \oint_{\partial\Omega}\boldsymbol{F}_i\cdot\hat{n}\,\mathrm{d}S - \oint_{\partial\Omega}\boldsymbol{F}_v\cdot\hat{n}\,\mathrm{d}S = 0 \tag{1}$$

where $\hat{n} = \{n_x, n_y, n_z\}^{\mathrm{T}}$ is the outward pointing unit normal vector to the boundary $\partial\Omega$. The conservative flux formulation is written in terms of primitive variables to facilitate preconditioning.

DOI: 10.1002/fld

The preconditioning matrix

$$\mathbf{\Gamma}_q^{-1} = \mathrm{diag}[1, 1, 1, 1, \beta(M_r)] \tag{2}$$

is a constant diagonal matrix that only depends on the reference Mach number (M_r) [39]. In Equation (1), $\boldsymbol{M} = [\partial \boldsymbol{Q}/\partial \boldsymbol{q}]$ is a transformation matrix from conservative variables $\boldsymbol{Q} = \{\rho, \rho u, \rho v, \rho w, \rho e_t\}^{\mathrm{T}}$ to primitive variables $\boldsymbol{q} = \{\rho, u, v, w, p\}^{\mathrm{T}}$. Here, ρ is the density; u, v, and w are the components of velocity in the x, y, and z directions, respectively; p is the pressure; e_t is the specific total energy; and $\boldsymbol{F}_i$, $\boldsymbol{F}_v$ are the inviscid and viscous flux vectors.

The flow solver used in the present study is a node-based, finite volume implicit scheme built on an unstructured grid framework capable of handling mixed elements [54, 55]. The solver uses upwind differencing to discretize the convective terms and the modified Roe's flux-difference scheme described in References [39, 56] to evaluate the fluxes. The viscous flux terms are evaluated using the edge-based positive scheme presented in Reference [54]. The one-equation turbulence model of Spalart and Allmaras [40, 41] is used for turbulent flows. The reader is referred to References [54, 55] for a detailed description of the discretization and solution methodology.

2.2. *Discrete adjoint analysis*

Consider the following form of steady-state non-linear governing equations, where Q represent the steady-state solution vector, X, the grid co-ordinates and α_k, the set of design variables. The discrete residual vector R at steady state is given by

$$R(Q, X, \alpha_k) \equiv R_I(Q, X, \alpha_k) + B(Q, X, \alpha_k) = 0 \tag{3}$$

Here, R_I is the discretized residual at the interior and B, the residual at the boundaries (accounting for the boundary conditions). Let $F(Q, X, \alpha_k)$ represent the cost/output function of interest. For the discrete adjoint formulation, the output function of interest may be augmented with the non-linear discrete flow equations via adjoint variables γ as

$$F(Q, X, \alpha_k) = F(Q, X, \alpha_k) + \{\gamma\}^{\mathrm{T}} R(Q, X, \alpha_k) \tag{4}$$

Equation (4) results from the fact that for a steady solution the residual vector is zero. Hence, the inner product of any vector with the residual vector must also be zero. Linearizing Equation (4) yields

$$\nabla F = \{\gamma\}^{\mathrm{T}} \left\{ \frac{\partial R}{\partial X} \frac{\partial X}{\partial \alpha_k} + \frac{\partial R}{\partial \alpha_k} \right\} + \left\{ \frac{\partial F}{\partial X} \frac{\partial X}{\partial \alpha_k} + \frac{\partial F}{\partial \alpha_k} \right\} + \left(\left\{ \frac{\partial F}{\partial Q} \right\} + \{\gamma\}^{\mathrm{T}} \left[\frac{\partial R}{\partial Q} \right] \right) \left\{ \frac{\mathrm{d}Q}{\mathrm{d}\alpha_k} \right\} \tag{5}$$

Since $\{\gamma\}$ is yet to be defined, it may be chosen to force the coefficients of $\{\partial Q/\partial \alpha_k\}$ to be zero. The equation for adjoint variables is given by

$$\left[\frac{\partial R}{\partial Q} \right]^{\mathrm{T}} \{\gamma\} = - \left\{ \frac{\partial F}{\partial Q} \right\}^{\mathrm{T}} \tag{6}$$

The interested reader is referred to References [42, 57] for a complete discussion on the solution procedure, linearization of flux limiters, parallel implementation and demonstration of consistency of linearization.

DOI: 10.1002/fld

3. ADJOINT ERROR ESTIMATION

Typically, in engineering applications of CFD, an accurate estimate of the output function $F(Q)$ is desired. But often, a compromise must be made between the fidelity of solution obtained and the available resources. To elaborate on this, consider discretization of the computational domain (Ω) using a coarse-mesh Ω_H and a fine-mesh Ω_h. H and h ($H>h$) may represent suitably defined length scales based on the approximation such as finite difference, finite element or finite volume. Let $R_H(Q_H)$ and $R_h(Q_h)$ be the non-linear residual vectors obtained by discretization of the flow equations on Ω_H and Ω_h. Let $F_H(Q_H)$ and $F_h(Q_h)$ be estimates of $F(Q)$ from Ω_H and Ω_h. $F_H(Q_H)$ and $F_h(Q_h)$ are evaluated using Q_H and Q_h, the discrete solutions on Ω_H and Ω_h, respectively. The coarse-mesh Ω_H is affordable in terms of memory and computation time. However, the estimate $F_H(Q_H)$ may not be accurate enough for engineering applications. The fine-mesh estimate $F_h(Q_h)$ may satisfy the desired accuracy criteria, but is prohibitively expensive to compute. A computationally efficient error correction procedure [14–17, 21–23] is introduced that produce improved estimates of output functions without ever solving on the fine-mesh Ω_h.

By introducing a perturbation δQ_h to the fine-mesh solution Q_h, and performing a Taylor's series expansion of $F_h(Q_h+\delta Q_h)$ and $R_h(Q_h+\delta Q_h)$ yields

$$F_h(Q_h+\delta Q_h)=F_h(Q_h)+\left\{\frac{\partial F_h}{\partial Q_h}\right\}\delta Q_h+\cdots \tag{7}$$

$$R_h(Q_h+\delta Q_h)=R_h(Q_h)+\left[\frac{\partial R_h}{\partial Q_h}\right]\delta Q_h+\cdots \tag{8}$$

Now, defining $\delta Q_h = Q_H^h - Q_h$, where Q_H^h is the solution at fine-mesh obtained through prolongation of coarse-mesh solution Q_H, Equations (7) and (8) can be written as

$$F_h(Q_h)=F_h(Q_H^h)+\left\{\frac{\partial F_h}{\partial Q_h}\right\}_{Q_H^h}(Q_h-Q_H^h)+\cdots \tag{9}$$

$$R_h(Q_h)=R_h(Q_H^h)+\left[\frac{\partial R_h}{\partial Q_h}\right]_{Q_H^h}(Q_h-Q_H^h)+\cdots \tag{10}$$

Here, $F_h(Q_H^h)$ is the fine-mesh estimate of the function evaluated using Q_H^h; $\{\partial F_h/\partial Q_h\}_{Q_H^h}$ is the linear sensitivities of the fine-mesh function with respect to Q_H^h; $R_h(Q_H^h)$ is the residual vector evaluated at the fine-mesh using Q_H^h; and $[\partial R_h/\partial Q_h]_{Q_H^h}$ is the fine-mesh Jacobian evaluated using Q_H^h. Q_H^h is given by

$$Q_H^h = P_H^h\, Q_H \tag{11}$$

and P_H^h is a suitably defined prolongation operator.

For a steady-state problem, $R_h(Q_h)=0$. Assuming the well-posedness of Equation (10) yields

$$(Q_h-Q_H^h)\approx -\left[\frac{\partial R_h}{\partial Q_h}\right]^{-1}_{Q_H^h} R_h(Q_H^h) \tag{12}$$

DOI: 10.1002/fld

Substituting Equation (12) in Equation (9)

$$F_h(Q_h) \approx F_h(Q_H^h) - \left\{\frac{\partial F_h}{\partial Q_h}\right\}_{Q_H^h} \left[\frac{\partial R_h}{\partial Q_h}\right]^{-1}_{Q_H^h} R_h(Q_H^h) \tag{13}$$

$$\approx F_h(Q_H^h) + \{\gamma_h\}^{\mathrm{T}}_{Q_H^h} R_h(Q_H^h) \tag{14}$$

where $\{\gamma_h\}^{\mathrm{T}}_{Q_H^h}$ is the adjoint solution vector at the fine-mesh evaluated using Q_H^h. The adjoint equation for $\{\gamma_h\}_{Q_H^h}$ can be written as

$$\left[\frac{\partial R_h}{\partial Q_h}\right]^{\mathrm{T}}_{Q_H^h} \{\gamma_h\}_{Q_H^h} = -\left\{\frac{\partial F_h}{\partial Q_h}\right\}^{\mathrm{T}}_{Q_H^h} \tag{15}$$

To avoid the need for computing $\{\gamma_h\}^{\mathrm{T}}_{Q_H^h}$ on the fine-mesh, it may also be approximated through interpolation as

$$\{\gamma_h\}_{Q_H^h} \approx \gamma_H^h = P_H^h \gamma_H \tag{16}$$

where γ_H is the adjoint solution at the coarse-mesh given by Equation (6). The computable estimate of the output function is given by

$$F(Q) = F_h(Q_H^h) + \{\gamma_H^h\}^{\mathrm{T}} R_h(Q_H^h) \tag{17}$$

In the above expression, $\{\gamma_H^h\}^{\mathrm{T}} R_h(Q_H^h)$ is the *error correction* term that approximates the error in output function as the inner product of the adjoint solution and the primal residual error. The auxiliary computations needed by this procedure are: prolongation of coarse-mesh flow and adjoint solutions to the fine-mesh; and functional and residual evaluations on the fine-mesh.

3.1. Prolongation operators

The prolongation operation given by Equation (11) can be expanded as

$$Q_H^h = P_H^h Q_H \equiv \sum_{I=1}^{n} \phi_I^k (Q_H)_I \tag{18}$$

where $\Phi^k = \{\phi_1^k, \phi_2^k, \ldots, \phi_n^k\}$ are the MLS shape functions, k is the order of the basis function, and n is size of the MLS support stencil. The MLS shape functions $\Phi^k = \{\phi_1^k, \phi_2^k, \ldots, \phi_n^k\}$ are obtained by a least squares minimization problem described in Appendix A. Linear and quadratic basis functions are considered with cubic spline [47] weights. Weight functions are defined with compact circular or rectangular supports [47, 57]. Once the supports are built, weights are applied, either isotropically as radial weights, or anisotropically as tensor product weights. Mixed supports are also defined; i.e. circular support with isotropic weights for tetrahedrals and pyramids (inviscid regions) and rectangular support with anisotropic weights for prisms and hexahedrals (boundary layer regions).

For tetrahedral meshes the stencil is built using circular supports. For mixed-element meshes, circular supports are used to build the stencil in regions of tetrahedrals and pyramids, and rectangular

DOI: 10.1002/fld

supports are used in regions of prisms and hexahedrals. The interested reader is referred to Reference [57] for a detailed discussion on the MLS approximation and the approach to build a compact support.

