AIAA JOURNAL
Vol. 48, No. 9, September 2010

Validation of an Output-Adaptive, Tetrahedral Cut-Cell Method for Sonic Boom Prediction

Michael A. Park*
NASA Langley Research Center, Hampton, Virgnia 23681
and
David L. Darmofal†
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

DOI: 10.2514/1.J050111

A cut-cell approach to computational fluid dynamics that uses the median dual of a tetrahedral background grid is described. The discrete adjoint is also calculated for an adaptive method to control error in a specified output. The adaptive method is applied to sonic boom prediction by specifying an integral of offbody pressure signature as the output. These predicted signatures are compared to wind-tunnel measurements to validate the method for sonic boom prediction. Accurate midfield sonic boom pressure signatures are calculated with the Euler equations without the use of hybrid grid or signature propagation methods. Highly refined, shock-aligned anisotropic grids are produced by this method from coarse isotropic grids created without prior knowledge of shock locations. A heuristic reconstruction limiter provides stable flow and adjoint solution schemes while producing similar signatures to Barth–Jespersen and Venkatakrishnan limiters. The use of cut cells with an output-based adaptive scheme automates the volume grid generation task after a triangular mesh is generated for the cut surface.

Nomenclature

A	=	area
a	=	speed of sound
C_l	=	coefficient of lift
C_p	=	coefficient of pressure
E	=	total energy per unit volume
$\mathbf{F}$	=	flux
f	=	output function
$\mathcal{H}$	=	van Leer approximate Riemann solver
h	=	isotropic element size
I	=	estimated error
l	=	body length
M	=	Mach number
N	=	number of control volumes
$\mathbf{n}$	=	outward-pointing normal
p	=	pressure
$\mathcal{Q}$	=	conserved state
$\mathbf{q}$	=	primitive state
$\mathbf{R}$	=	residual
r	=	radius
s	=	integration surface
t	=	time
tol	=	error tolerance
u, v, w	=	components of velocity
V	=	control volume
Γ	=	control volume boundary
γ	=	specific heat ratio
ϵ	=	estimated error ratio
κ	=	element
λ	=	adjoint state
ρ	=	density
Φ, ϕ	=	limiting function
ψ	=	reconstruction weights
Ω	=	domain
ω	=	spatial error convergence

Presented as Paper 2008-6594 at the 26th AIAA Applied Aerodynamics Conference, Honolulu, HI, 18–21 August 2008; received 31 July 2009; revision received 2 November 2009; accepted for publication 10 November 2009.

*Research Scientist, Computational Aerosciences Branch, Mail Stop 128. Senior Member AIAA.

†Professor, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue, Room 33-207. Senior Member AIAA.

I. Introduction

THE acceptance of an aircraft's sonic boom to the general population is a requirement for supersonic flights over land and therefore the commercial viability of a supersonic transport. Predicting how sonic boom signatures are perceived is a challenging task that requires the prediction of the signature on the ground. This is a task complicated by the long propagation distances, atmosphere variations, and the Earth's turbulent boundary layer. A detailed review of the history and state of the art of sonic boom modeling is provided by Plotkin [1].

The propagation of a sonic boom is often separated into three logical stages or regions, depicted in Fig. 1, to facilitate analysis [2]. The near field is a region near the aircraft, where shocks are formed and strongly influenced by nonlinear phenomena such as shock–shock interaction, shock curvature, and crossflow. Higher-pressure portions of the signature travel faster than lower pressure portions of the signature because of variations in the local speed of sound. This slight speed difference causes the shocks to deform by elongating and coalescing in the midfield. The signature is also refracted by variations in the atmospheric speed of sound. In the far field, the signature will typically form an N-wave. The boundaries of these regions are case-specific.

The propagation of the relatively weak pressure signatures of a sonic boom beyond the near field is difficult using common discretization and gridding techniques. This problem is more acute for unstructured-grid methods that are often employed to capture the geometrical complexity of the model, especially if the grids are not aligned with the shocks. To improve alignment, isotropic unstructured grids are stretched to align the tetrahedra with the freestream Mach angle to improve signal propagation for initial grids [3]. This alignment issue has also given rise to hybrid methods [4–6], in which near-body unstructured-grid solutions are interpolated to shock-aligned structured-grid methods to increase accuracy.

Adaptive methods have also been applied to sonic boom using adaptation indicators based on Mach number and density distributions [7,8]. Adaptive approaches based on these types of flow

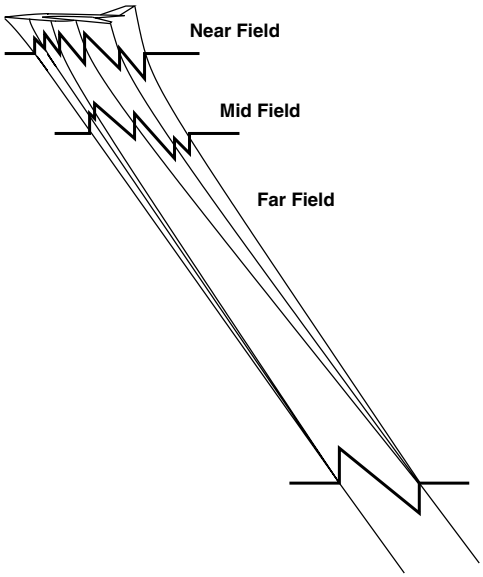


Fig. 1 Sonic boom signature propagation zones.

features and local error estimates [9–11] can be ineffective as they do not account for propagation of errors. An alternative method is to estimate the error in the calculation of a specified engineering output functional [12–15]. Output error indicators use the dual or adjoint solution of an output functional to account for the impact of local error as well as the transport of these local errors throughout the problem domain to improve the calculation of that output functional. This output-adaptive approach has been applied to sonic boom prediction with discontinuous Galerkin [16], Cartesian finite volume [17], and body-fitted unstructured-grid finite volume [18,19]. In this work, anisotropic output-based adaptation is combined with a tetrahedral cut-cell discretization and applied to 3-D sonic boom prediction. The combination provides an automated, robust method for the prediction of nontrivial 3-D sonic boom problems.

Cut-cell methods with Cartesian background grids [20–23] have been very successful for Euler simulations including output-based adaptation [24]. The regular structure of the Cartesian background grid permits extremely efficient solution schemes. Cartesian background grids have the capability to only provide anisotropic resolution in the Cartesian directions [23], but can be effective if a Cartesian direction is aligned with the bow shock [17]. Simplex meshes have the ability to stretch the triangular and tetrahedral elements in arbitrary directions. This permits the efficient representation of anisotropic features (i.e., shocks). The cut-cell method is also applicable to simplex meshes [25–28]. When the constraint of providing a body-fitted grid is removed, the grid adaptation task becomes much simpler. The complexities of adaptation on curved domain boundaries [29] is eliminated and robustness is dramatically increased. This increase in robustness may enable automated, efficient, high Reynolds number simulations [26,27].

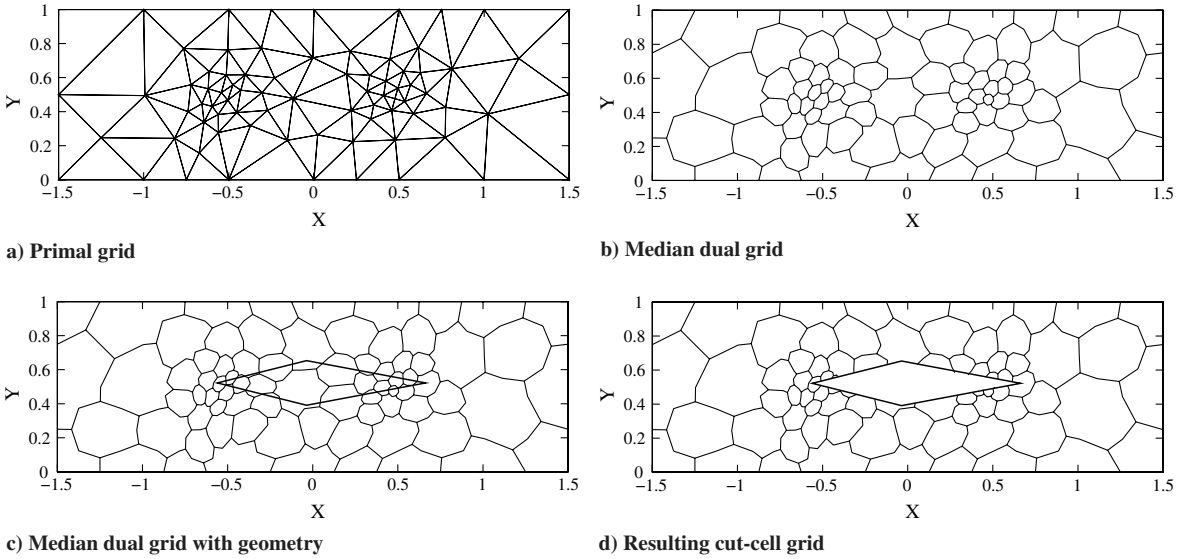
The 3-D simplex cut-cell method is introduced with particular attention to the robust determination of the intersection of the geometry with the underlying tetrahedral mesh in Sec. II. The flow and adjoint solvers are described and a heuristic limiter is introduced that significantly improves convergence in Sec. III. The output-based adaptive method is then reviewed in Sec. IV. Finally, the output-based simplex cut-cell method is validated for sonic boom applications by comparison to wind-tunnel data for representative configurations in Sec. V.

II. Cut-Cell Determination

To introduce the 3-D cut-cell method a simple 2-D example is presented. The primal triangular grid is shown in Fig. 2a. The control volumes used by the flow solver are the median duals of this grid, shown in Fig. 2b. These median duals are constructed by gathering the three dual faces that are inside each primal triangle, which each connect the triangle center to one of the triangle side midpoints. The geometry is a diamond airfoil, shown with the uncut median-dual background grid in Fig. 2c. The airfoil geometry is Boolean-subtracted from this background grid, removing the portion of the background grid that is external to the flow domain (Fig. 2d). The resulting cut and uncut duals are the control volumes of the finite volume method.

In the 3-D case, the domain of the simulation is constructed by Boolean subtraction of a manifold triangular boundary representation from a background grid. This triangulation can come from many sources. Two examples are CAD geometry [30–32] and component-based geometry [22]. Figure 3a is a triangular surface grid of a cylinder constructed on a CAD solid [32].

The background grid contains closed simplicial polytope control volumes. In 3-D, these polyhedra are the median duals of a tetrahedral grid. The 3-D median dual about a single primal node is shown in Fig. 3b. Just as in the 2-D case, this dual control volume may not be convex. Each dual polyhedra of a tetrahedral grid contains $\mathcal{O}(100)$ triangles. Figure 4 illustrates the two dual triangles associated with an edge of a primal tetrahedron. There are six edges in a tetrahedron, which contains a total of 12 triangular dual faces shared by the duals at each of its four nodes. For robustness and a decrease in execution time and memory usage, a triangular dual face



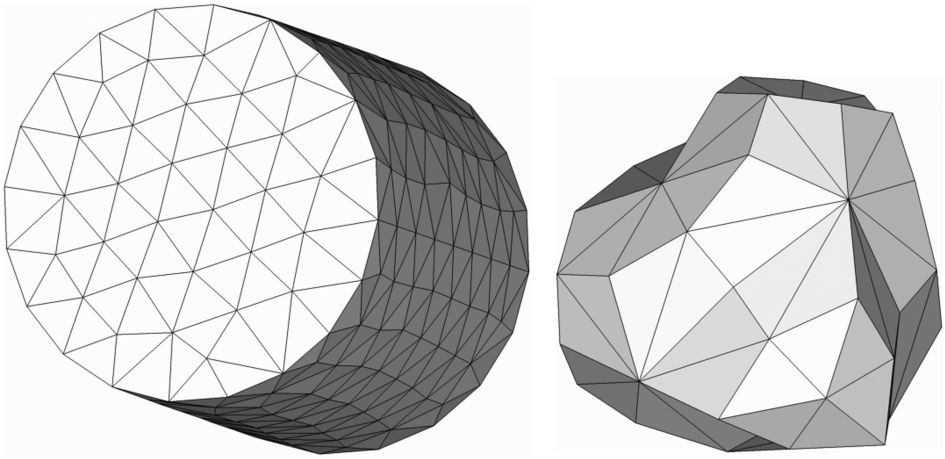
a) Primal grid

b) Median dual grid

c) Median dual grid with geometry

d) Resulting cut-cell grid

Fig. 2 Cut-cell illustration of an diamond airfoil in 2-D.



a) Cylindrical cutting surface **b) Median dual**

Fig. 3 Cylindrical cutting surface and median dual.