4. ADAPTIVE INDICATORS

4.1. Adjoint-based approach

The present output-based adaptive strategy suggested by Venditti and Darmofal [28, 33, 34] is based on the adjoint error correction procedure described in Section 3. The adaptive procedure strives to improve the error estimates $\{\gamma_H^h\}^{\mathrm{T}} R_h(Q_H^h)$ by reducing the level of error in the computable error correction. By including the error in computable estimates, Equation (14) can be written as

$$F(Q) - F_h(Q_H^h) \approx \{\gamma_H^h\}^{\mathrm{T}} R_h(Q_H^h) + \{\gamma_h - \gamma_H^h\}^{\mathrm{T}} R_h(Q_H^h) \tag{19}$$

In the above equation, the first term on the right-hand side is the computable error correction and the second term is the error in computable correction. The relationship between the primal (flow) and dual (adjoint) problem gives rise to another expression for the second term. Neglecting the non-linear effects, the second term can be written as

$$(\gamma_h - \gamma_H^h)^{\mathrm{T}} R_h(Q_H^h) \approx R_h^{\gamma}(\gamma_H^h)(Q_h - Q_H^h) \tag{20}$$

where $R_h^{\gamma}(\gamma_H^h)$ is the adjoint residual given by

$$R_h^{\gamma}(\gamma) = \left[\frac{\partial R_h}{\partial Q_h}\right]^{\mathrm{T}}_{Q_H^h} \{\gamma\} + \left\{\frac{\partial F_h}{\partial Q_h}\right\}_{Q_H^h} \tag{21}$$

A conservative adaptive indicator can be formed by including the errors in computing the adjoint solution also in the formulation. The error indicator (E_I) formed by including both the primal and adjoint residual errors is

$$E_I = \frac{(\gamma_h - \gamma_H^h)^{\mathrm{T}} R_h(Q_H^h) + R_h^{\gamma}(\gamma_H^h)(Q_h - Q_H^h)}{2} \tag{22}$$

The error intensity (E_I) at each fine-mesh node i can be evaluated as

$$(E_I)_i = \frac{\{|(\gamma_h - \gamma_H^h)_i^{\mathrm{T}} R_h(Q_H^h)_i| + |R_h^{\gamma}(\gamma_H^h)_i (Q_h - Q_H^h)_i|\}}{2} \tag{23}$$

By approximating

$$[\gamma_h - \gamma_H^h] \approx [(\gamma_H^h)^{\mathrm{HO}} - (\gamma_H^h)^{\mathrm{LO}}] \equiv [\gamma^{\mathrm{HO}} - \gamma^{\mathrm{LO}}]$$

$$[Q_h - Q_H^h] \approx [(Q_H^h)^{\mathrm{HO}} - (Q_H^h)^{\mathrm{LO}}] \equiv [Q^{\mathrm{HO}} - Q^{\mathrm{LO}}]$$

Equation (23) can be written as

$$(E_I)_i = \frac{|(\gamma^{\mathrm{HO}} - \gamma^{\mathrm{LO}})_i^{\mathrm{T}} R_h(Q^{\mathrm{HO}})_i| + |R_h^{\gamma}(\gamma^{\mathrm{HO}})_i (Q^{\mathrm{HO}} - Q^{\mathrm{LO}})_i|}{2} \tag{24}$$

DOI: 10.1002/fld

where γ^{HO}, Q^{HO} and γ^{LO}, Q^{LO} are the higher-order and lower-order prolongated adjoint and flow solutions. Higher-order prolongation is performed using quadratic basis and lower-order prolongation is achieved using linear basis. This has advantages computationally, as the need to solve for the adjoint solution at the fine-mesh is avoided and the accuracy is not affected significantly. The only computational costs on this larger mesh are function evaluations, flow and adjoint residual evaluations, and dot product of vectors.

The formation of adaptation parameters at the coarse-mesh from the error intensities at the fine-mesh differs slightly for uniformly and non-uniformly refined fine-meshes, but, the underlying principle is essentially the same. The adaptation parameter A_p^1 for a coarse-mesh node k can be obtained from a uniformly refined fine-mesh by looping over all the coarse-mesh edges surrounding node k, and adding one-half of the error intensities (E_I) from each of the embedded fine-mesh nodes located at the midpoint of these edges. The adaptation parameter A_p^1 at node k is given by

$$(A_p^1)_k = \sum_{j=1}^{n(k)} \frac{(E_I)_j}{2} \tag{25}$$

where $n(k)$ is the summation over all the embedded fine-mesh nodes (which are at the midpoint of the edges surrounding node k). The adaptation parameters can be obtained from a non-uniformly refined fine-mesh by looping over all the coarse-mesh elements that contain fine-mesh nodes and split the error intensity E_I between all the nodes that make the element. The adaptation parameter A_p^1 at node k is given by

$$(A_p^1)_k = \sum_{i=1}^{e(k)} \sum_{j=1}^{n(i)} \frac{(E_I)_j}{d_i} \tag{26}$$

where $e(k)$ is the number of coarse-mesh elements incident at node k, $n(i)$ is the number of fine-mesh nodes contained by element i and d_i is the element size (number of nodes that make element i).

4.2. Feature-based approach

The second adaptive strategy is based on feature detection [9–12] and strives to identify and resolve the significant features of the flow. The adaptation parameter (A_p^2) can be defined as

$$A_p^2 = \{e_1, e_2, e_3\} \tag{27}$$

where e_1, e_2, e_3 are the error indicators given by

$$e_1 = \max\left[-\frac{\mathbf{V} \cdot \text{grad}\, Q}{|\mathbf{V}|}, 0\right] \tag{28}$$

$$e_2 = \max\left[+\frac{\mathbf{V} \cdot \text{grad}\, Q}{|\mathbf{V}|}, 0\right] \tag{29}$$

$$e_3 = \left|\text{grad}\, Q - \frac{\mathbf{V}}{|\mathbf{V}|}\left(\frac{\mathbf{V} \cdot \text{grad}\, Q}{|\mathbf{V}|}\right)\right| \tag{30}$$

DOI: 10.1002/fld

$|\cdot|$ represent the magnitude, Q is any suitable flow property and $\mathbf{V}$ denotes the velocity vector. Each of these error indicators can isolate a particular type of feature. The first two error indicators represent expansions and compressions in the flow direction and the third represents gradients normal to the flow direction [10]. At viscous boundaries, A_p^2 is defined only by the magnitude of the gradient of Q ($|\text{grad}\, Q|$) because of the no-slip boundary conditions ($\mathbf{V}=0$).

5. ADAPTATION MECHANICS

Simple adaptation mechanics are employed in the present study. The adaptation procedure employed in the present study constitutes two stages: (1) formation of element-adaptation flags; (2) h-refinement.

The adaptation parameters A_p^1 or A_p^2 given in Equations (25)–(27) are formed at all the coarse-mesh nodes. These nodal values are transferred to the elements by a simple averaging. The mean (μ) and standard deviation (σ) of the adaptation parameters over all the coarse-mesh elements are computed and elements are flagged for refinement, if their adaptation parameter is greater than a proposed error limit (e_{lim}) given by

$$e_{\text{lim}} = \mu + \text{crel} * \sigma \tag{31}$$

where crel is a relaxation factor usually greater than 0.5. For adaptation parameter A_p^2, each error indicator is treated independently, allowing particular features in the flow field to be isolated.

An unstructured mesh refinement module is developed using Python and C++ to perform adaptation. The adaptive mesh library is based on References [50–52] and performs isotropic h-refinement of the elements. The refinement template [53, 57] controls the pattern of subdivision of the mesh elements. The reader is referred to Reference [57] for the refinement patterns of different element types used in the study. In the present study, only node addition is considered and mesh coarsening is not implemented. h-Refinement is performed in four steps:

1. isotropically refine all flagged elements.
2. loop over all elements and identify the refinement pattern for each element by adding new nodes if needed; this step should be repeated till no new nodes are added and all the elements have valid refinement patterns.
3. split the element based on its refinement pattern.
4. perform boundary projection.

For h-refined tetrahedral meshes, quality improvement by local reconnection and Laplacian smoothing is performed using AFLR3 [49]. No quality improvement is performed on mixed-element meshes. In the present study, the mixed-element meshes from AFLR3 [49] typically have prisms in the boundary layer, tetrahedrals in the inviscid regions and pyramids in the transition region. Anisotropic refinement of prisms resulted in poor quality pyramids and tetrahedrals in the boundary layer. This is not desirable and hence, anisotropic refinement of prisms is not supported in the present study. Because of this, no refinement is allowed in the streamwise direction of boundary layer, especially for prisms. If streamwise refinement is allowed, this resulted in the refinement of the whole layer to preserve the shape of prism.

The adaptation process create boundary nodes at the midpoint of the edges, by a simple averaging of the node co-ordinates. Boundary node projection is accomplished by using a transfinite, visually

DOI: 10.1002/fld

continuous, triangular interpolant explained in References [5, 58]. The interpolant is based on side-vertex interpolation in triangles [58]. The cubic Hermite interpolant uses outward surface normals to reconstruct the surface and the resulting reconstructed surface is a G^1 representation with a continuously varying outward normal vector. The reader is referred to Reference [5] for a derivation of the interpolant. This is a slightly better approximation than the simple averaging of the nodes. In boundary layer regions if the projected boundary node resulted in negative volumes, it is replaced with the averaged value.

6. RESULTS AND DISCUSSIONS

Grid adaptation is performed using adjoint-based and feature-based approaches and the improvement in functional estimates is observed. For adjoint-based adaptation, the adaptive indicators are evaluated by prolongating the flow and adjoint solutions to a fine-mesh using the MLS approximation described in appendix. The fine-mesh is obtained by uniform or non-uniform refinement.

6.1. Inviscid flow

The first test case is inviscid flow over an Onera M6 wing at an angle of attack of 3.06° and Mach number of 0.8395. Adjoint-based adaptation is performed for both lift (C_L) and drag (C_D) coefficients on the wing and compared with feature-based adaptation. A lambda shock is typical of these flow conditions and the estimates of C_L and C_D are largely dependent on the accurate prediction of the shock and its location. For the adjoint-based adaptation, the non-uniformly refined fine-mesh with 976 503 nodes, 286 728 surface triangles and 5 372 918 tetrahedrals is used to establish the adaptive indicators.

Figures 1(a) and (b) show the convergence of C_D and C_L after four adaptive iterations. The adjoint-C_D adaptation reaches the finest-mesh estimate from non-uniform refinement in two iterations with less than 200 000 nodes compared to 976 503 nodes at the finest-mesh; a factor 5 reduction in mesh size for the same C_D value. Also, the C_D value from the final adjoint-C_D adaptation reaches the uniformly refined finest-mesh estimate with less than 400 000 nodes compared to approximately 2 000 000 nodes at the finest-mesh; again, a factor 5 reduction in mesh size for the same C_D value. Adjoint adaptation with correction achieves super-convergent estimates and may be converging to the asymptotic value for C_D. This can be verified by observing the C_D values from uniform refinement. The extrapolated value is obtained from a linear fit of the uniformly refined mesh C_D values similar to References [37, 38]. The feature-based adaptation is converging to an erroneous value significantly larger than the finest-mesh C_D. From Figure 1(a), it can be observed that the C_D calculation from adjoint-C_L adaptation is better than feature-based adaptation, but, slightly inferior to adjoint-C_D adaptation. It can be noticed from Figure 1(b) that C_L is better predicted by the adjoint approach, and the feature approach is converging to a lower estimate of C_L. The error corrected C_L estimate from adjoint adaptation has achieved the finest-mesh value in two iterations. Also, the C_L calculation from adjoint-C_L and C_D adaptations compare favourably.

The initial and adjoint-C_D adapted grids are shown in Figures 2(a) and (b). There is significant refinement near the leading and trailing edges and at the lambda shock location. Figure 3(a) shows the initial grid on the upper wing surface. The surface grids from adjoint-C_D and feature adaptations are shown in Figures 4(a) and 5(a). The feature-based adapted grid has excessive refinement near the leading and trailing edges, but, have poor shock resolution. The poor C_D results from the feature

DOI: 10.1002/fld

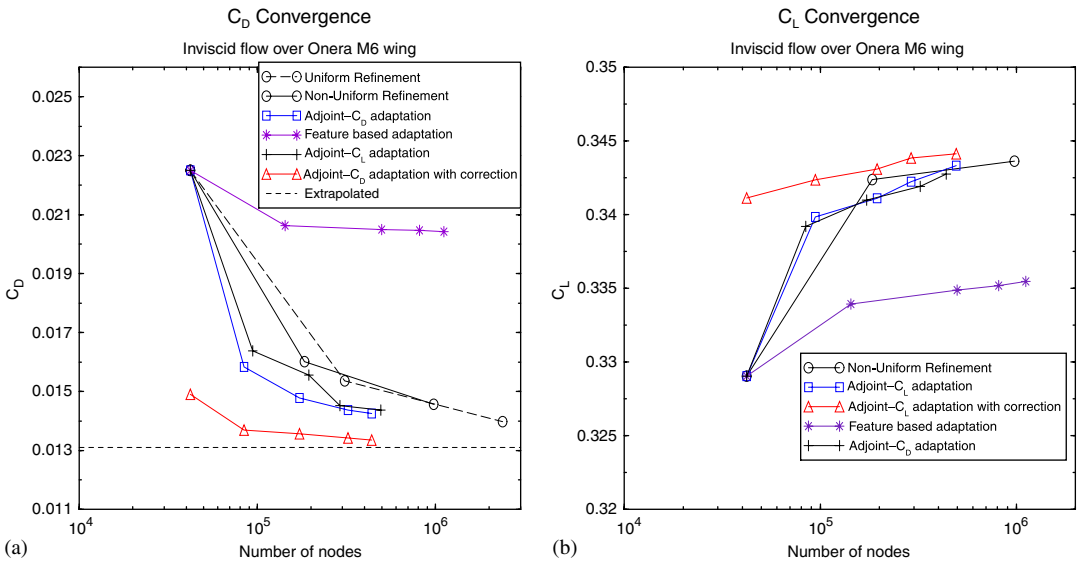

Figure 1. Inviscid flow over Onera M6 wing: $M_\infty = 0.8395$, $AOA = 3.06°$. C_D and C_L convergence: (a) C_D vs number of nodes; correction computed with isotropic cubic spline weights and linear–quadratic basis; and (b) C_L vs number of nodes; correction computed with isotropic cubic spline weights and quadratic–quadratic basis.

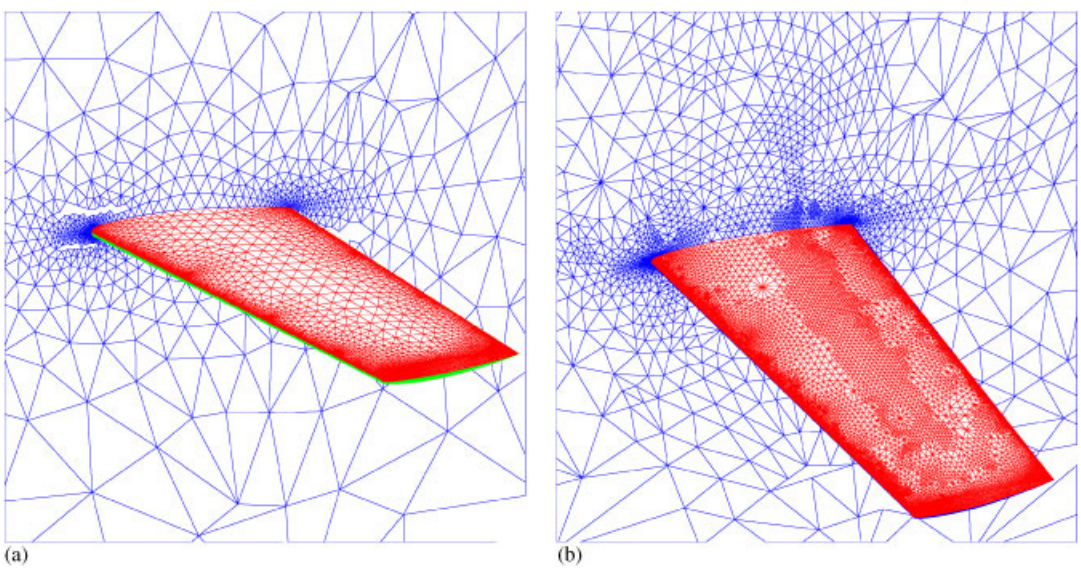

Figure 2. Inviscid flow over Onera M6 wing: $M_\infty = 0.8395$, $AOA = 3.06°$: (a) initial Onera M6 wing grid with 42 114 nodes; and (b) Onera M6 wing grid after two adjoint-C_D adaptations with 172 299 nodes.

DOI: 10.1002/fld

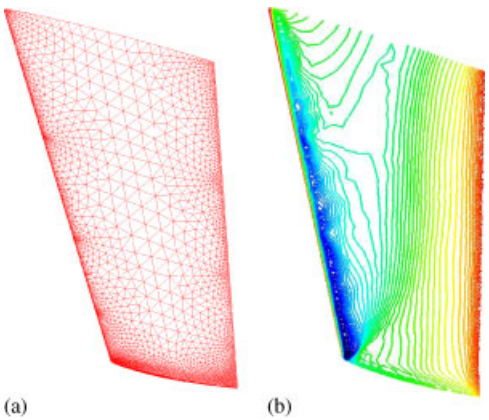

Figure 3. Inviscid flow over Onera M6 wing: $M_\infty = 0.8395$, $AOA = 3.06°$: (a) initial surface grid on upper wing (total nodes: 42 114); and (b) density contours on upper wing for initial grid.

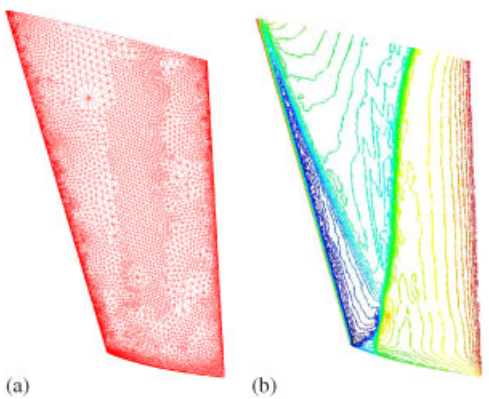

Figure 4. Inviscid flow over Onera M6 wing: $M_\infty = 0.8395$, $AOA = 3.06°$: (a) surface grid on upper wing after two adjoint-C_D adaptations (total nodes: 172 299); and (b) density contours on upper wing for adjoint-C_D adapted grid.

approach may be attributed to the failure to resolve the lambda shock. A look at the initial density contours in Figure 3(b) can explain this behaviour. The initial grid has resolved the shock poorly and the feature-based approach may need a well resolved initial grid to identify the features. The resolution of the initial grid did not pose a problem for the adjoint approach. The density contours on the upper wing surface of the adjoint and feature-adapted grids are shown in Figures 4(b) and 5(b). The lambda shock is clearly visible and captured well by the adjoint approach. However, the feature approach has failed to resolve the shock in four iterations of adaptation. Figure 6(a) shows the C_L adaptation contours on the upper wing surface for the first adaptation. The adapted grid after two C_L adaptations is shown in Figure 6(b) for the upper wing surface. It can be observed that there is considerable refinement near the lambda shock and near the leading and trailing edges of the wing. The C_p distributions from the initial, final adjoint-adapted, and feature-adapted grids

DOI: 10.1002/fld

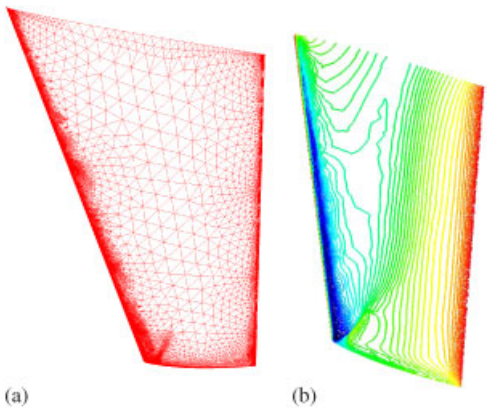

Figure 5. Inviscid flow over Onera M6 wing: $M_\infty = 0.8395$, $AOA = 3.06°$: (a) surface grid on upper wing after two feature adaptations (total nodes: 498 863); and (b) density contours on upper wing for feature-adapted grid.

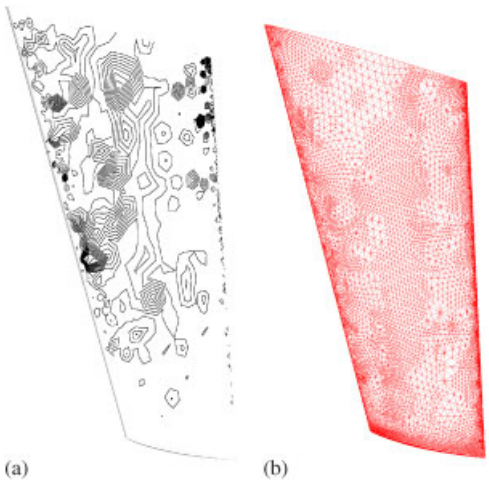

Figure 6. Inviscid flow over Onera M6 wing: $M_\infty = 0.8395$, $AOA = 3.06°$: (a) adjoint-C_L adaptation parameters on upper wing surface in the initial grid; and (b) surface grid on upper wing after two adjoint-C_L adaptations (total nodes: 177 540).

at span wise locations $z/c = 0.65$ and 0.9 are compared with experimental data [59] in Figures 7(a) and (b). From Figures 7(a) and (b), it can be observed that there is excellent agreement between the C_p distributions from the adjoint-adapted grids and experimental data, whereas, the feature approach has failed to resolve the lambda shock.

6.2. *Viscous laminar flow*

For the viscous case, only adjoint adaptation is performed and uniformly refined grids are used to evaluate the adaptive indicators. The viscous test case is laminar flow over a unit cylinder

DOI: 10.1002/fld

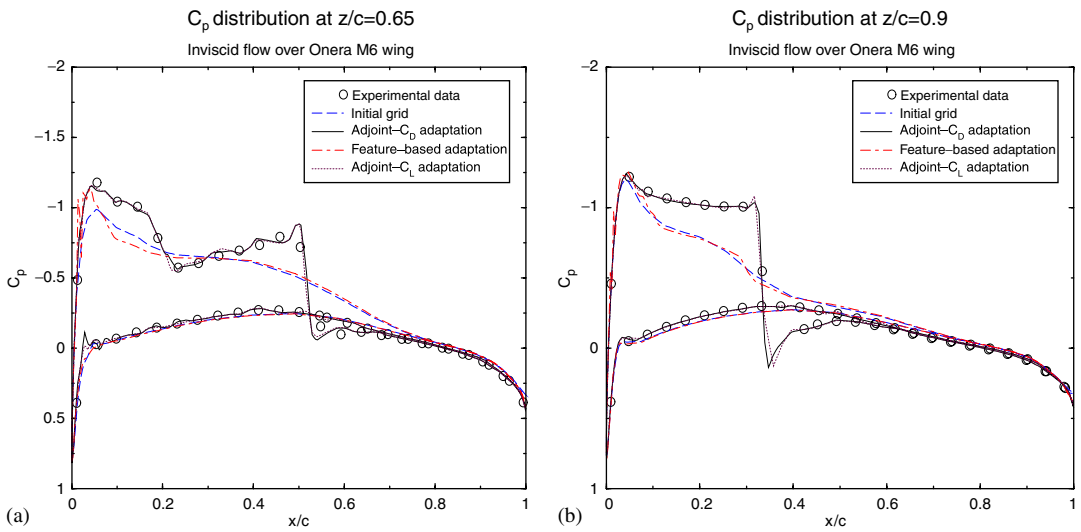

Figure 7. Inviscid flow over Onera M6 wing: $M_\infty = 0.8395$, $AOA = 3.06°$: (a) C_p distribution on the wing at span wise location $z/c = 0.65$; and (b) C_p distribution on the wing at span wise location $z/c = 0.9$.