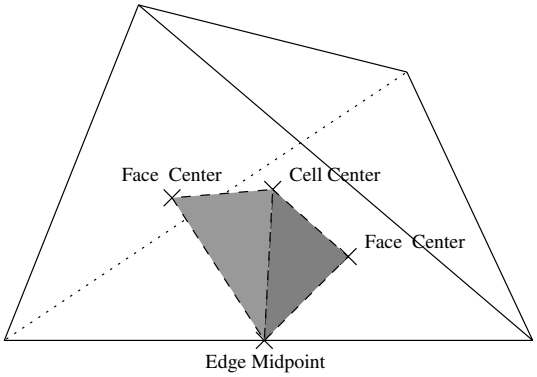


Fig. 4 Two dual triangles associated with a single tetrahedral edge.

is only represented once in the intersection procedure and shared by the two adjacent control volumes.

The major steps of the Boolean-subtraction procedure are as follows:

1) The set of background grid duals that intersect the cutting surface are gathered with conservative approximate intersection tests.

2) All pierce points and cuts of the dual and cutting surface triangle–triangle intersections are calculated (Fig. 5).

3) The pierce points are inserted and the cuts are recovered by Delaunay subtriangulating the intersected triangles.

4) The inside/outside status of all subtriangles adjacent to cuts is determined.

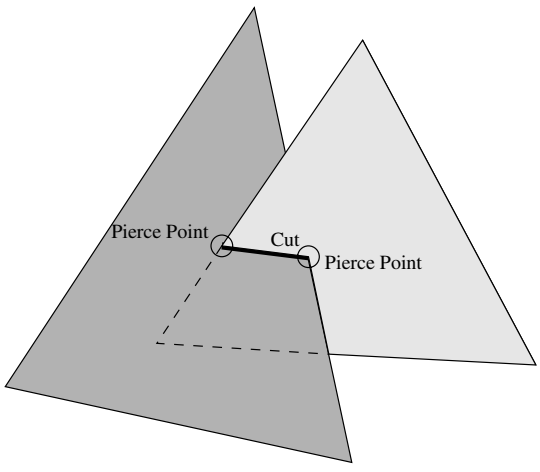


Fig. 5 Cut and two pierce points resulting from a triangle pair.

5) The inside/outside status of subtriangles is relaxed to neighboring subtriangles and triangles to segregate distinct regions and remove the portion of the dual outside of the computational domain. These steps are described in detail in the remainder of this section.

The Boolean subtraction of two manifold triangular polyhedra (surface grid and each of the background grid control volumes) reduces to a series of triangle–triangle intersections [22]. For computational efficiency, a near tree [33,34] is employed to only perform the intersection test for triangle pairs that have a possibility of intersecting. Only the duals that have a potential of intersecting the cut surface are created for the intersection test. This potential is determined by an approximate, but conservative, intersection of the primal tetrahedra and the cut-surface triangulation. When a tetrahedra is intersected by the cut surface the dual surrounding each of the four nodes is marked for creation. These steps reduce the complexity of the intersection problem that can be on the order of the number of surface triangles times by number of volume triangles for a naive implementation.

The intersection test of two triangles is evaluated with only floating-point arithmetic. To ensure that each intersection test is only performed in a single orientation, a hierarchical data structure is employed [28]. The triangle–triangle intersection determination decomposes into triangle-segment intersection determination via volume computations, in which the segments are the three sides of the triangle. The intersection produces two segment-triangle pierce points and a cut connecting the two pierce points (see Fig. 5). The triangles and their segments have a unique orientation, because they are only represented once in the data structure. When they are compared in this unique configuration, they always return the same intersection determination. This consistency is pivotal to the robustness of the cutting scheme and permits the use of floating-point arithmetic, which is faster and much simpler than using adaptive precision arithmetic. Performing the same intersection test in different orientations would make an inconsistency extremely likely because of differing floating-point round-off errors.

In the rare case when floating-point arithmetic results in an exact degeneracy (zero volume), the cut surface is perturbed slightly (by a factor of machine epsilon) and the entire cutting procedure is restarted to maintain consistency. These exact degeneracies have only been observed on initial grids when the user exactly aligns the background grid and surface triangulation. The actual perturbation can be replaced with a virtual perturbation [22,35], in which a unique tie-break to the degenerate volume computing determinate is applied. This process has not been adopted in the current work, because the degeneracies are detected early in the cutting procedure (within seconds) and a single perturbation vector applied to the entire surface triangulation has been sufficient to eliminate the degeneracies. The virtual perturbation technique may be implemented as a topic for future work if it becomes necessary.

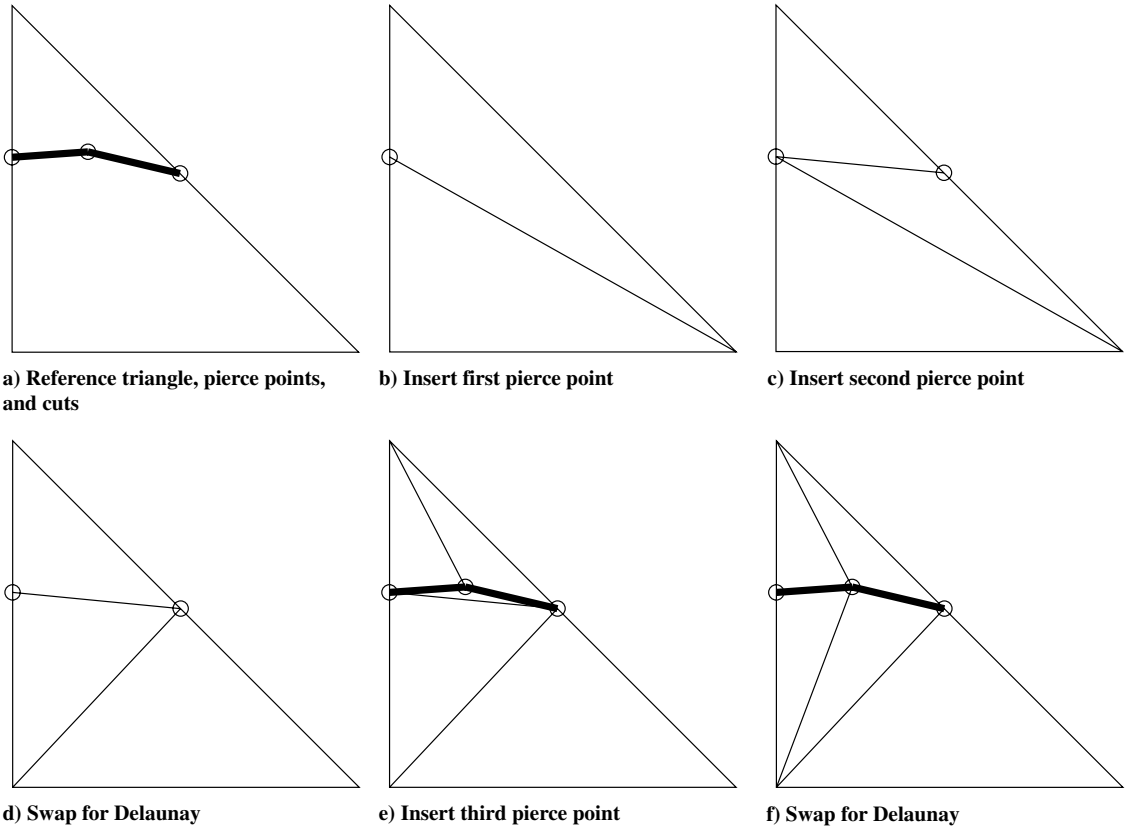
Each of the intersected triangles is constrained Delaunay triangulated into a set of subtriangles to include the pierce points and cuts [22]. Local barycentric coordinates are employed in a reference triangle (Fig. 6). The goal is to produce a Delaunay triangulation in the reference triangle, not the physical triangle. In this example, two cuts (thick lines) and three pierce points (circles) are introduced into this reference triangle. The triangle–triangle pierce points are inserted, one at a time, into the triangles with an iterative method [36]. A point insertion involves splitting the subtriangle that surrounds the new point into two or three (Fig. 6b). The target subtriangle that will be split is selected by examining the subtriangles that result after splitting. This target subtriangle is selected so that the smallest of the resultant subtriangles has the greatest signed area in floating-point arithmetic. This area calculation is always performed in a unique configuration so the resulting triangulation will have a nonnegative area in finite precision arithmetic. The subtriangles are provided to the flow solver in this same orientation preventing the introduction of negative area triangles in the flow solver. Shewchuk [37] describes a Delaunay triangulation scheme that uses exact arithmetic for area calculations, but can provide negative area triangles in floating-point arithmetic. After insertion, the subtriangle sides are swapped to regain a Delaunay grid (Figs. 6d and 6f). Cuts that are not present are recovered [38] producing a constrained Delaunay grid of the reference triangle.

Once the triangles have been subtriangulated to include all of the pierce points and cuts, subtriangles are categorized inside or outside of the domain. Each cut has four adjacent subtriangles (see Fig. 7). The subtriangle pair, S1 and S2, lie on a triangle from the boundary surface triangulation, and the subtriangle pair, V1 and V2, lie on a triangle from the face of a background dual volume. Thus, each subtriangle in a pair are in the same plane since they have the same parent triangle. The normals of the cut-surface triangles (S1 and S2) point into the domain. The signed volume of a tetrahedron formed from the nodes of an S subtriangle and the third node of a V subtriangle are positive if the V subtriangle is inside the domain. The V subtriangle that creates a positive volume tetrahedra is given an inside status. The V subtriangle that creates a negative volume tetrahedra is given an outside status. The inside/outside status of the S subtriangles is determined in the same fashion, each dual knows the orientation of its V subtriangle normals by construction.

The inside/outside determination procedure uses only local subtriangles. It does not use a global search over all triangles, which is required by ray-casting [22]. The localization of the inside/outside determination allows the current implementation to use computer memory cache more efficiently than ray tracing. The inside subtriangle status is propagated to adjacent subtriangle and uncut triangle neighbors with a flood-fill scheme, which propagates status until a cut is reached. The validity of the cut-cell topology is verified during the flood-fill operation. If a subtriangle pair is set to the same status (i.e., V1 and V2 in Fig. 7 are both inside) the flood-fill is terminated and the user is alerted to the location, because this indicates that the cutting surface is not manifold.

An example is provided to illustrate the inside/outside determination of a multiple region cut cell in 2-D. Figure 8a contains a nonconvex background grid control volume and a wing trailing edge (thicker line). The cutting and inside/outside determination is applied and lines adjacent to intersections are given an integer (Fig. 8b). This integer is 0 for lines outside of the domain and positive for lines inside of a domain. The positive integers used to mark inside lines are unique. A relaxation is performed so that adjoining lines not separated by an intersection are both set to the larger of their two integers. This provides inside/outside determination for lines that are not directly intersected and categorizes the lines into distinct regions (Fig. 8c), in which each region has a distinct integer. These distinct regions are each represented as a separate control volume in the flow solver.

The 3-D cut-surface and dual-volume surface grids from Fig. 3 are shown in Fig. 9a. The cut surface intersecting the median dual is shown as a wire frame so that the median dual is visible. Figure 9b shows the result of the Boolean subtraction. The surface resulting from the subtraction contains the inside triangles from both surfaces and the inside subtriangles of the cut triangles.



a) Reference triangle, pierce points, and cuts **b) Insert first pierce point** **c) Insert second pierce point**

d) Swap for Delaunay **e) Insert third pierce point** **f) Swap for Delaunay**

Fig. 6 Subtriangle construction by Delaunay point insertion into reference triangle.

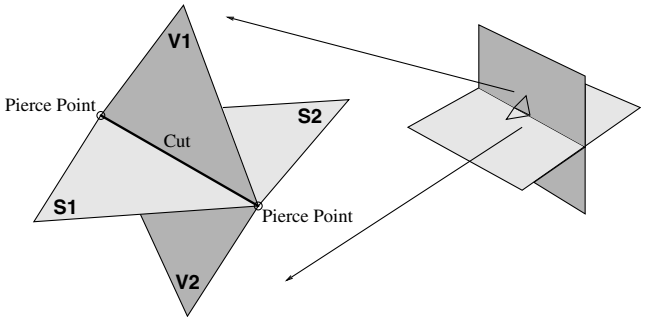


Fig. 7 Inside/outside determination of subtriangles at cuts.