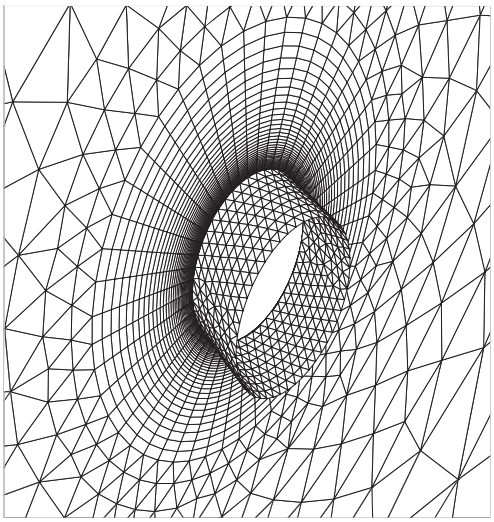

Figure 8. Laminar flow over a cylinder: $M_\infty = 0.1$, $Re = 20$. Initial cylinder grid with the near symmetry plane removed.

at a Mach number of 0.1 and Reynolds number of 20. The cylinder is capped with symmetry planes at both ends and has a height of half its diameter. The initial cylinder grid shown in Figure 8 contains 22 242 nodes, 4608 surface quadrilaterals, 2520 surface triangles, 36 864 prisms and 5618 tetrahedrals. Adaptation is performed for C_D using the adjoint approach. The

DOI: 10.1002/fld

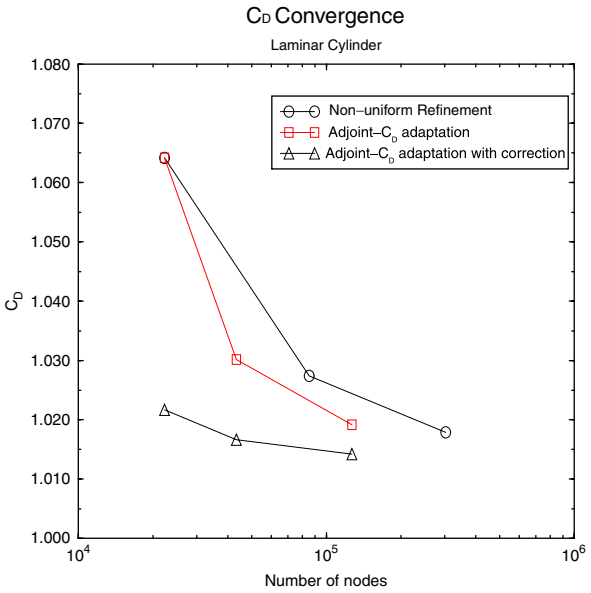

Figure 9. Laminar flow over a cylinder: $M_\infty = 0.1$, $Re = 20$. C_D convergence. Correction computed with mixed cubic spline weights and quadratic–quadratic basis.

convergence of C_D is shown in Figure 9. The adjoint-based adaptation converges to the finest-mesh estimate in two iterations. The adapted grid has 126 812 nodes compared to 303 152 nodes for the finest-mesh; a factor 2 reduction in mesh size is achieved for the same C_D value. A more accurate estimate of C_D is attained by combining error correction with adaptation if it is assumed that the exact drag value is lower than the finest uniform refinement grid calculation.

The initial and final adapted symmetry plane grids are shown in Figures 10(a) and (b). Near field views of the symmetry plane grids are shown in Figures 11(a) and (b). From Figure 10(b), it can be observed that there is significant refinement in the front and wake of cylinder. The wake regions are always a source of drag, and it can be noticed that the adjoint-based adaptation has identified these regions for enrichment. The leading edge stagnation point and the regions of flow acceleration near the top and bottom of the cylinder have been considerably refined. These are the regions of the flow where pressure changes rapidly in the streamwise direction. This is confirmed by a look at the pressure contours on the symmetry plane of the initial grid in Figure 12(a). The pressure contours on the symmetry plane of the final adapted grid is shown in Figure 12(b), and it can be noticed that the pressure contours are symmetric and are better resolved. The velocity magnitude ($U = \sqrt{u^2 + v^2 + w^2}$) contours on the farther symmetry plane for the initial and adapted grids are shown in Figures 13(a) and (b). From the Figures 13(a) and (b), it can be observed that the shear layer is better resolved in the adapted grid in the wake regions. The tangential component of velocity vector for the adapted grid is shown in Figure 14. The attached symmetric vortices behind the cylinder can be observed.

DOI: 10.1002/fld

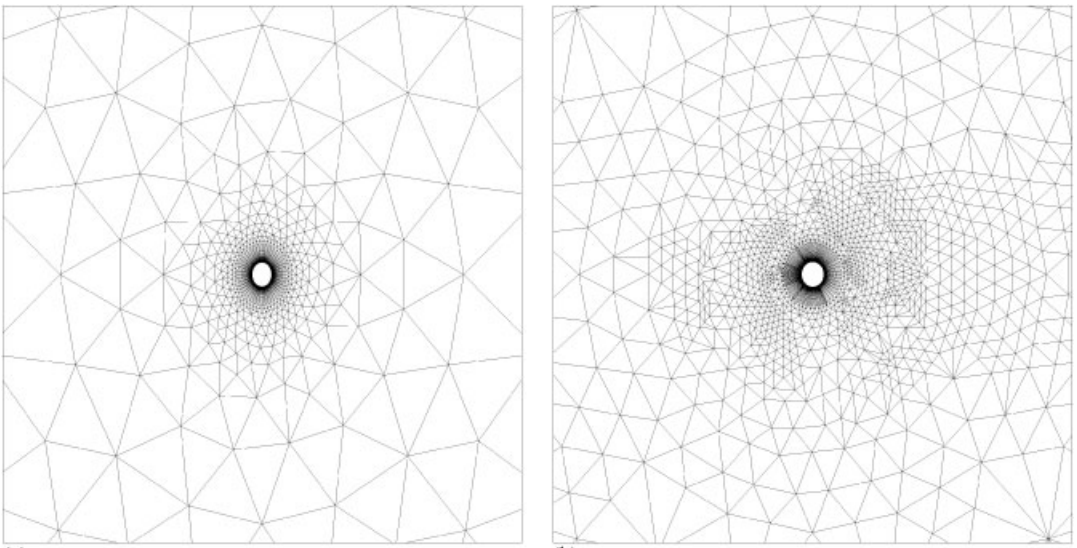

Figure 10. Laminar flow over a cylinder: $M_\infty = 0.1$, $Re = 20$: (a) initial symmetry plane grid (total nodes: 22 242); and (b) symmetry plane grid after two adjoint-C_D adaptations (total nodes: 126 812).

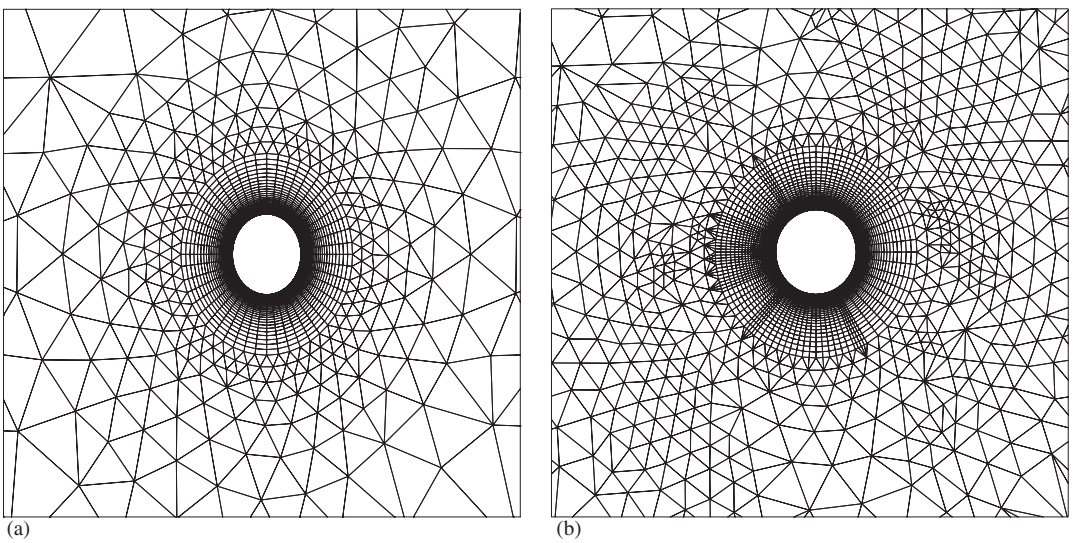

Figure 11. Laminar flow over a cylinder: $M_\infty = 0.1$, $Re = 20$: (a) near field view of initial symmetry plane grid; and (b) near field view of adjoint-adapted symmetry plane grid.

6.3. Turbulent flow

Turbulent flow is simulated over a NACA 0012 rectangular wing at a Mach number of 0.95, chord-based Reynolds number of 3 000 000 and angle of attack of 0°. The test case is a supercritical flow with strong shocks on the upper and lower surfaces of the wing. There is shock/boundary layer

DOI: 10.1002/fld

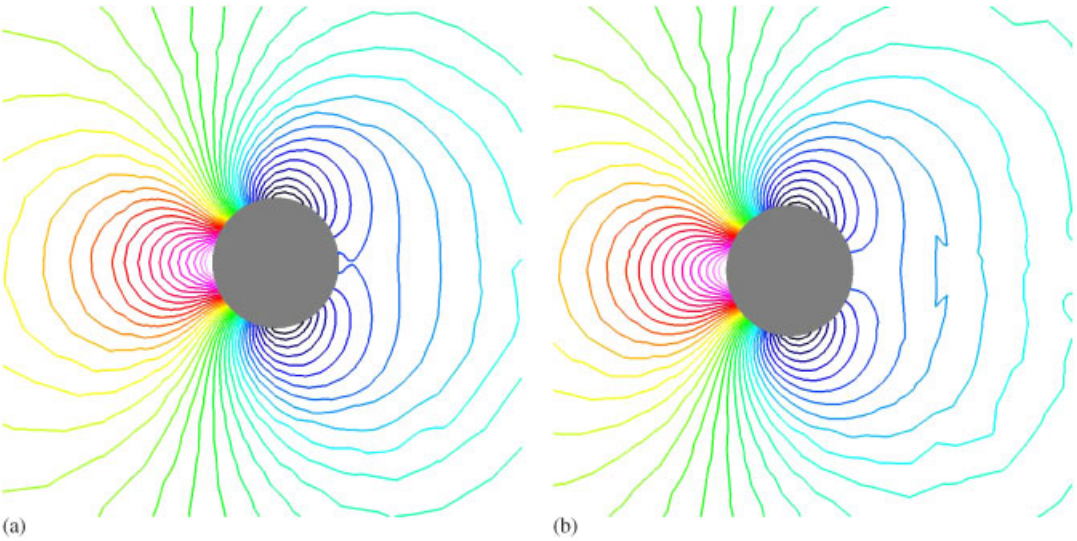

Figure 12. Laminar flow over a cylinder: $M_\infty = 0.1$, $Re = 20$: (a) pressure contours on the symmetry plane for initial grid; and (b) pressure contours on the symmetry plane for adjoint-adapted grid.