III. Flow and Adjoint Solvers

Fully unstructured Navier–Stokes three-dimensional (FUN3D) is a suite of codes for finite volume computational fluid dynamics (CFD) [39,40]. The FUN3D website[‡] contains the user manual and an extensive list of references. FUN3D is able to solve incompressible, Euler, and Reynolds-averaged Navier–Stokes flow equations, either tightly or loosely coupled to a turbulence model. The Euler equations are used in this study. Domain decomposition is employed to fully exploit the distributed memory of a cluster of computers to increase problem size and reduce the execution time of the simulation process.

A. Governing Equations

The Euler equations are

$$\frac{\partial \mathcal{Q}}{\partial t} + \nabla \cdot \mathbf{F} = 0 \tag{1}$$

$$\mathcal{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix} \tag{2}$$

$$\mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(p+E) \end{bmatrix} \hat{i} + \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ v(p+E) \end{bmatrix} \hat{j} + \begin{bmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ w(p+E) \end{bmatrix} \hat{k} \tag{3}$$

where ρ is density; u, v, and w are velocity; E is total energy per unit volume; and p is pressure. These quantities are related by the ideal-gas relation:

$$p = (\gamma - 1)\left(E - \rho \frac{u^2 + v^2 + w^2}{2}\right) \tag{4}$$

with the specific heat ratio $\gamma = 1.4$ for air.

The divergence theorem is applied over a set of control volumes to produce a finite volume scheme:

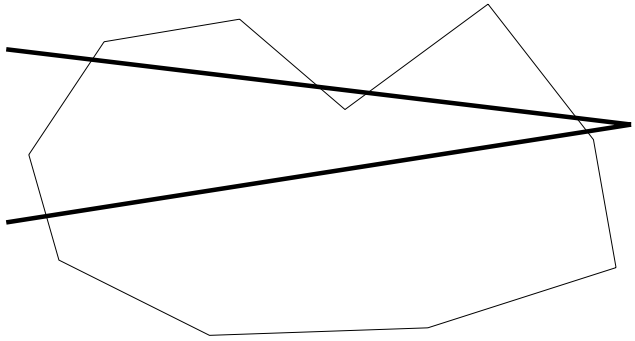
$$\int_{V_i} \left(\frac{\partial \mathcal{Q}}{\partial t} + \nabla \cdot \mathbf{F}\right) \mathrm{d}V = V_i \frac{\mathrm{d}\mathcal{Q}_i}{\mathrm{d}t} + \int_{\Gamma_i} \mathbf{F} \cdot \mathbf{n} \, \mathrm{d}\Gamma = 0 \tag{5}$$

where Γ_i are the boundaries of the control volumes with volume V_i and $\mathbf{n}$ is an outward-pointing normal. The average of $\mathcal{Q}$ in each control volume is $\mathcal{Q}_i$. The flux integration is approximated as

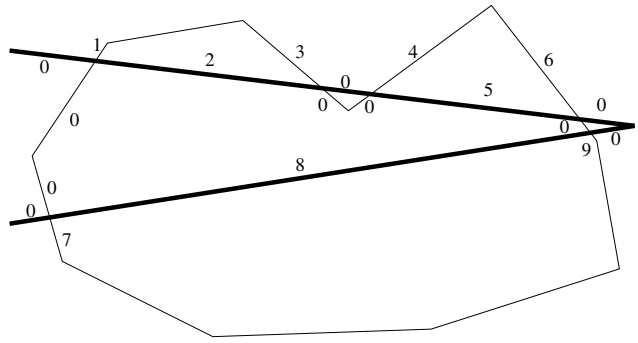
$$\int_{\Gamma_i} \mathbf{F} \cdot \mathbf{n} \, \mathrm{d}\Gamma \approx \sum_{f \in \Gamma_i} \mathcal{H}(\mathbf{q}_{lf}, \mathbf{q}_{rf}, \mathbf{n}_f) A_f = \mathbf{R}_i(\mathbf{Q}) \tag{6}$$

where R_i is the steady-state discrete residual for control volume i, the summation is over the faces of the control volume [39]. The van Leer

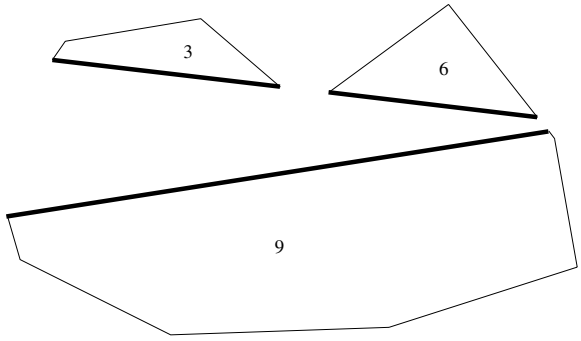
[‡]Data available online at http://fun3d.larc.nasa.gov [retrieved 1 June 2010].



a) Background grid control volume and wing trailing edge



b) Inside/outside determination at intersections



c) Completion of flood-fill resulting in 3 regions

Fig. 8 Inside/outside and multiple region determination in 2-D.

[41] approximate Riemann solver $\mathcal{H}$ is used to compute the flux from the primitive states,

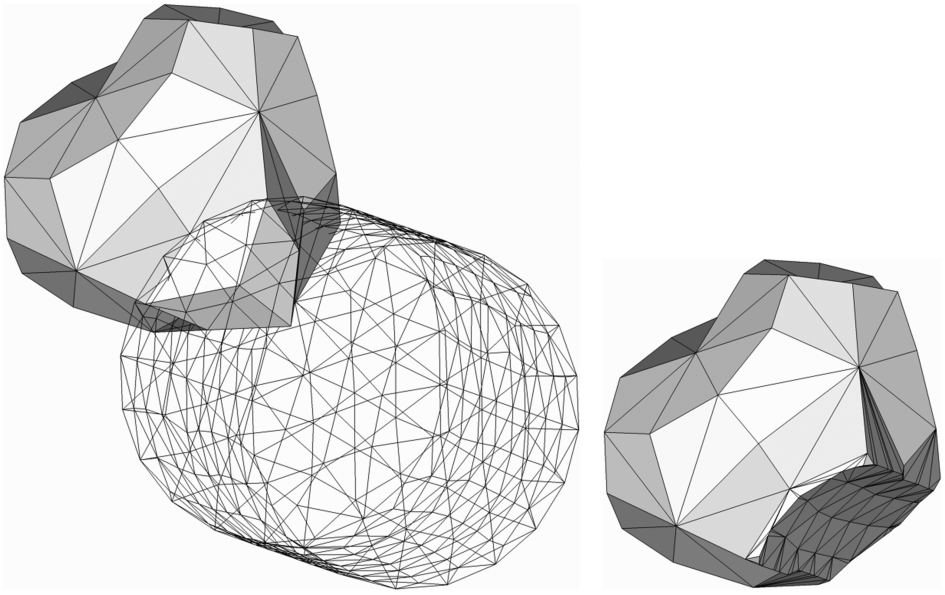
$$\mathbf{q} = \begin{bmatrix} \rho \\ u \\ v \\ w \\ p \end{bmatrix} \tag{7}$$

at the borders of the neighboring control volumes, $\mathbf{q}_{rf}$ and $\mathbf{q}_{lf}$. These face values are reconstructed from cell averages (the reconstruction method is described below). The discrete equations are established simultaneously for each control volume,

$$\mathbf{V} \frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t} + \mathbf{R}(\mathbf{Q}) = 0 \tag{8}$$

which makes the discrete solution vector $\mathbf{Q} \in \mathcal{R}^{5N}$, discrete residual vector $\mathbf{R} \in \mathcal{R}^{5N}$, and $\mathbf{V} = \mathrm{diag}(V_i)$, where N is the number of control volumes. The flux integration scheme (including face state reconstruction from cell averages) is detailed in the following sections.

A backward Euler solution update scheme is employed with a variable pseudo time step [39]. An approximate nearest neighbor



a) Median dual and cutting surface **b) Resulting polyhedra**

Fig. 9 Median dual, cylindrical cutting surface, and resulting polyhedra.

linearization is used to reduce the memory required for the implicit point-iterative method. The solution update is limited to 15% of the current ρ and p to increase robustness during initial transients when starting from freestream.

B. Adjoint Equations

After the flow solution is known, the discrete adjoint equations [42,43] are solved to enable output-based adaptation. Given an output function f, the discrete adjoint equations are

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right)^T \lambda = -\left(\frac{\partial f}{\partial \mathbf{Q}}\right)^T \tag{9}$$

The linear adjoint equations in Eq. (9) are solved with a dual-consistent time-marching method [44,45]. The dual-consistent solution method guarantees that the adjoint equations will have the same eigenvalues and therefore the same asymptotic convergence rate as the iterative method for the flow equations.

C. Inviscid Flux Integration

The existing FUN3D body-fitted approach lumps the median-dual pieces to a single effective area and normal direction for each edge they surround [46]. After lumping, all of the inviscid terms are calculated with a loop over edges, which is computationally efficient. Conserved states $\mathbf{Q}$, used in the time advancement scheme, are converted to primitive states $\mathbf{q}$ for face state reconstruction. The primitive state is extrapolated from the nodes to establish the primitive state at these lumped faces $\mathbf{q}_f$ using the gradients $\nabla \mathbf{q} = [\mathbf{q}_x, \mathbf{q}_y, \mathbf{q}_z]$ reconstructed from the cell-averaged state $\mathbf{q}_0$ (see Sec. III.D), face center $\mathbf{x}_f$, and node $\mathbf{x}_0$,

$$\mathbf{q}_f = \mathbf{q}_0 + \nabla \mathbf{q}(\mathbf{x}_f - \mathbf{x}_0) \tag{10}$$

for the unlimited scheme. For the case of supersonic flow, a limiting function is used to reduce the gradient contribution to the reconstruction (see Sec. III.E).

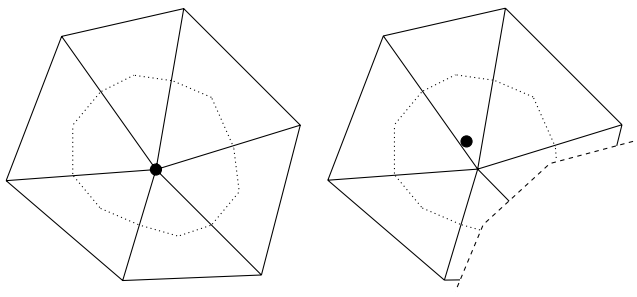
At the completion of cut-cell preprocessing, the dual polyhedra can be classified into three groups: uncut active duals interior to the computational domain, cut duals, and inactive uncut duals exterior to the computational domain. The state is stored at each node in the primal grid (Fig. 10a, filled circle). All nodes that correspond to dual polyhedra that have been cut or are inactive are removed. A new degree of freedom is inserted at each cut dual polyhedra centroid (Fig. 10b, filled circle). Multiple degrees of freedom are added when a polyhedra is split into multiple distinct regions by the cut surface.

Once a dual control volume is cut, the approximation that the state is centered at the primal node is removed and the state becomes centered at the control volume centroid. This results in a discontinuous change in location once the control volume is infinitesimally cut. This discontinuous behavior may cause difficulties for shape sensitivities and design. Removing this issue remains a topic for future work, but may be addressed by computing the uncut-cell centroids.

The median-dual triangles that surround any edges that involve a cut cell are explicitly represented and employed in flux integration. Edges that involve uncut active duals use the lumped effective areas and normals of the body-fitted scheme. Cut-cell flux integration requires more work than the body-fitted scheme, because there are multiple triangles separating the two control volumes that would be approximated as a single flux evaluation in the body-fitted scheme. It also requires more memory to store the extra triangles that would be approximated as a single effective area. The cut cells are a minority of the control volumes for a typical case, so the additional expense of using cut cells does not dominate the execution time or storage.

The body-fitted node-based scheme stores the state on the boundary of the domain. The state is interpolated between adjacent boundary nodes to integrate the boundary flux. For cut-cell boundary flux integration, the state is extrapolated with the reconstructed gradients from the cell centroids to the boundary face:

$$\mathbf{q}_{bf} = \mathbf{q}_0 + \nabla \mathbf{q}(\mathbf{x}_{b_f} - \mathbf{x}_0) \tag{11}$$



a) Uncut control volume **b) Cut-cell control volume**

Fig. 10 Dual control volumes in 2-D.

D. Gradient Reconstruction

Finite volume schemes store cell-averaged data as solution unknowns. Gradients are reconstructed from neighboring cell-averaged data to create a more accurate scheme. Barth [47] introduced a fitting procedure to reconstruct gradients,

$$\begin{bmatrix} \psi_1(x_1 - x_0) & \psi_1(y_1 - y_0) & \psi_1(z_1 - z_0) \\ \vdots & \vdots & \vdots \\ \psi_n(x_n - x_0) & \psi_n(y_n - y_0) & \psi_n(z_n - z_0) \end{bmatrix} \begin{pmatrix} \alpha_x \\ \alpha_y \\ \alpha_z \end{pmatrix} = \begin{pmatrix} \psi_1(\alpha_1 - \alpha_0) \\ \vdots \\ \psi_n(\alpha_n - \alpha_0) \end{pmatrix} \tag{12}$$

for each primitive state $\alpha \in \mathbf{q} = [\rho, u, v, w, p]^T$. For uncut cells, the cell neighbors $1, \ldots, n$ that surround the central cell 0 are often more numerous than the three unknowns, so the overdetermined system is solved with the method of least squares. For cut cells, however, the number of neighbors can be low, resulting in poor conditioning. To improve conditioning for cut cells, the gradient reconstruction system is extended to include the neighbors of cut-cell neighbors. To reduce numerical instabilities, a Gram–Schmidt QR factorization [39] is used to invert Eq. (12) by precomputing and storing R, which is only a function of the problem geometry. Mavriplis [48] discusses the properties of unweighted and various weighted reconstruction schemes.

The discrete adjoint solution exhibits extreme values in small cells that are adjacent to much larger control volumes when these small cells are included in the unweighted reconstruction scheme of the larger cells. This behavior is problematic for error estimation because of the use of a high-order recovery [28]. For the unweighted reconstruction, the reconstructed gradient is highly sensitive to the solution in small cells, causing large contributions to the adjoint residual in these small cells.

To relieve this problem, the gradient reconstruction system includes a square root of volume weighting $\psi_i = \sqrt{V_i/V_0}$. The reconstructed gradients are still exact for linear functions with this weight. This weighting has the added benefit that a cell is smoothly included or removed from the reconstruction stencil when a control volume is infinitesimally cut. This smooth transition should aid the computation of design sensitivities.

E. Reconstruction Limiting

Barth and Jespersen [46] introduced limits on an unstructured-grid reconstruction scheme to maintain monotonicity. Face reconstruction using a limiter of this form is

$$\mathbf{q}_f = \mathbf{q}_0 + \Phi \nabla \mathbf{q}(\mathbf{x}_f - \mathbf{x}_0) \tag{13}$$

where the diagonal matrix limiting function Φ is computed in each control volume. The same Φ is employed in all face reconstructions for a given control volume. This type of limiter can compromise the convergence of the flow and therefore a dual-consistent adjoint solver [49,50]. Venkatakrishnan [51] studied this limiter in its original form as well as with the limiter function held constant after iterative convergence stalls. He proposed a new limiter to improve convergence, but both the frozen scheme and new limiter can result in stalled convergence. The Venkatakrishnan limiter is not monotone, it permits under and overshoots. Frozen limiters are derivative approximations that impede error estimation, output-based adaptation, adjoint iterative convergence, and design sensitivities [18,24,52].

Balasubramanian and Newman [52] propose applying the Barth–Jespersen and Venkatakrishnan limiters on an edge-by-edge basis instead of having a single value of Φ for each control volume. They reported an improvement in the iterative convergence of the modified limiters for both the flow and adjoint systems of a wing in transonic inviscid flow. Berger et al. [53] examine edge-based limiting in an appendix. They show that edge-based limiting can introduce new extrema into the solution.

In this study, the limiter will be used in the context of an output adaptive scheme that requires the adjoint solution. An exact linearization and steady iterative convergence of the flow and adjoint solvers is paramount to the robustness of the adaptive scheme. This iterative convergence is so critical that the accuracy of the limited scheme will be sacrificed; accuracy will be regained with adaptive grid refinement. A heuristic edge-based limiter[§] is used to improve the convergence of the flow solver while providing the exact linearization required for adjoint convergence. Concessions are made to improve iterative convergence; it is not total-variation-diminishing or linearity-preserving.

The heuristic limiter was developed by examining its effect on shock capturing for regular and irregular grids and empirically adjusting its formulation to increase the width of shocks. It is a scalar limiting function ϕ that considers only the cell-averaged values of pressure and their reconstructed gradients in the cells adjacent to the face being reconstructed. Face reconstruction using a limiter of this form is

$$\mathbf{q}_f = \mathbf{q} + \phi \nabla \mathbf{q}(\mathbf{x}_f - \mathbf{x}_0) \tag{14}$$

where the scalar limiting function ϕ is computed for each face f. The same ϕ is used for the left q_{lf} and right q_{rf} face reconstructions.

The basic concept employed in this heuristic limiter is to reduce the reconstruction gradient in locations in which the pressure gradients are large relative to pressure. This clearly could result in limiting in regions for which the solution varies linearly (though with large magnitude), however, in combination with adaptation the proposed limiter has been found robust and accurate [28]. The specific form of the limiter relies on δp, a measure of the change in the pressure between the control volumes. To form δp, the reconstructed gradient of pressure for the control volumes on the right and left of the face (Fig. 11), ∇p_1 and ∇p_2, are used with the right and left extrapolation vectors to the face, δx_1 and δx_2,

$$\delta p = \left\| \begin{matrix} \delta x_{1_x} \nabla p_{1_x} - \delta x_{2_x} \nabla p_{2_x} \\ \delta x_{1_y} \nabla p_{1_y} - \delta x_{2_y} \nabla p_{2_y} \\ \delta x_{1_z} \nabla p_{1_z} - \delta x_{2_z} \nabla p_{2_z} \end{matrix} \right\| \tag{15}$$

This sensor is active for linear functions and does not specifically penalize extrema. The gradient reconstruction is reduced such that the δp sensor is large, with the intention of spreading the detected jump over a number of control volumes. Adaptation will be employed to narrow the width of the discontinuity. The tanh function is employed to smooth the combined nondimensional pressure-jump ratio,

$$\phi_{\text{heuristic}} = 1 - \tanh\left(\frac{\delta p}{\min(p_1, p_2)}\right) \tag{16}$$

and restricts the limiter to the range (0,1]. A tanh function is employed to provide a smoothly varying and differentiable function that enables residual convergence that can be impeded by a nonsmooth limiting function. This limiter is active (to some degree) in all regions with pressure variations, so it will not switch on and off intermittently during iterative convergence. The design accuracy of the limited scheme is therefore below second order, even for smooth flows. The limiter is more active when the pressure variation is significant, as compared to the local pressure.

The cut cells require pressure, extrapolated to the boundaries, to compute boundary fluxes. This reconstruction requires limits to prevent unrealizable face states and must be smoothly differentiable to facilitate iterative convergence:

$$\delta p_d = \left\| \begin{matrix} \delta x_x \nabla p_x \\ \delta x_y \nabla p_y \\ \delta x_z \nabla p_z \end{matrix} \right\| \tag{17}$$

[§]Private communication with J. A. White, 2007.

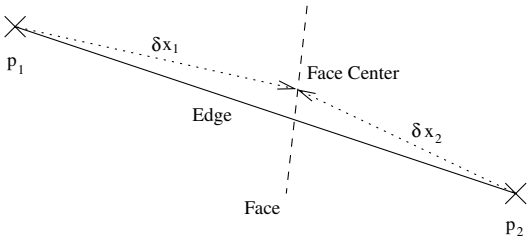


Fig. 11 Edge and face geometry.

$$\phi_{\text{extrapolation}} = 1 - \tanh\left(\frac{\delta p}{p}\right) \tag{18}$$

The extrapolation limiter is formulated to mimic the interior-face limiter using only the data from the cell adjacent to the boundary. These limiters reduce, but do not eliminate, the incidence of unrealizable face reconstructions, as discussed in Sec. III.F.

A diamond airfoil in Mach 2.0 flow at a 5 deg angle of attack is provided to illustrate the convergence issues of reconstruction limiters (see Fig. 12). The grid is anisotropically adapted to resolve the shocks. A symmetry plane grid of the 3-D extruded airfoil domain is colored with pressure in Fig. 12a. Figure 12b shows the convergence history of the residual 2-norms of the five conservation equations for the Barth–Jespersen, Venkatakrishnan, and proposed heuristic limiters. The convergence of the Barth–Jespersen and Venkatakrishnan limiters initially stall. They both converge after Φ is frozen at iteration 150. The heuristic limiter converges without modification.

A plane in the center of the 3-D domain of the extruded airfoil is shown in Fig. 13. A close-up of pressure around the diamond airfoil is shown in Fig. 13a. The minimum Φ or ϕ involved in the face reconstruction for a control volume is shown in Figs. 13b–13d. The Barth–Jespersen limiter (Fig. 13b) is active over large portions of the domain, including regions with small variations. This Φ function has a large amount of high-frequency variation. The Venkatakrishnan limiter (Fig. 13c) is the least active and its activity is narrowly restricted to the shock and strongest expansion regions. The heuristic limiter (Fig. 13d) is more active than the Venkatakrishnan limiter and includes more of the expansion region. The ϕ scalar is active in regions with moderate to large pressure variation. It has a wider active region than the Venkatakrishnan limiter and a smoother variation than the Barth–Jespersen limiter. The effect of limiter function on propagated pressure signatures for sonic boom prediction is examined in Sec. V.

F. Realizability

Even with the use of reconstruction limiters, it is still possible to reconstruct states with negative ρ or p. These unrealizable states cause catastrophic problems for flux calculations. To allow the calculation to proceed, unrealizable reconstructed face states are set to the cell-averaged value, locally reducing the scheme to first order. These unrealizable states are most common during start up from freestream conditions. As the simulation continues, the incidence of this clipping is reduced and often eliminated.

The cell-averaged state must also be prevented from reaching unrealizable values. Inviscid supersonic flow expands around corners to reach extremely low ρ and p. The updated values of ρ and p are artificially floored at 1% of freestream values. Flooring the ρ and p effectively changes the iterative time advancement scheme. This modification to the flow solver time advancement scheme can disrupt the iterative convergence of the flow and adjoint solver.

G. Body-Fitted and Cut-Cell Supersonic Vortex Uniform Refinement

A supersonic vortex has been used by a number of researchers to verify the accuracy of schemes and error-estimation techniques [11,54–58]. The 2-D geometry of the problem is given in Fig. 14. The domain is a section of an annulus. This domain is extruded for the 3-D domain used for this study. The solution only varies in the radial direction and is given by

$$\rho = \rho_i \left\{1 + \frac{\gamma+1}{2} M_i^2 \left[1 - \left(\frac{r_i}{r}\right)^2\right]\right\}^{\frac{1}{\gamma-1}} \tag{19}$$

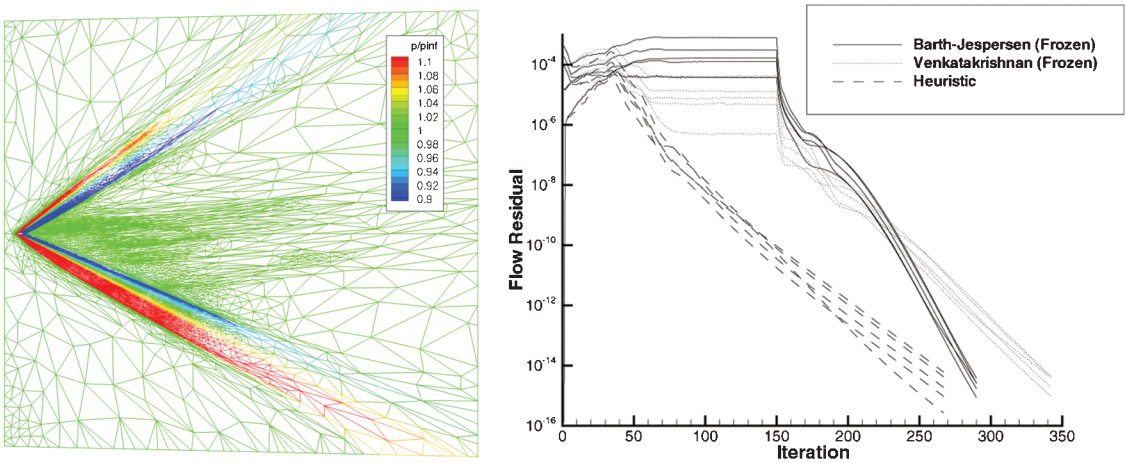
$$p = p_i \left\{1 + \frac{\gamma+1}{2} M_i^2 \left[1 - \left(\frac{r_i}{r}\right)^2\right]\right\}^{\frac{\gamma}{\gamma-1}} \tag{20}$$