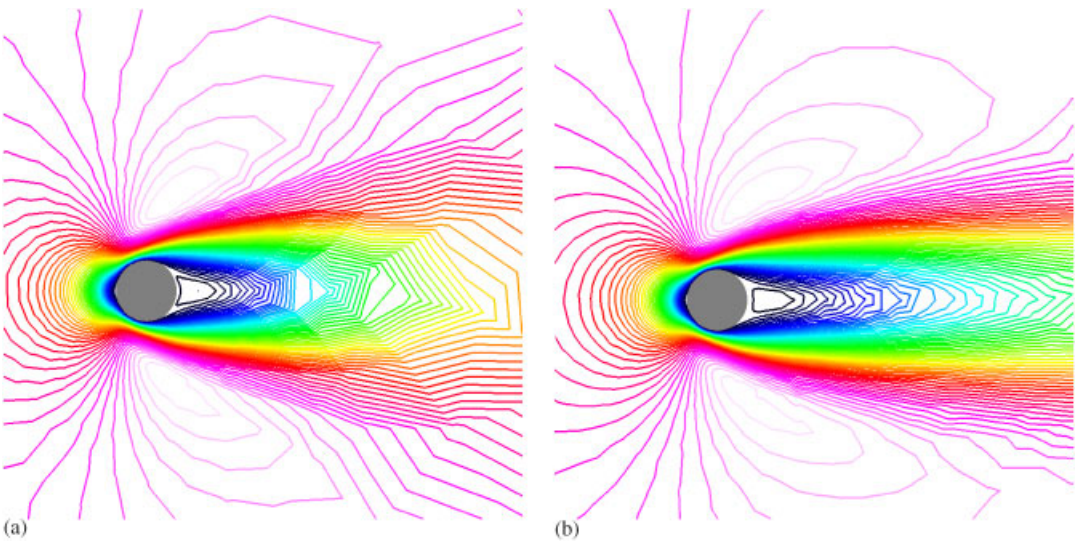

Figure 13. Laminar flow over a cylinder: $M_\infty = 0.1$, $Re = 20$: (a) velocity magnitude contours on the symmetry plane for initial grid; and (b) velocity magnitude contours on the symmetry plane for adjoint-adapted grid.

interaction and because of the presence of strong shocks, the boundary layer thickens and separates on the upper and lower wing surfaces. The output function considered is the drag coefficient C_D on the wing. The estimates of C_D are largely dependent on the accurate prediction of the upper and lower wing shocks, especially their locations, and adequate resolution of the separation zones

DOI: 10.1002/fld

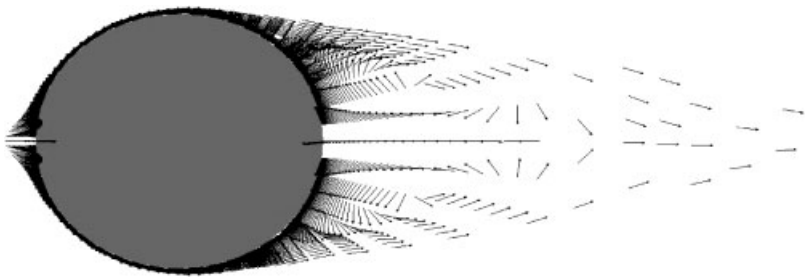

Figure 14. Laminar flow over a cylinder: $M_\infty = 0.1$, $Re = 20$. Tangential component of velocity vector on the symmetry plane for adjoint-adapted grid.

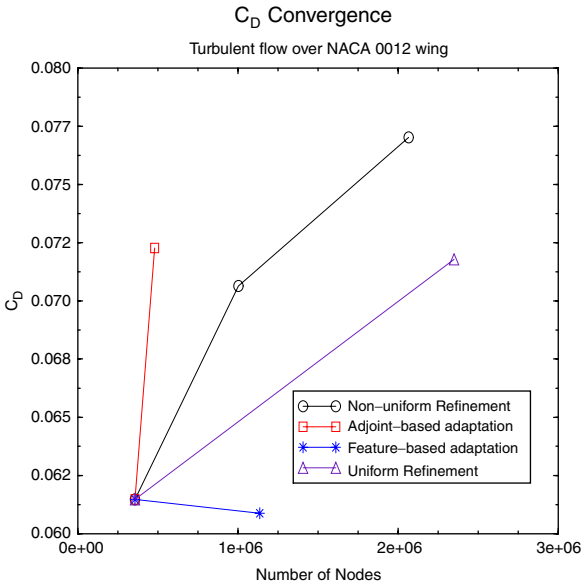

Figure 15. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. C_D convergence.

in the boundary layer. The initial grid contains 356 420 nodes, 2576 surface quadrilaterals, 23 072 surface triangles, 611 744 prisms and 258 874 tetrahedrals with a wall spacing of 8×10^{-06} of the mean aerodynamic chord. For the adjoint-based adaptation, the non-uniformly refined intermediate fine-mesh with 1 003 430 nodes, 4570 surface quadrilaterals, 71 972 surface triangles, 1 706 813 prisms and 767 814 tetrahedrals is used to establish the adaptive indicators. The wall spacing on the fine-mesh is same as the initial grid. Only a single iteration of adaptation is performed for the turbulent case, because of memory constraints.

The convergence of C_D is shown in Figure 15. The estimate of C_D from the adjoint-adapted grid with 478 952 nodes is better than the estimates from the non-uniformly refined fine-mesh with 1 003 430 nodes, and the uniformly refined mesh with 2.35×10^6 nodes. Also, note that

DOI: 10.1002/fld

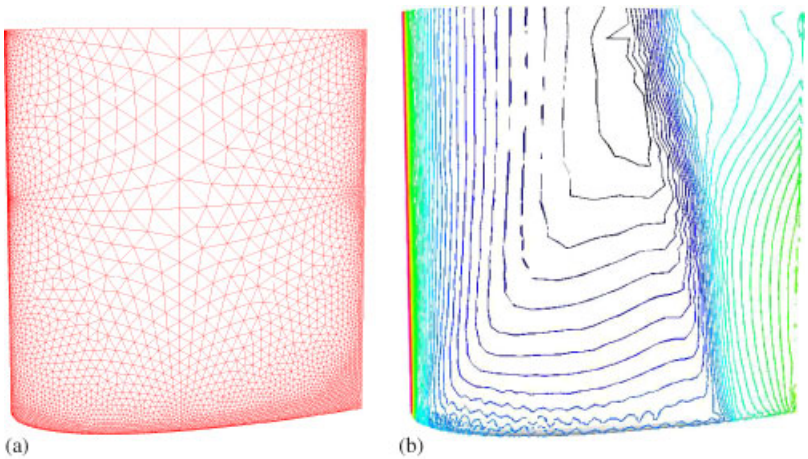

Figure 16. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Initial grid (total nodes: 356 420): (a) surface grid on upper wing; and (b) upper wing surface density contours.

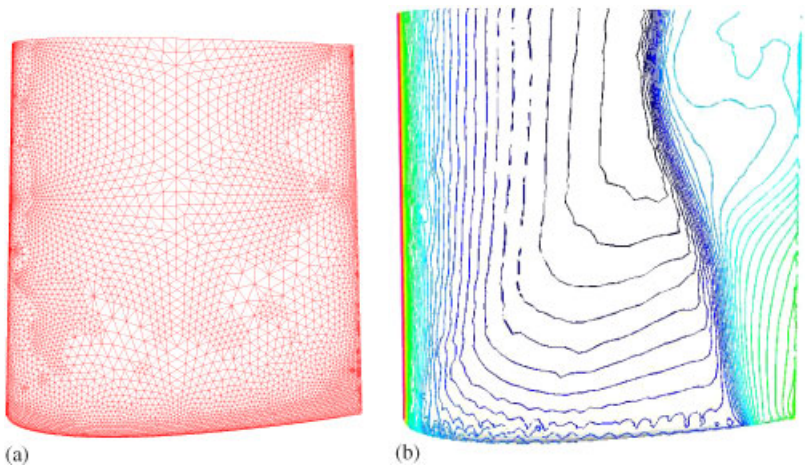

Figure 17. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Adjoint-adapted grid (total nodes: 478 952): (a) surface grid on upper wing after one adjoint-C_D adaptation; and (b) upper wing surface density contours.

the adapted grid C_D has surpassed the estimate of the fine-mesh employed to form the adaptive indicators. The feature-based approach has an opposite (decreasing) trend in C_D than all the other methods (uniform refinement, non-uniform refinement, and adjoint-based adaptation). The surface grids on the upper wing surface for the initial, adjoint-adapted and feature-adapted grids are shown in Figures 16(a), 17(a) and 18(a). The initial, adjoint-adapted and feature-adapted symmetry plane grids are shown in Figures 19(a), 20(a) and 21(a). For the adjoint-adapted grid shown in

DOI: 10.1002/fld

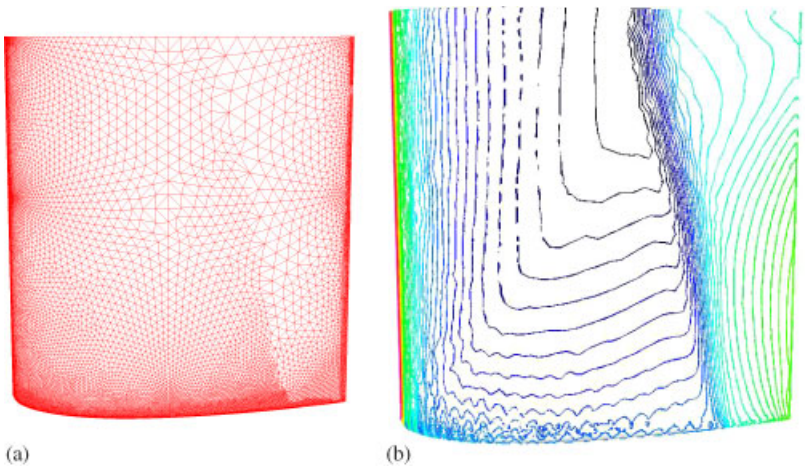

Figure 18. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Feature-adapted grid (total nodes: 1 135 637): (a) surface grid on upper wing after one feature adaptation; and (b) upper wing surface density contours.