$$a = \sqrt{\gamma p/\rho} \tag{21}$$

$$M = \frac{a_i M_i r_i}{a r} \tag{22}$$

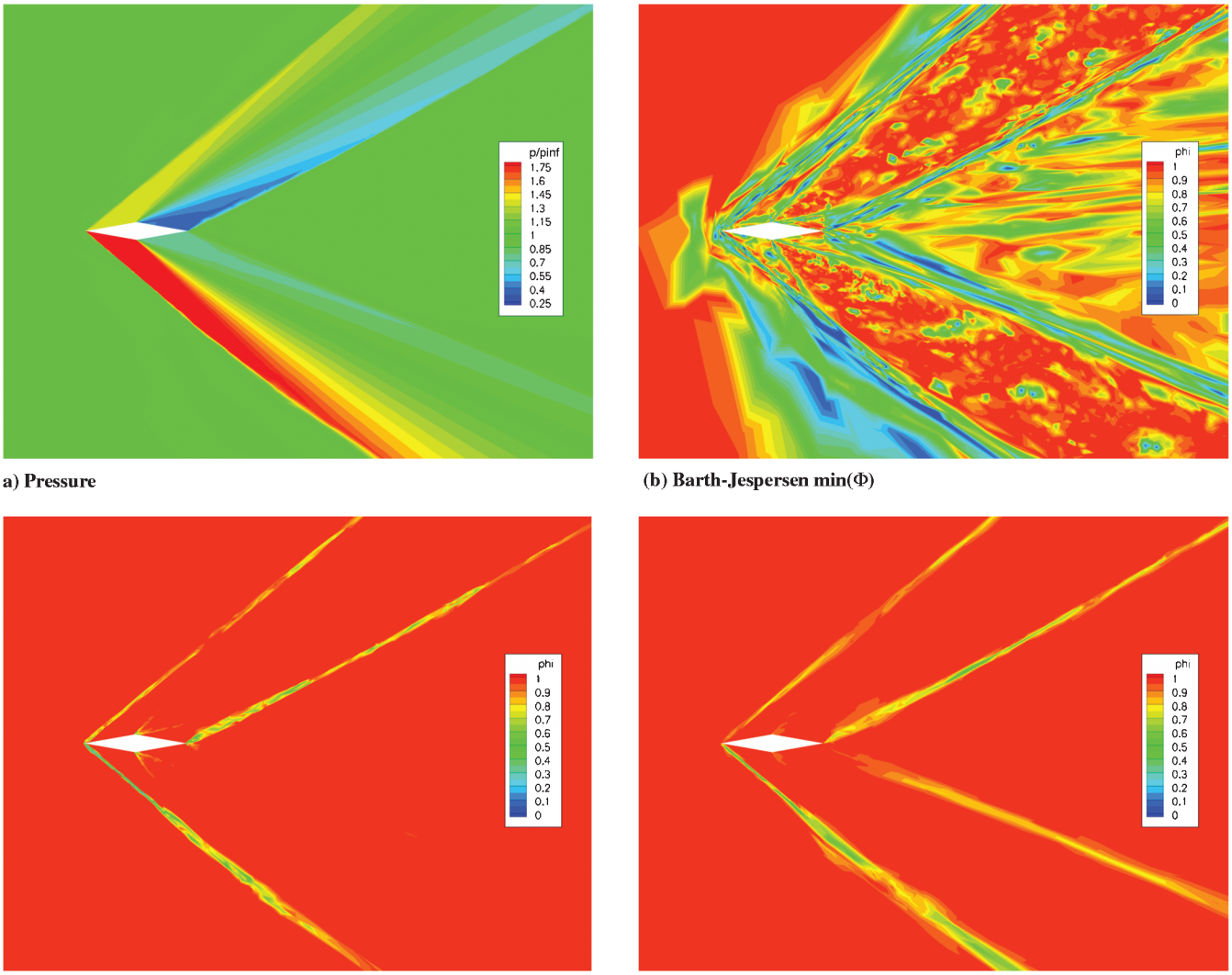
where M is Mach number and a is the speed of sound. There is an error in the equations provided in Ilinca et al. [57] The flow conditions at the inside radius of the vortex domain $r_i = 1$ are $\rho_i = 1$, $p_i = 1/\gamma$, $M_i = 2.25$. The outer radius for the vortex domain is $r_o = 1.384$. The vortex is modeled for a 90 deg turning angle.

A series of uniformly refined grids are employed to verify the design order accuracy of the existing body-fitted and current cut-cell approaches. The body-fitted approach has also been verified by Thomas et al. [59]. The coarsest body-fitted grid is shown in Fig. 15a. It is constructed of nearly right tetrahedra. The coarsest background grid used for the cut-cell approach is shown in Fig. 15b. It is constructed of right tetrahedra. The cut surface has 400 triangulated linear segments. This fine resolution may be excessive, but it



a) Pressure colored symmetry plane grid **b) Convergence history (B-J and V limiters frozen at iteration 150)**

Fig. 12 Diamond airfoil pressure colored symmetry plane grid and convergence history.



a) Pressure **(b) Barth-Jespersen min(Φ)**

c) Venkatakrishnan min(Φ) **d) Heuristic min(ϕ)**

Fig. 13 Diamond airfoil pressure and limiter function.

eliminates the geometry error of the cut surface as a source of error for this uniform refinement study.

The coefficient of lift is computed on the inner and outer curved surfaces of the vortex domain:

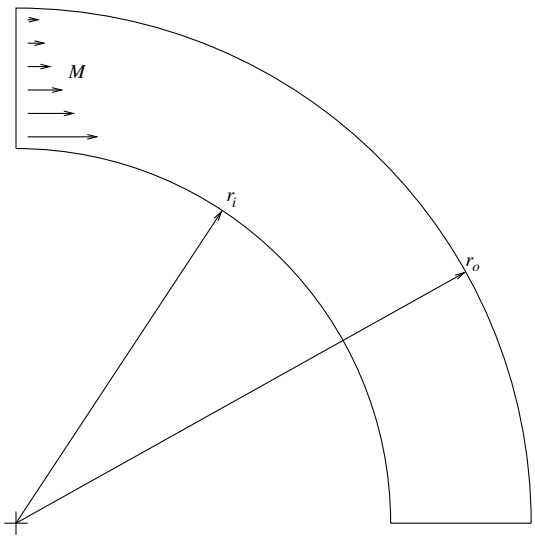


Fig. 14 Supersonic vortex geometry.

$$C_l = \int_0^{\pi/2} C_{p_o} r_o \sin(\theta)\, \mathrm{d}\theta - \int_0^{\pi/2} C_{p_i} r_i \sin(\theta)\, \mathrm{d}\theta \tag{23}$$

The exact pressure is constant along the curved surfaces:

$$p_o/p_i = \left\{1 + \frac{\gamma+1}{2} M_i^2 \left[1 - \left(\frac{r_i}{r_o}\right)^2\right]\right\}^{\frac{\gamma}{\gamma-1}} = 3.98035302289919 \tag{24}$$

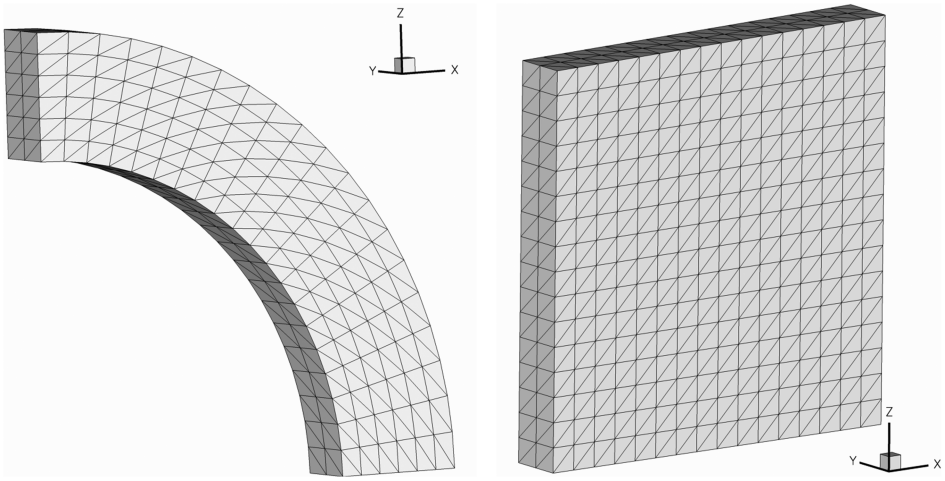
The coefficient of pressure is

$$C_{p_o} = 2\frac{p_o/p_i - 1}{\gamma M_i^2} = 0.841016726038573 \tag{25}$$

The coefficient of lift for the vortex domain per unit span is

$$C_l = r_o C_{p_o} = 1.16396714883738 \tag{26}$$

The error in computed lift divided by the exact lift is shown in Fig. 16 for a series of uniformly refined body-fitted grids and cut-cell background grids with both the heuristically limited and unlimited reconstruction schemes. The inflow conditions are specified as the analytic solution. The outflow conditions are taken from the interior of the domain. The tangency boundary conditions are applied to the inner and outer boundaries as well as the two sides. The body-fitted and cut-cell methods employing the heuristic limiter have a similar lift error level. This error level is higher than the unlimited reconstruction schemes. The error introduced by the heuristic limiter



a) Uniformly refined body-fitted grid (441 control volumes)

b) Uniformly refined cut-cell background grid (867 control volumes uncut and 409 control volumes after cut)

Fig. 15 Coarsest uniformly refined body-fitted and cut-cell grids.

does not appear to be exacerbated by the irregular shape of the cut cells. Both unlimited reconstruction methods asymptote to second order as indicated by the triangle with a slope of two. The unlimited cut-cell method has more error than the unlimited body-fitted method for the same characteristic length (degrees of freedom). The higher error level of the cut-cell method is offset by an increase in the adaptive mechanics robustness, which can produce more efficient grids.

IV. Output-Based Adaptation

Venditti [14] describes an output-based error-estimation and adaptation scheme. The adaptation intensity I^e_κ for each control volume κ is formed on an embedded grid:

$$I^e_\kappa = \frac{1}{2}\sum_{i=1}^{5}\{|[R^\lambda(\lambda^H)]_{i,\kappa}[Q^H - Q^L]_{i,\kappa}| + |[\lambda^H - \lambda^L]_{i,\kappa}[R(Q^H)]_{i,\kappa}|\} \tag{27}$$

where λ^H, λ^L, Q^H, and Q^L are the high- and low-order interpolants of the adjoint and flow solutions on the embedded grid. The flow R and adjoint R^λ residual operators are also on the embedded grid. To formulate the error estimate, an embedded grid is required. Constructing the entire embedded grid can be infeasible for large 3-D grids and has prevented the use of adjoint error-estimation techniques for large-scale problems even with a parallel implementation [18].

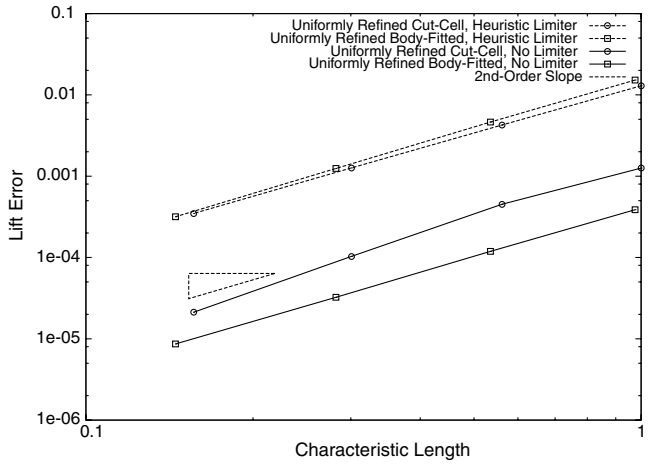


Fig. 16 Lift error convergence for uniformly refined body-fitted and cut-cell grids.

While the embedded grid can be formed in sections, this increases the error-estimation scheme complexity. Forming a portion or the entire embedded grid is also complicated by the need to respect curved boundaries and recompute the intersection tests of cut cells. These difficulties have motivated the desire to employ only the current grid in the error-estimation procedure. A procedure is described that obtains an indicator for output adaptation with the current grid, but does not provide a functional error correction.

Park [28] provides a derivation of the single-grid adaptive indicator by placing it in the context of the embedded grid approach of Venditti [14]. In the interest of brevity, the single-grid error estimation and adaptive indicator is provided without a derivation:

$$I_\kappa = \frac{1}{2}\sum_{i=1}^{5}\{|[R^\lambda(\hat{\lambda})]_{i,\kappa}[\hat{Q} - \bar{Q}]_{i,\kappa}| + |[\hat{\lambda} - \bar{\lambda}]_{i,\kappa}[R(\hat{Q})]_{i,\kappa}|\} \tag{28}$$

It has the same pieces as the Venditti [14] error indicator, where the five conservation equations are contracted by the summation over i. The vector $I \in \mathcal{R}^N$ has a single value for each grid control volume κ. The original residual operators are used. The $\hat{\lambda}$ and $\hat{Q}$ higher-order reconstructions and the $\bar{\lambda}$, and $\bar{Q}$ lower-order reconstructions on the current grid are described by Park [28]. The $\hat{\lambda}$ and $\hat{Q}$ reconstructions are formed with a fit of quadratic functions to cell-averaged states and their gradients. The difference between the $\hat{}$ and $\bar{}$ reconstructions is intended to provide adequate guidance for the relative distribution of error, not a sharp bound on error.