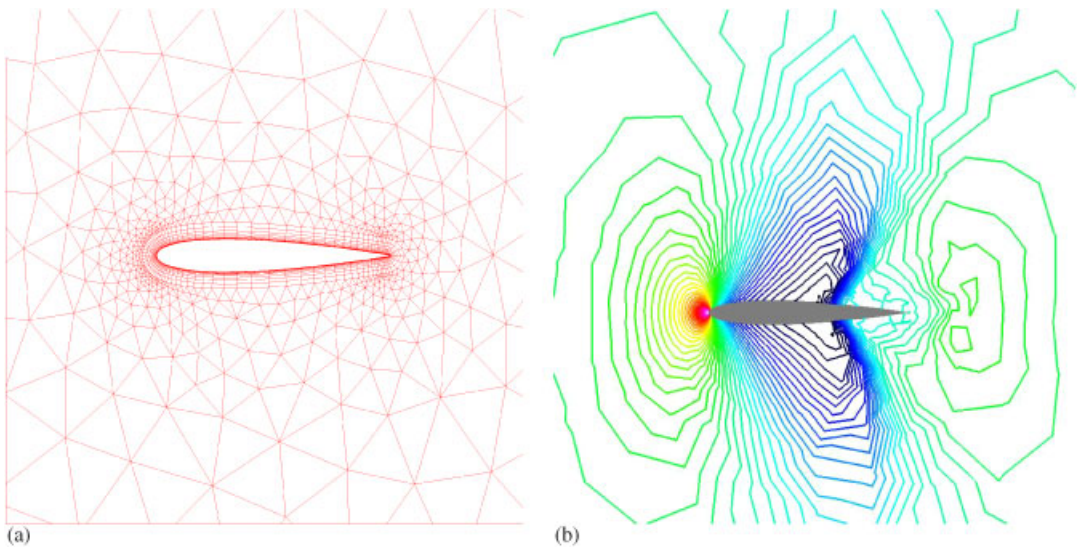

Figure 19. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Initial grid (total nodes: 356 420): (a) symmetry plane grid; and (b) pressure contours on the symmetry plane.

Figures 17(a) and 20, besides refinement near the leading edge, trailing edge and surface of the wing, there is moderate refinement in the wake regions, regions upstream of leading edge (outside boundary layer) and regions near the outer edge of boundary layer. The feature-adapted grid in Figures 18(a) and 21 has considerable refinement near the leading and trailing edges of the

DOI: 10.1002/fld

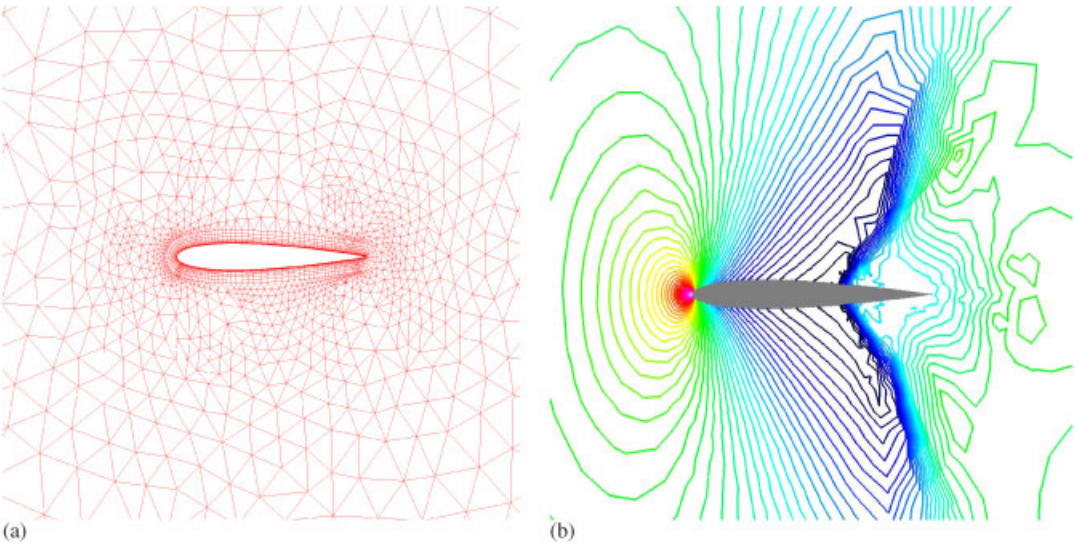

Figure 20. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Adjoint-adapted grid (total nodes: 478 952): (a) symmetry plane grid after one adjoint-C_D adaptation; and (b) pressure contours on the symmetry plane.

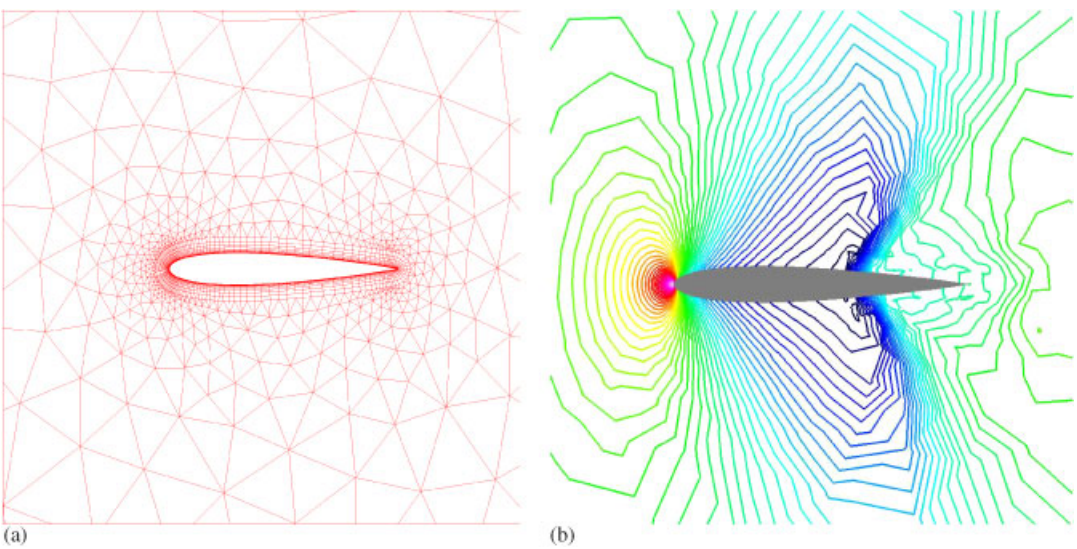

Figure 21. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Feature-adapted grid (total nodes: 1 135 637): (a) symmetry plane grid after one feature adaptation; and (b) pressure contours on the symmetry plane.

DOI: 10.1002/fld

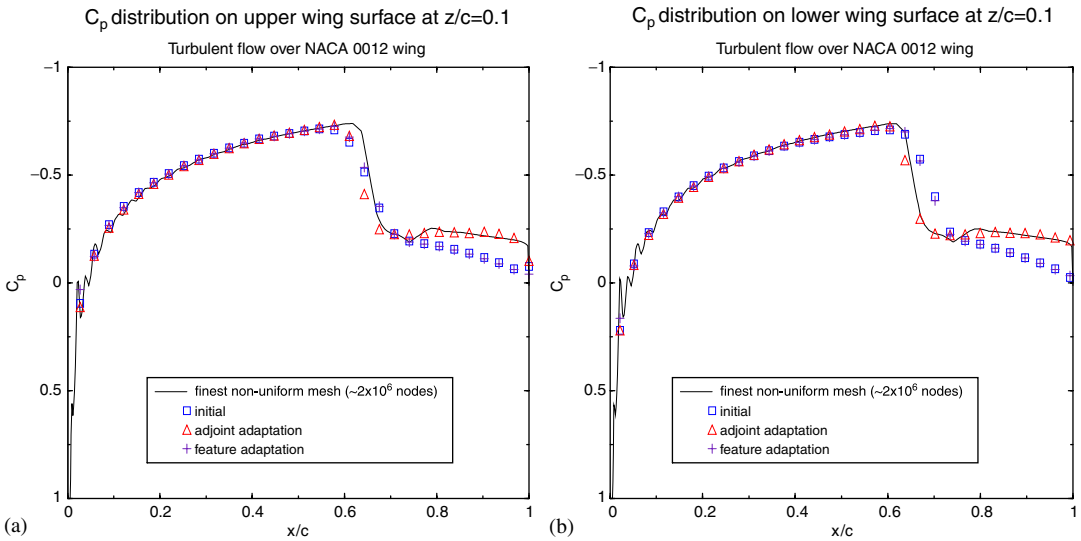

Figure 22. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. C_p distribution on the wing at span wise location $z/c = 0.1$: (a) upper wing; and (b) lower wing.

wing and has moderate surface refinement. There is no significant refinement in rest of the regions.

The initial density contours on the upper surface of the wing is shown in Figure 16(b) and it can be noticed that the shock is smeared. From the density contours of the adjoint-adapted grid in Figure 17(b), it can be observed that the curvature of the shock is captured well and there is a crisper shock compared to the initial grid. However, in the density contours of the feature-adapted grid shown in Figure 18(b), the shock curvature is less captured, and there is no significant reduction in the smearing of shock compared to the initial grid. The pressure contours on the symmetry plane of the initial grid is shown in Figure 19(b). The shocks are poorly resolved outside the boundary layer and in the inviscid regions. In the pressure contours of the adjoint-adapted grid shown in Figure 20(b), there is less smearing of the shocks and their resolution in the inviscid regions have improved greatly. There is no improvement in the pressure contours of the feature-adapted grid shown in Figure 21(b). The feature adaptation has been handicapped by the poor resolution on the initial grid. However, this does not pose a problem for the adjoint adaptation. A better resolved initial grid may be needed for the feature adaptation. This is further affirmed by a look at the C_p plots on the upper and lower wing surfaces at span wise locations $z/c = 0.1$ and 0.9 in Figures 22 and 23.

Figures 24(a) and (b) show the Mach number contours on the symmetry plane for the adjoint-adapted and feature-adapted grids. The Mach number contours are plotted here to see the resolution of the separation zone in the boundary layer. From Figures 24(a) and (b), the thickening of the boundary layer because of the shocks can be noticed. Figures 25(a) and (b) show the Mach number contours near the trailing edge for the adjoint-adapted and feature-adapted grids. It can be observed that the separation zone is better resolved by the adjoint adaptation, while, it is poorly resolved in the feature approach. The feature approach predicts a large separation with the flow separating

DOI: 10.1002/fld

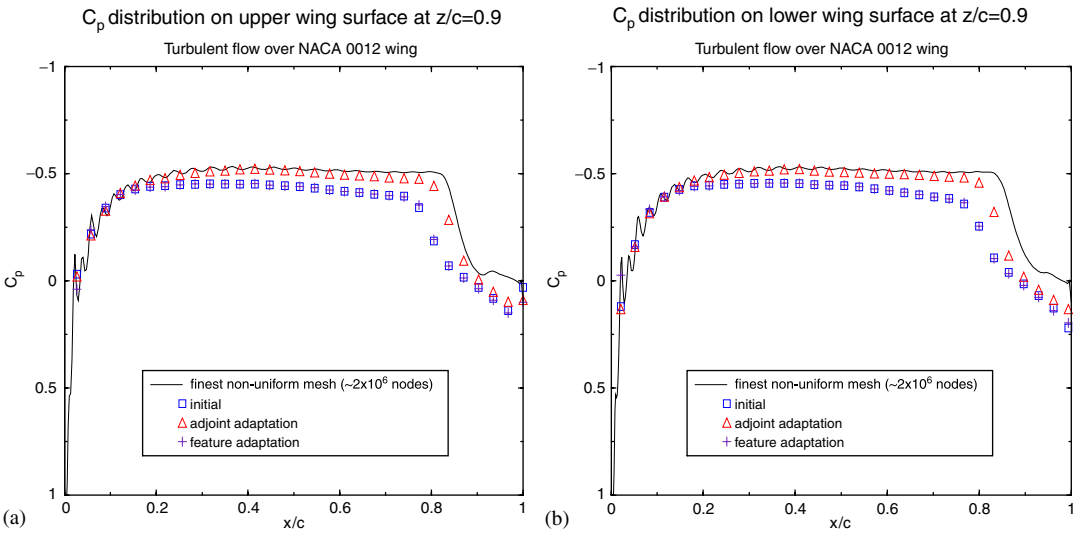

Figure 23. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. C_p distribution on the wing at span wise location $z/c = 0.9$: (a) upper wing; and (b) lower wing.

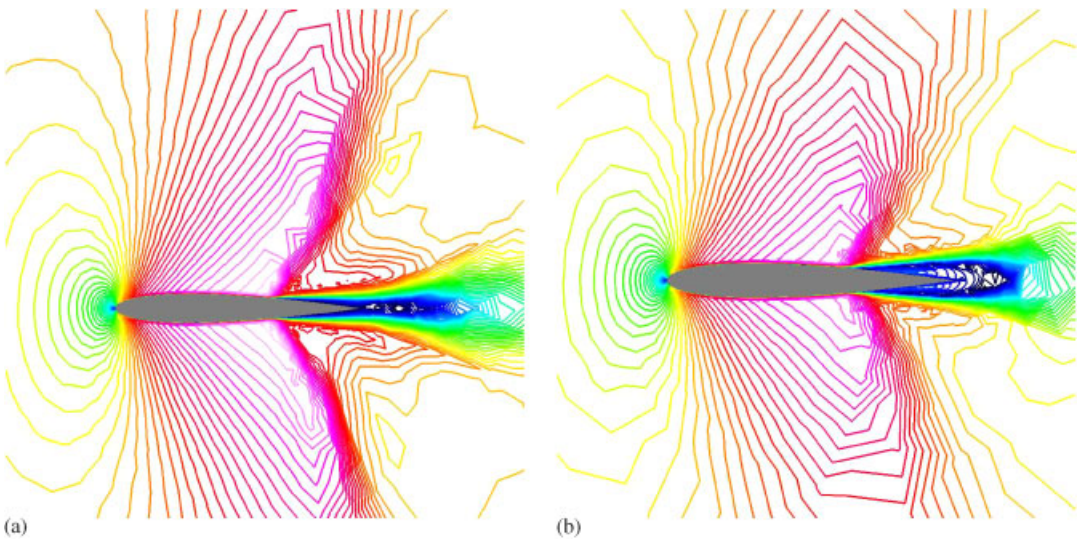

Figure 24. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Mach number contours on the symmetry plane: (a) adjoint-adapted grid; and (b) feature-adapted grid.

immediately after the shock. But, the actual separation occurs further downstream close to the trailing edge as predicted by the adjoint adaptation. The loss of accuracy in the feature-adapted grid may be attributed to the poor resolution of the separation zone.

DOI: 10.1002/fld

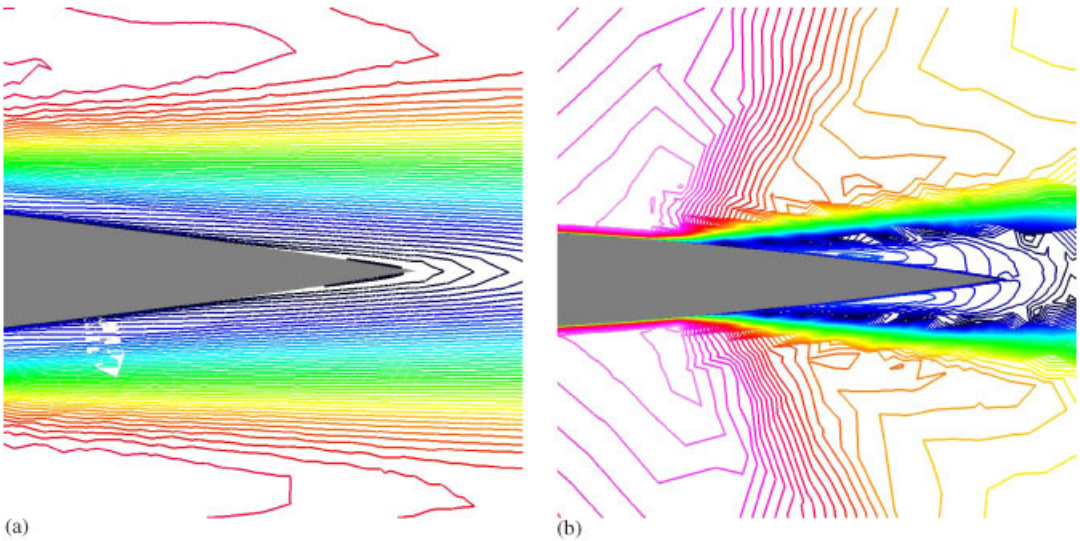

Figure 25. Turbulent flow over NACA 0012 rectangular wing: $M_\infty = 0.95$, $AOA = 0°$, $Re = 3\,000\,000$. Mach number contours near the trailing edge on the symmetry plane: (a) adjoint-adapted grid; and (b) feature-adapted grid.

7. CONCLUSIONS

Adjoint-based and feature-based adaptive strategies have been implemented to improve the accuracy of the chosen output to a prescribed tolerance. Grid adaptation results presented for inviscid, laminar and turbulent flows demonstrate the robustness of the adjoint-based approach over the feature-based approach. In all the adjoint adaptation cases presented, the same level of functional accuracy has been accomplished with a much smaller mesh size (typically a factor of 3–5 reduction in mesh size) compared to the uniformly and non-uniformly refined fine-meshes. The feature approach has suffered by a poor resolution of the initial grid and failed to make significant improvements to functional accuracy. However, the resolution of the initial grid did not pose a problem for the adjoint adaptation. A better resolved initial grid is needed by the feature-based approach.

In the present study, adjoint error estimation and grid adaptation are performed in a single processor, and because of memory constraints, complex geometries and large mesh sizes could not be handled. Future work may be to perform error estimation and grid adaptation in parallel along the lines of Reference [60] to handle large real life applications.

APPENDIX A: MOVING LEAST SQUARES (MLS) APPROXIMATION

The MLS procedure was proposed by Lancaster and Salkauskas [43] for performing smooth interpolation of scattered data. The idea is to start with a weighted least squares fit for an arbitrary data point in the domain, and then move this point over the entire parameter domain, where a weighted least squares fit is computed and evaluated for each point individually. Consider the following form of approximation, where $\mathbf{x}$ represent the co-ordinates of the points and u, the data

DOI: 10.1002/fld

at these points. The MLS approximation can be written as

$$u^h(\mathbf{x}) = \sum_{I=1}^{n} \phi_I^k \, u_I \tag{A1}$$

where $u^h(\mathbf{x})$ is the MLS approximation at $\mathbf{x}$; $\Phi^k(\mathbf{x}) = \{\phi_1^k(\mathbf{x}), \phi_2^k(\mathbf{x}), \ldots, \phi_n^k(\mathbf{x})\}$ are the MLS shape functions of order k; and n is the size of MLS support stencil. Equation (A1) can be rewritten as

$$u^h(\mathbf{x}) = \sum_{i=1}^{m} p_i(\mathbf{x}) a_i(\mathbf{x}) = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{a}(\mathbf{x}) \tag{A2}$$

where $p_i(\mathbf{x})$ are the monomial basis functions; m is the number of terms in the polynomial basis; and $a_i(\mathbf{x})$ are the unknown coefficients. The common basis functions in three dimensions are

linear basis

$$\mathbf{p}^{\mathrm{T}}(\mathbf{x}) = (1, x, y, z) \tag{A3}$$

quadratic basis

$$\mathbf{p}^{\mathrm{T}}(\mathbf{x}) = (1, x, y, z, xy, yz, zx, x^2, y^2, z^2) \tag{A4}$$

Now, the local least squares problem can be defined [43, 47] as

$$\begin{aligned} F &= \sum_I w(\mathbf{x} - \mathbf{x}_I)\,(u^h(\mathbf{x}, \mathbf{x}_I) - u(\mathbf{x}_I))^2 \\ &= \sum_I w(\mathbf{x} - \mathbf{x}_I) \left(\sum_i p_i(\mathbf{x}_I)\, a_i(\mathbf{x}) - u_I \right)^2 \end{aligned} \tag{A5}$$

In the above equation, the polynomial basis $\mathbf{p}^{\mathrm{T}}$ is defined at $\mathbf{x}_I$ and the unknown coefficients a_i are at $\mathbf{x}$ obtained from the weighted least squares fit for the local approximation. $w(\mathbf{x} - \mathbf{x}_I)$ are the weight functions with compact support given by

cubic spline [47]

$$w(\bar{s}) = \begin{cases} \frac{2}{3} - 4\bar{s}^2 + 4\bar{s}^3, & \bar{s} \leqslant \frac{1}{2} \\ \frac{4}{3} - 4\bar{s} + 4\bar{s}^2 - \frac{4}{3}\bar{s}^3, & \frac{1}{2} < \bar{s} \leqslant 1 \\ 0, & \bar{s} > 1 \end{cases} \tag{A6}$$

where $s = \|\mathbf{x} - \mathbf{x}_I\|$; $\bar{s} = s/\rho$; and $\rho = s_{\max}$ is the radius of the circular support [47, 57]. The above cubic spline weights have been constructed to possess C^2 continuity. For rectangular supports [47, 57], tensor product weights (also referred as anisotropic weights) can be defined as

$$w(\mathbf{x} - \mathbf{x}_I) = w\left(\frac{\|x - x_I\|}{\rho_1}\right) w\left(\frac{\|y - y_I\|}{\rho_2}\right) w\left(\frac{\|z - z_I\|}{\rho_3}\right) \tag{A7}$$

where $\rho_1 = \|x - x_I\|_{\max}$, $\rho_2 = \|y - y_I\|_{\max}$ and $\rho_3 = \|z - z_I\|_{\max}$ are the lengths of the rectangular support. The unknown coefficients a_i and the MLS shape functions ϕ_i^ks are obtained by solving Equation (A5) as a least squares minimization problem:

$$\text{find } \mathbf{a} = \{a_0, a_1, \ldots, a_n\} \text{ such that } \mathrm{F}(\mathbf{a}^*) = \min \mathrm{F}(\mathbf{a})$$

DOI: 10.1002/fld

The MLS problem defined by Equation (A5) typically results in an overdetermined system with more equations than unknowns. Also, the MLS system may be severely ill-conditioned on highly stretched meshes. To avoid numerical instability and have a stable algorithm, the least squares problem is solved using singular value decomposition (SVD) [61].

APPENDIX B: NOMENCLATURE

Error estimation variables

P_H^h	prolongation operator
Q_H	flow solution vector at coarse-mesh
Q_h	flow solution vector at fine-mesh
Q_H^h	prolongated flow solution vector at fine-mesh
γ_H	adjoint solution vector at coarse-mesh
γ_h	adjoint solution vector at fine-mesh
γ_H^h	prolongated adjoint solution vector at fine-mesh
Φ^k	moving least squares shape functions of order k
A_p^1	adjoint-based adaptation parameter
A_p^2	feature-based adaptation parameter

Error estimation definitions

uniformly refined mesh	fine-mesh obtained using h-refinement
non-uniformly refined mesh	fine-mesh obtained from a grid generator by manually setting the point spacing
linear–linear basis	linear basis for both flow and adjoint solutions
quadratic–quadratic basis	quadratic basis for both flow and adjoint solutions
linear–quadratic basis	linear basis for flow solution and quadratic basis for adjoint solution
quadratic–linear basis	quadratic basis for flow solution and linear basis for adjoint solution
isotropic	radial weights defined by circular supports
anisotropic	tensor product weights defined by rectangular supports
mixed	both radial and tensor product weights defined by circular-rectangular supports

REFERENCES

1. Thompson JF, Soni BK, Weatherill NP. *Handbook of Grid Generation*. CRC Press: Boca Raton, FL, 1999.
2. Barth TJ. Aspects of unstructured grids and finite-volume solvers for the Euler and Navier–Stokes equations. *VKI Lecture Series 1994-05*, von Karman Institute for Fluid Dynamics, 1994.
3. Mavriplis DJ. Unstructured mesh generation and adaptivity. *Technical Report*, ICASE, Institute for Computer Applications in Science and Engineering, MS 132C, NASA Langley Research Center, Hampton, VA 23681-0001, December 1997. *Report No. 95-26*.
4. Mavriplis DJ. Unstructured grid techniques. *Annual Review of Fluid Mechanics* 1997; 473–514.