Venditti [14] provides a procedure to calculate a new grid spacing request from the adaptive indicator I_κ and an error tolerance tol_Ω. The adaptation indicator is summed to find the global indicator $I_\Omega = \sum I_\kappa$. The ratio of the remaining error to a user-specified error tolerance tol_Ω is

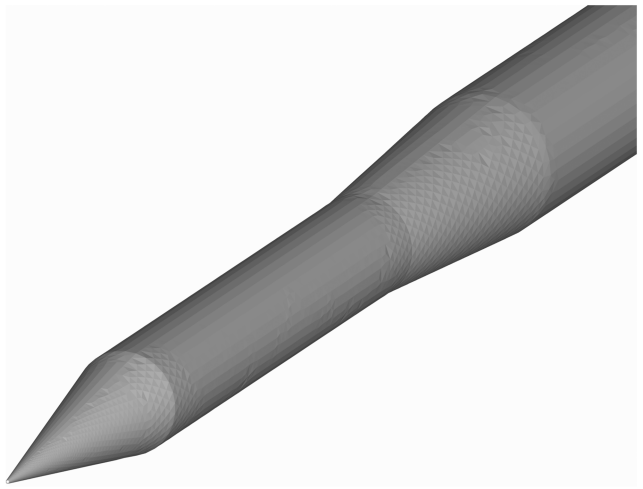
$$\epsilon_\Omega = \frac{I_\Omega}{\text{tol}_\Omega} \tag{29}$$

The ratio of the control volume indicator to an equal share of tol_Ω is

$$\epsilon_\kappa = \frac{N}{\text{tol}_\Omega} I_\kappa \tag{30}$$

where N is the number of control volumes. When a cost function does not have an intuitive error tolerance, i.e., sonic boom surface pressure integrals, the tol_Ω is set to half I_Ω at each adaptive iteration. The requested isotropic element length h is computed with an estimate of the spacing on the original mesh h^0 and the global and local error ratios,

a) Drawing



b) Shaded surface

Fig. 17 Double-cone-cylinder geometry [61].

$$h_\kappa = h_\kappa^0 \left(\frac{1}{\epsilon_\Omega \epsilon_\kappa} \right)^\omega \tag{31}$$

where an exponent of $\omega = 0.20$ is based on a *a priori* estimate of the spatial error convergence. The anisotropy of mesh elements is based on the Mach Hessian, where the element size in the smallest spacing direction is dictated by the adjoint adaptation parameter. The parallel metric-based grid mechanics described by Park [28] and Park and Darmofal [60] are used to modify tetrahedral background grid.

V. Application to Sonic Boom Prediction

The output-based adaptation algorithm is applied to accurately predict near-field signatures by reducing the estimated error in the calculation of an offbody pressure integral. The integral of quadratic pressure deviation over a surface s in the domain is

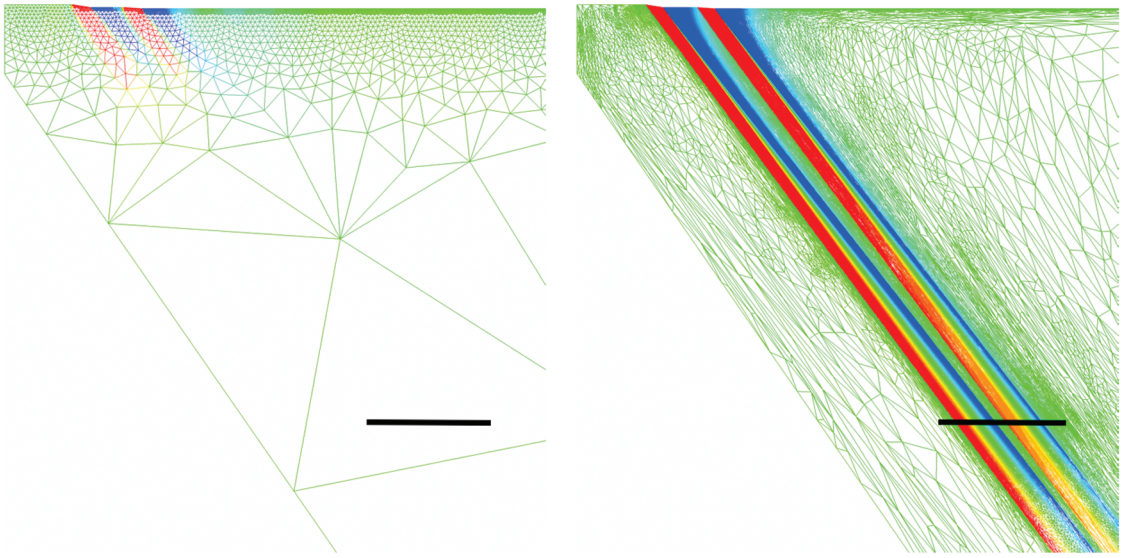
$$f = \frac{1}{A_s} \iint_s \left(\frac{p - p_\infty}{p_\infty} \right)^2 \mathrm{d}s \tag{32}$$

where A_s is the area of the integration surface. This focuses the adaptation on improving the calculation of pressure deviation from freestream near this surface. Previous applications have been performed with the integral of pressure deviation [18,19]. However, the square of this deviation has been shown to produce more accurate signatures with less control volumes [24]. A cylindrical integration surface is employed that is aligned to the x axis and optionally clipped in the Cartesian directions.

A. Double-Cone Cylinder

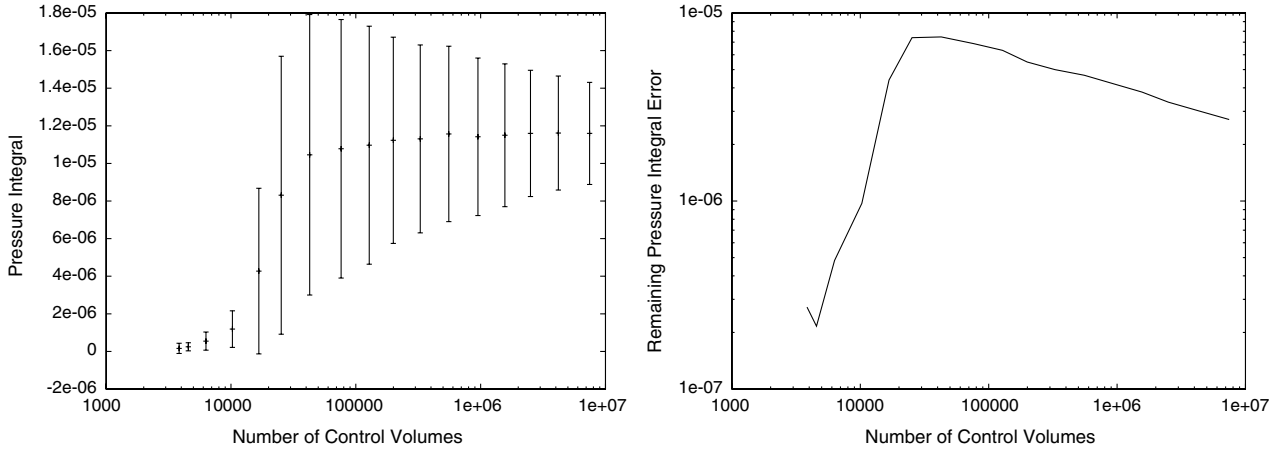
A double-cone geometry, denoted as model 8 in a 1965 wind-tunnel report [61], is shown in Fig. 17 with a shaded triangular surface grid. The same case was employed to evaluate [18] and then validate [19] a parallel adaptive body-fitted grid approach. This configuration has also been used by other researchers to evaluate their signature prediction techniques [5,8]. The pressure integral output function was defined as a cylinder, six body lengths in radius, centered about the geometry axis. The cylinder is clipped forward of three body lengths behind the nose, aft of nine body lengths behind the nose, and outside of 0.1 body lengths off the centerline to focus only on the region where wind-tunnel data are available. The surface grid is Boolean-subtracted from a 9 deg wedge-shaped background tetrahedral volume grid. A tangency boundary condition is applied to the radial faces of the wedge. A symmetry plane of the volume grid is shown in Fig. 18. The location of the integration surface is denoted with a black line. The initial grid (4000 control volumes) was created with no prior knowledge of where the shocks would propagate through the domain (Fig. 18a). The freestream Mach number is 1.26 and the heuristic limiter is employed during adaptation. The parallel execution scheme used 32 partitions, and the 17th adapted grid (7,500,000 control volumes) is shown in Fig. 18b. The shocks have been implicitly targeted and refined to propagate the signal to the pressure integral surface. The anisotropy of the grid, based on the Mach Hessian, is clearly evident. This anisotropy reduces the number of required control volumes.

The adaptation history of the pressure integral with error bars the width of twice the remaining error estimate I_Ω are shown in Fig. 19. Error is underestimated on the initial few adapted grids before the shocks are propagated to the integration surface. Once this connection is established, the error is reduced. The adaptation history of the pressure signature extracted at six body lengths is shown in Fig. 20. The circular symbols are digitized from a wind-tunnel report [61]. The solid line is the final adapted signature. The signal is absent on the original coarse grid. The extrema of the pressure signal start to form and grow. The inflection points at $x/l = 2.3$ is the last part of the



a) Initial grid **b) Output adapted grid**

Fig. 18 Symmetry planes of initial and output adapted double-cone 3-D volume grids with integration surface shown as black line.



(a) Pressure integral **(b) Pressure integral remaining error**

Fig. 19 Model 8 pressure integral and uncertainty convergence at six body lengths.

signal to form. The over- and undershoots of the signal intensify on the final few adapted grids as the grid-shock alignment improves. The final adapted grid is simulated with the Venkatakrishnan, heuristic, and Barth–Jespersen limiters in Fig. 21. The circular symbols are digitized from a wind-tunnel report [61]. The Venkatakrishnan limiter has similar over- and undershoots to the heuristic limiter. The Barth–Jespersen limiter produces a signature without over- and undershoots. All of these limiters have very similar signatures, except at the discontinuities.

B. Straight-Line Segmented Leading Edge

The straight-line segmented-leading-edge (SLSLE) low-boom configuration (Fig. 22) is described by Mack and Kuhn [62,63]. These reports provide wind-tunnel data from two tests, performed at the NASA Langley Research Center (LaRC) Unitary Plan Wind Tunnel Facility [62] and the NASA John H. Glenn Research Center at Lewis Field (GRC) 10×10 ft Wind Tunnel Facility [63]. The test condition is Mach 2.0. The model geometry is rotated to provide the wind-tunnel lift coefficient of $C_L = 0.08309$ at zero deg angle of attack [4]. The configuration has a finite thickness trailing edge, which was modeled with a transpiration boundary condition [28] to prevent a strong inviscid supersonic corner flow expansion. Preliminary body-fitted results for this configuration [18] extended the blunt trailing edge to sharp trailing edge to avoid the strong supersonic expansion.

The original symmetry plane and cut-surface grid colored with pressure is shown in Fig. 23a. A linear distribution of pressure is shown in each control volume, resulting in a discontinuous pressure distribution on the surface. The initial background grid is isotropic.

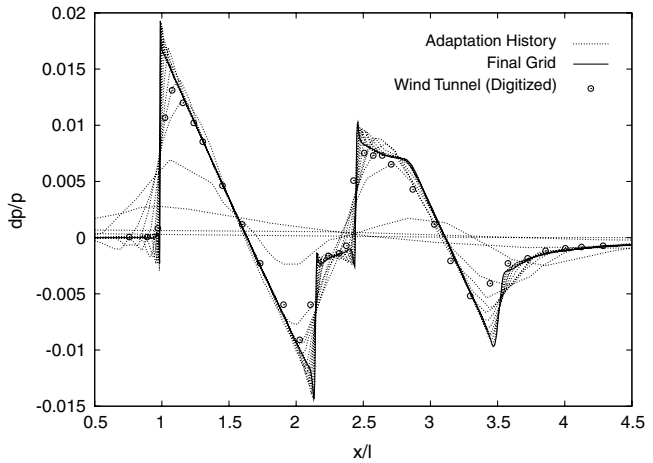


Fig. 20 Model 8 pressure signature adaptation history at six body lengths.