DOI: 10.1002/fld

5. Weatherill NP, Hassan O, Marcum DL, Marchant MJ. Grid generation by the Delaunay triangulation. *26th Computational Fluid Dynamics Lecture Series*, von Karman Institute for Fluid Dynamics, January 1994.
6. Baker TJ. Mesh adaptation strategies for problems in fluid dynamics. *Finite Elements in Analysis and Design*, 1997; **25**:243–273.
7. Weatherill NP, Hassan O, Marchant MJ, Marcum DL. Adaptive inviscid flow solutions for aerospace geometries on efficiently generated unstructured tetrahedral meshes. *AIAA 1993-3390*, 1993.
8. Marchant MJ, Weatherill NP. Adaptive techniques for compressible inviscid flows. *Journal of Numerical Methods in Applied Mechanics and Engineering* 1993; **106**:83–106.
9. Weatherill NP, Marchant MJ. Grid adaptation using a distribution of sources applied to inviscid compressible flow simulations. *International Journal for Numerical Methods in Fluids* 1994; **19**:739–764.
10. Marcum DL, Weatherill NP. A procedure for efficient generation of solution adapted unstructured grids. *Computer Methods in Applied Mechanics and Engineering* 1995; **127**:259–268.
11. Marcum DL. Adaptive unstructured grid generation for viscous flow applications. *AIAA Journal* 1996; **34**:2440–2453.
12. Marcum DL, Kelly PG. Solution adaptive unstructured grid generation using pseudo-pattern recognition techniques. *AIAA 1997-1860*, 1997.
13. Pizadeh SZ. An adaptive unstructured grid method by grid subdivision, local remeshing and grid movement. *AIAA 1999-3255*, 1999.
14. Giles MB. On adjoint equations for error analysis and optimal grid adaptation in cfd. *Technical Report*, Oxford University Computing Laboratory, Numerical Analysis Group, Oxford, England, 1997. *Technical Report 97/11*.
15. Giles MB, Pierce NA. Improved lift and drag estimates using adjoint Euler equations. *AIAA 1999-3293*, 1999.
16. Giles MB, Pierce NA. Adjoint recovery of superconvergent functionals from pde approximations. *SIAM Review* 2000; **42**(2):247–264.
17. Giles MB, Pierce NA. Adjoint error correction for integral outputs. *Technical Report*, Oxford University Computing Laboratory, Numerical Analysis Group, Oxford, England, 2001. *Technical Report 01/18*.
18. Giles MB. Defect and adjoint error correction. In *Computational Fluid Dynamics 2000*, Satofuka N (ed.). Springer: Berlin, 2001.
19. Giles MB, Pierce NA. Adjoint error correction for functional outputs. In *Lecture Notes in Computational Science and Engineering*: *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, Barth TJ, Deconinck H (eds). Springer: Berlin, 2000.
20. Giles MB, Pierce NA, Süli E. Progress in adjoint error correction for integral functionals. *Computing and Visualisation in Science* 2004; **6**:113–121.
21. Venditti DA, Darmofal DL. A multilevel error estimation and grid adaptive strategy for improving the accuracy of integral outputs. *AIAA 1999-3292*, 1999.
22. Venditti DA, Darmofal DL. Adjoint error estimation and grid adaptation for functional outputs: application to quasi-one-dimensional flow. *Journal of Computational Physics* 2000; **164**:204–227.
23. Venditti DA. Grid adaptation for functional outputs of compressible flow simulations. *Ph.D. Thesis*, Massachusetts Institute of Technology, June 2002.
24. Babuška I, Rheinboldt WC. Error estimates for adaptive finite element computations. *SIAM Journal on Numerical Analysis* 1978; **15**(4):736–754.
25. Babuška I, Miller A. The post-processing approach in the finite element method—Part 1: calculation of displacements, stresses and other higher derivatives of the displacements. *International Journal for Numerical Methods in Engineering* 1984; **20**:1085–1109.
26. Becker R, Rannacher R. Weighted a posteriori error control in finite element methods. *Technical Report*, Universitat Heidelberg, 1996. Preprint No. 1996-1. http://gaia.iwr.uni-heidelberg.de
27. Becker R, Rannacher R. A feed-back approach to error control in finite element methods: basic analysis and examples. *Technical Report*, Universitat Heidelberg, 1996. Preprint No. 1996-8. http://gaia.iwr.uni-heidelberg.de
28. Becker R, Rannacher R. An optimal control approach to a posteriori error estimation in finite element methods. *Technical Report*, Universitat Heidelberg, 2001. Preprint No. 2001-3. http://gaia.iwr.uni-heidelberg.de
29. Süli E. A posteriori error analysis and global error control for adaptive finite element approximations of hyperbolic problems. *Proceedings of the 16th Biennial Conference in Numerical Analysis*, Griffiths D (ed.), Dundee, 1995.
30. Süli E, Houston P. Adaptive finite element approximations of hyperbolic problems. In *Lecture Notes in Computational Science and Engineering*: *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, Barth TJ, Deconinck H (eds). Springer: Berlin, 2000.

DOI: 10.1002/fld

31. Peraire J, Patera AT. Bounds for linear-functional outputs of coercive partial differential equations: local indicators and adaptive refinement. In *Proceedings of the Workshop On New Advances in Adaptive Computational Methods in Mechanics*, Ladeveze P, Oden JT (eds). Elsevier: Amsterdam, 1997.
32. Paraschivoiu M, Peraire J, Patera AT. A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations. *Computational Methods in Applied Mechanics and Engineering* 1997; **150**:289–312.
33. Venditti DA, Darmofal DL. Grid adaptation for functional outputs: application to two-dimensional inviscid flows. *Journal of Computational Physics* 2002; **176**:40–69. Also, *AIAA 2000-2244*.
34. Venditti DA, Darmofal DL. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. *Journal of Computational Physics* 2003; **187**:22–46.
35. Müller J-D, Giles MB. Solution adaptive mesh refinement using adjoint error analysis. 2001. *AIAA 2001-2550*.
36. Balasubramanian R. Error estimation and grid adaptation for functional outputs using discrete adjoint sensitivity analysis. *Master's Thesis*, Mississippi State University, December 2002.
37. Park MA. Adjoint-based, three-dimensional error prediction and grid adaptation. *AIAA Journal* 2004; **42**(9):1854–1862. Also, *AIAA 2002-3286*.
38. Park MA. Three-dimensional turbulent RANS adjoint-based error correction. 2003. *AIAA 2003-3849*.
39. Briley WR, Taylor LK, Whitfield DL. High-resolution viscous flow simulations at arbitrary Mach number. *Journal of Computational Physics* 2003; **184**:79–105.
40. Spalart PR, Allmaras SR. A one-equation turbulence model for aerodynamic flows. 1992. *AIAA 1992-0439*.
41. Spalart PR, Shur M. On the sensitization of turbulence models to rotation and curvature. *Aerospace Science and Technology* 1997; **1**(5):297–302.
42. Balasubramanian R, Newman III JC. Discrete direct and adjoint sensitivity analysis for variable Mach number flows. *International Journal for Numerical Methods in Engineering* 2006; **66**(2):297–318.
43. Lancaster P, Salkauskas K. Surfaces generated by moving least squares methods. *Mathematics of Computation* 1981; **37**(155):141–158.
44. Franke R. Scattered data interpolation: tests of some methods. *Mathematics of Computation* 1982; **38**(157):181–200.
45. Farwig R. Rate of convergence of Shepard's global interpolation formula. *Mathematics of Computation* 1986; **46**(174):577–590.
46. Farwig R. Multivariate interpolation of arbitrarily spaced data by moving least squares methods. *Journal of Computational and Applied Mathematics* 1986; **16**:79–93.
47. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P. Meshless methods: an overview and recent developments. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:3–47 (Special Issue on Meshless Methods).
48. Levin D. The approximation power of moving least squares. *Mathematics of Computation* 1998; **67**(224):1517–1531.
49. Marcum DL. *AFLR3*: *Advancing Front/Local Reconnection Grid Generator*. Mississippi State University. http://www.erc.msstate.edu/simcenter/docs/solidmesh/
50. Beall MW, Shephard MS. A general topology-based mesh data structure. *International Journal for Numerical Methods in Engineering* 1997; **40**(9):1573–1596.
51. Beall MW. An object-oriented framework for the reliable automated solution of problems in mathematical physics. *Ph.D. Thesis*, Rensselaer Polytechnic Institute, May 1999.
52. Remacle J-F, Karamete BK, Shephard MS. Algorithm oriented mesh database. *Technical Report*, SCOREC, Rensselaer Polytechnic Institute, 2000. *Report 2000-5*. http://www.scorec.rpi.edu/cgi-bin/reports/GetByYear.pl?Year = 2000
53. Mavriplis DJ. Adaptive meshing techniques for viscous flow calculations on mixed element unstructured meshes. *International Journal for Numerical Methods in Fluids* 2000; **34**(2):93–111. Also, *NASA CR-201675*, *ICASE Report No. 97-20*.
54. Sheng C, Newman III JC, Remotigue M, Chen JP, Marcum DL, Whitfield DL. Development of unstructured computational capabilities to MSU-TURBO with an arbitrary Mach algorithm. *Technical Report*, Mississippi State University, 2002. *Technical Report MSSU-COE-ERC-02-16*.
55. Hyams DG. An investigation of parallel implicit solution algorithms for incompressible flows on unstructured topologies. *Ph.D. Thesis*, Mississippi State University, May 2000.
56. Roe P. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics* 1981; **23**:357–372.
57. Balasubramanian R. Adjoint-based error estimation and grid adaptation for functional outputs from CFD simulations. *Ph.D. Thesis*, Mississippi State University, December 2005. http://www.erc.msstate.edu/∼raavi/PhD-Dissertation.html

DOI: 10.1002/fld

58. Nielson GM. The side-vertex method for interpolation in triangles. *Journal of Approximation Theory* 1979; **25**:318–336.
59. Schmitt V, Charpin F. Pressure distributions on the ONERA-M6-Wing at transonic Mach numbers, Experimental Data Base for Computer Program Assessment. *Report of the Fluid Dynamics Panel Working Group 04*, *AGARD AR 138*, May 1979. http://www.grc.nasa.gov/WWW/wind/valid/m6wing/m6wing.html
60. Lee-Rausch EM, Park MA, Jones WT, Hammond DP, Nielsen EJ. Application of parallel adjoint-based error estimation and anisotropic grid adaptation for three-dimensional aerospace configurations. *AIAA 2005-4842*.
61. Press WH, Teukolsky SA, Vellerling WT, Flannery BP. *Numerical Recipes in C*: *The Art of Scientific Computing*. Cambridge University Press: Cambridge, MA, 1992.

DOI: 10.1002/fld