The final adapted symmetry plane and cut-surface grid colored with pressure is shown in Fig. 23b. The initial background grid contains 40,000 control volumes, and the final adapted background grid contains 5,700,000 control volumes.

The initial (Fig. 24a) and final (Fig. 24b) grid integration surfaces are colored with pressure deviation from freestream. These are the cylindrical integration surfaces used to compute the output for adaptation. The cylinder has a radius of 10 body lengths, which is the location of the most distant available wind-tunnel data. The cylinder is restricted to its lower quadrant between 32.6 and 41.0 body lengths aft of the model. The integration surface is piecewise linear in each tetrahedral background grid element. The pressure signature is not visibly propagated to the initial integration surface, which is poorly resolved due to the initial coarse grid. The final adapted grid integration surface is a much better approximation of a cylinder due to the background grid refinement. The peak signature pressure is larger at the horizon than the centerline, because the model is designed to have a reduced centerline pressure signature.

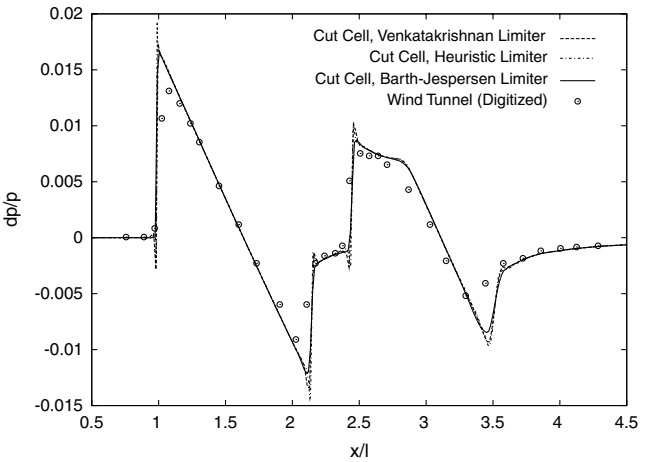
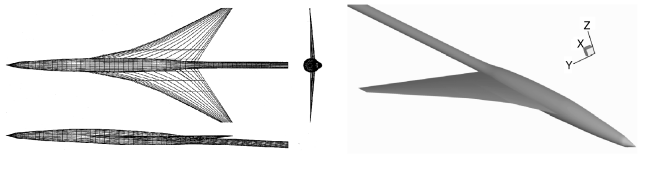
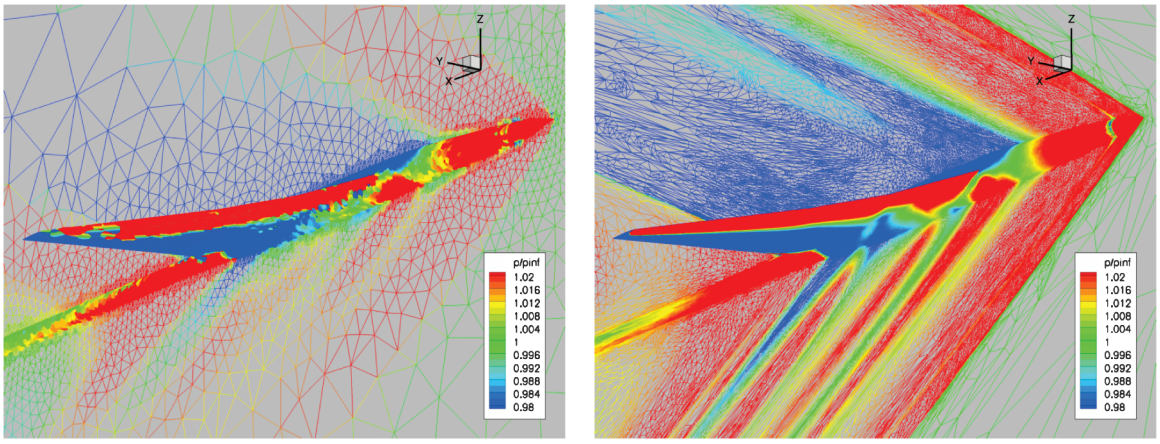


Fig. 21 Model 8 final adapted pressure signature at six body lengths for various limiters.



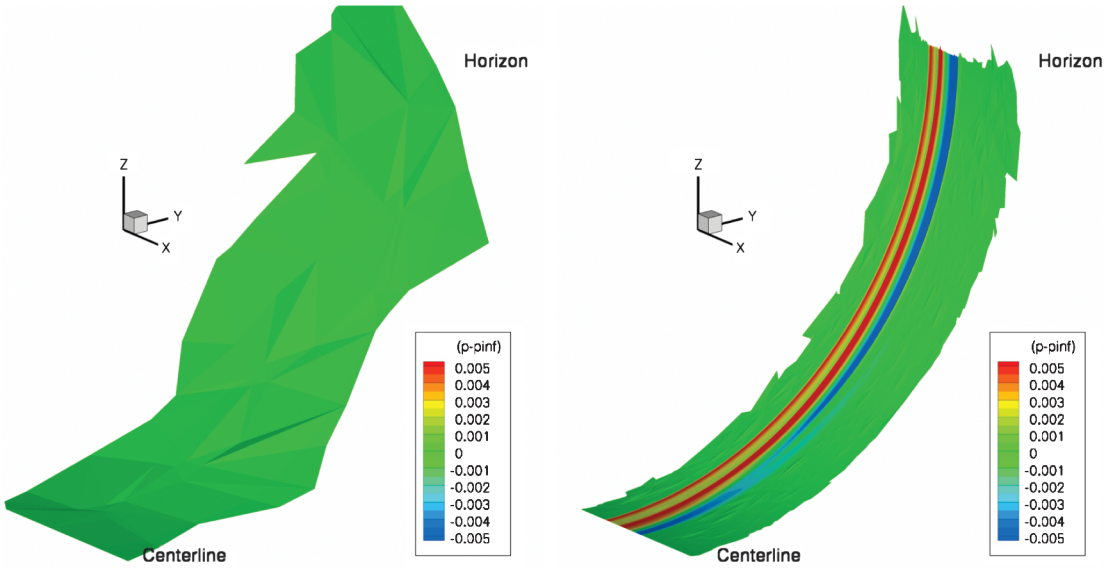
a) Three-view drawing **b) Shaded Surface**

Fig. 22 Low-boom-wing body-sting geometry [64].



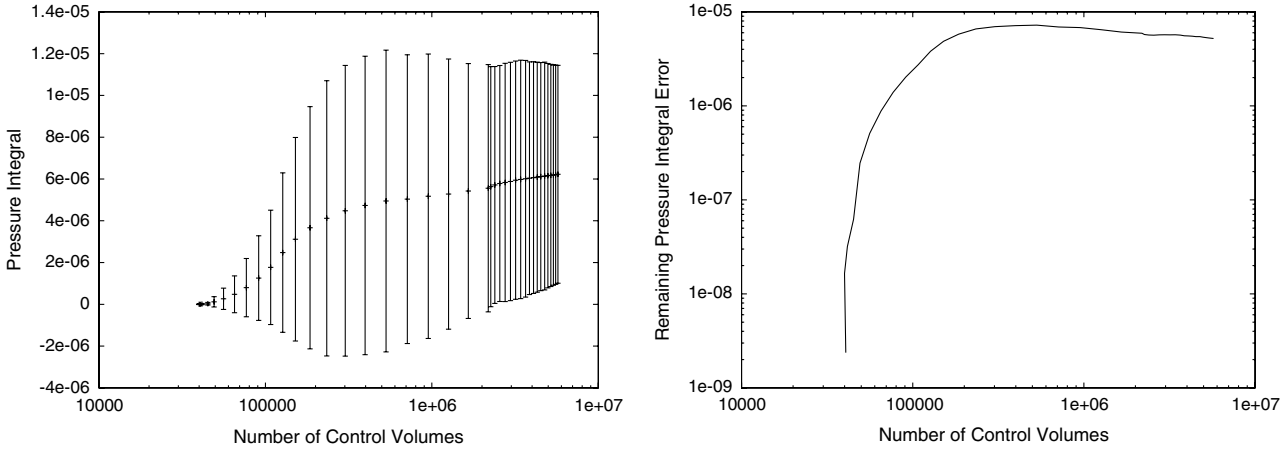
a) Initial grid **b) Final grid**

Fig. 23 SLSLE surface grid colored with pressure.



a) Initial grid **b) Final grid**

Fig. 24 SLSLE pressure on quarter-cylinder integration surface 10 body lengths below the model.



a) Pressure integral **b) Pressure integral remaining error**

Fig. 25 SLSLE pressure integral and uncertainty convergence at 10 body lengths.

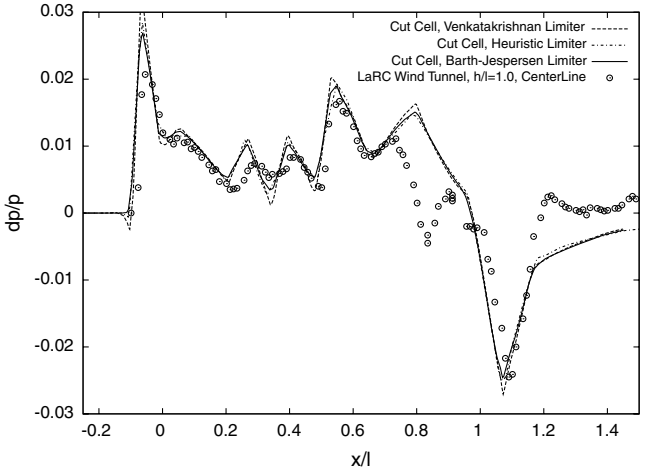


Fig. 26 SLSLE final adapted pressure signature at 1.0 body lengths for various limiters.

The adaptation history of the pressure integral and its error estimation is shown in Fig. 25. The requested error tolerance tol_Ω is set to half of I_Ω at each adaptive iteration for grids sized less than 2,000,000. Above 2,000,000 control volumes, the tol_Ω is set to I_Ω, reducing the rate of grid growth. The goal of increasing tol_Ω is to obtain more resolved results with a more efficiently distributed and aligned grid at the expense of wall clock time and more adaptation cycles [26]. The change in requested error tolerance is observed as a reduction in grid growth per adaptive iteration in Fig. 25a. This case shows a less dramatic reduction in the remaining error estimate over the final few grids than the previous cases, which may be due to the less aggressive $\text{tol}_\Omega = I_\Omega$ on the final grids.

The heuristic limiter is employed during adaptation. The Venkatakrishnan, heuristic, and Barth–Jespersen limiters are applied on the final grid. The resulting pressure signatures at one body length are shown in Fig. 26. The signatures of all three limiters are very similar, except near discontinuities. The Barth–Jespersen limiter reduces the over- and undershoots of the bow and tail shocks. The difference between the different limiter signatures is greater for this case than the cone-cylinder and delta-wing/body cases.

Centerline pressure signatures are presented in Fig. 27 for 1.0, 1.5, 2.0, and 2.5 body lengths below the model. LaRC wind-tunnel data are available for all four locations, but the closest GRC wind-tunnel data are available at 2.5 body lengths below the model. The LaRC and GRC wind-tunnel measurements are generally in good agreement at 2.5 body lengths in Fig. 27, but the small differences of the two measurements gives an indication of the level of uncertainty in the measurements. The agreement between the wind tunnel and computed signatures is good, except in the region near $x/l = 0.8$. Other investigators [4,18] also showed a difference between wind-tunnel and computed pressure signatures at $x/l = 0.8$. Both wind-tunnel measurements agree favorably with each other near $x/l = 0.8$. The signature sensitivity in this mismatch region was examined with the adjoint solution by Park [28]. When the wind-tunnel model was scanned a mismatch between the as-built and as-designed geometry was noted in a region that affects the signature near $x/l = 0.8$. Correcting this discrepancy resulted in a partial improvement in simulation and measurement comparison [28].

Figure 28 compares the adapted cut-cell method with the GRC wind-tunnel measurement at 10 body lengths. This is the same distance as the integration surface. The shock strength increases away from the configuration centerline. The front portion of the signature is well predicted in both Figs. 27 and 28. The aft portion of the computed signal shows the largest difference from the wind-tunnel data at all propagation distances. The discrepancy between the wind-tunnel and computed signatures near $x/l = 0.8$ decreases for the signatures away from the model centerline. This is the first published CFD prediction of the offcenterline signatures.

None of the previously reported simulations of this model provided offcenterline signature comparisons at 10 body lengths. Preliminary body-fitted grid output-based adaptation results by

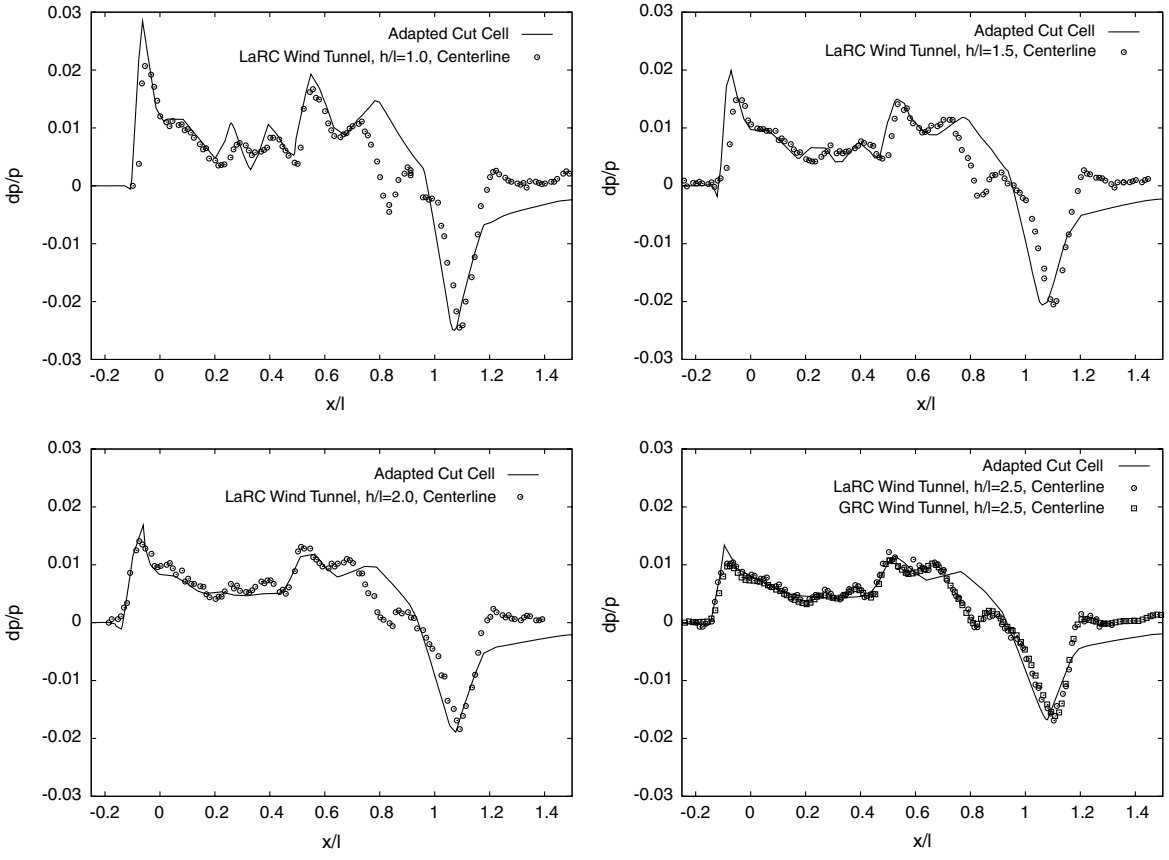


Fig. 27 SLSLE centerline pressure signatures for various locations below the model.

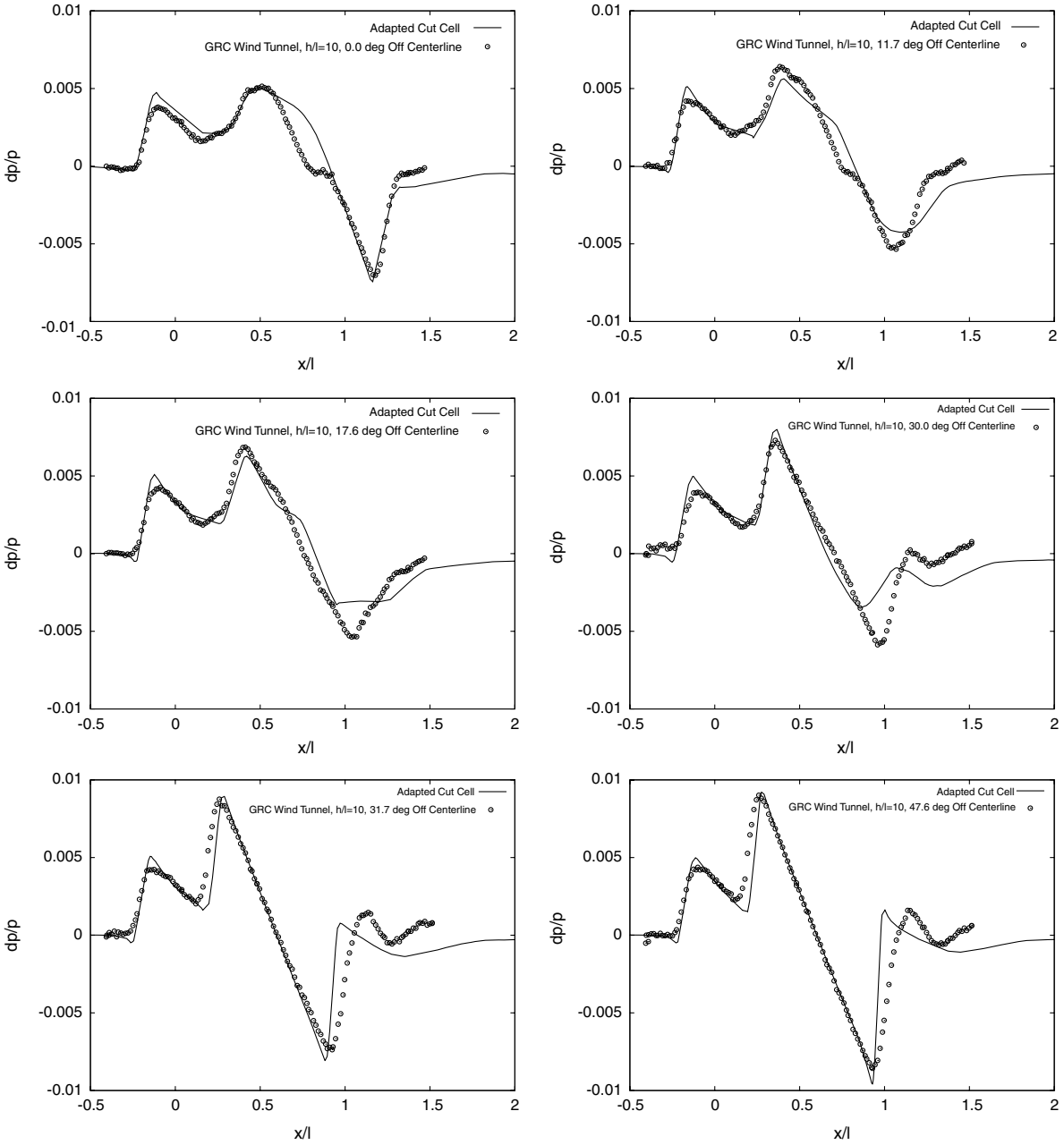


Fig. 28 SLSLE pressure signatures for various locations 10 body lengths below the model.

Lee-Rausch et al. [18] targeted the centerline pressure signature at 2.5 body lengths with a final grid of 2 million control volumes. Laflin at al. [4] used a hybrid method with an unstructured extreme-near-field grid of 1.25 million nodes adapted to propagate the signal less than 0.25 body lengths. A structured grid was employed to propagate the signal at 0.25 body length to the wind-tunnel data at 2.5 body lengths. Carter and Deere [64] showed centerline comparisons at 10 body lengths, but the grid size was not reported.

C. Low-Boom Wing Tail (LBWT-4)

The low-boom-wing tail (LBWT-4) [65] has a cranked-arrow wing with conventional tail surfaces and was designed [66] to accommodate four nacelles with boundary-layer diverters (Fig. 29). This model is also referred to as WBVHN4 [67]. This case was included in the NASA Fundamental Aeronautics Program Sonic Boom Workshop. The model was rotated 2 deg nose-up to provide the wind-tunnel normal force of 0.074. The flow conditions are Mach 2.0 and 0 deg angle of attack. Pressure on the complete configuration for the final adapted grid is shown in Fig. 30a. The forward portion of the configuration has lower magnitude shocks and expansions than the complex shock and expansion interactions near the nacelles and tail. The cutting surface grid was obtained from airplane simulations [65] and is relatively coarse. The facets of the cutting surface are visible as the pressure discontinues on the nacelles in Fig. 30b. The shock–shock interactions generated by the nacelles and boundary-layer diverters are also apparent. The interior of the nacelles has been modified to account for boundary-layer growth, which increased the

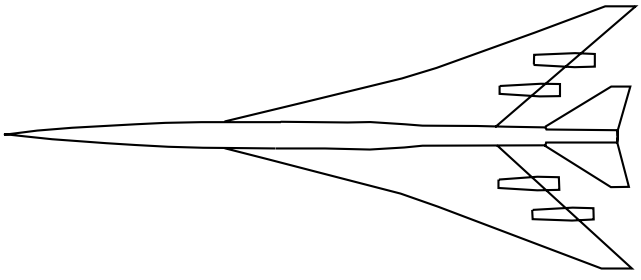
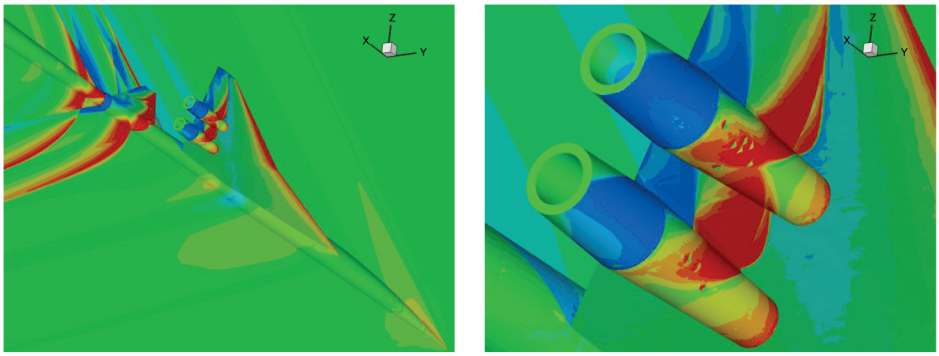
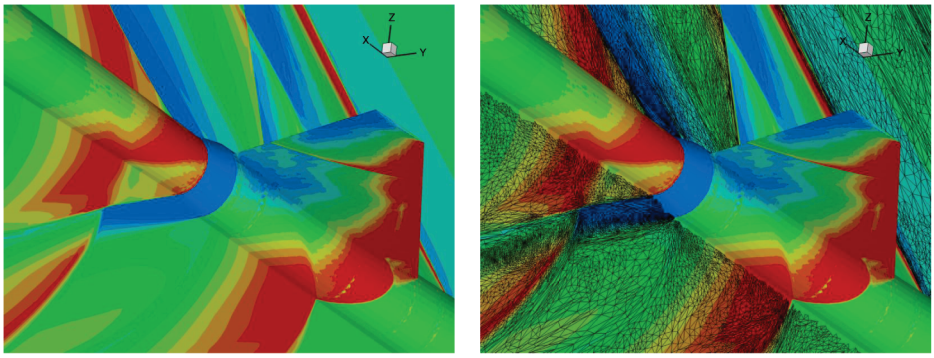


Fig. 29 LBWT planform [65].



a) Pressure contours for the entire configuration **b) Pressure contours for a nacelle detail**

Fig. 30 LBWT surface pressure.



a) Pressure contours near configuration tail **b) Pressure contours and symmetry plane grid near configuration tail**

Fig. 31 LBWT aft detail.

base area [65]. These base areas are treated with a transpiration boundary condition. The pressure on sting-body-tail juncture of the configuration is shown in Fig. 31. The symmetry plane of the background grid is over plotted in Fig. 31b. The final adapted grid is well aligned with the complex network of shocks and expansions.

The pressure signature at 1.16 body lengths is shown in Fig. 32. The wind-tunnel report [65] was scanned and digitized to provide comparison. The signature computed on the final adapted grid and wind-tunnel measurement compares favorably for the forward portion of the signal. The signature aft of a $x/l = 1.1$ has different details in the computed and measured signal. These aft locations include the influence of the nacelles and boundary-layer diverters. The aft portion of this signal had the greatest difference between the simulation methods used in the NASA Fundamental Aeronautics Program Sonic Boom Workshop. None of these methods predicted the location of the small peak in the wind-tunnel measurement at $x/l = 1.2$.

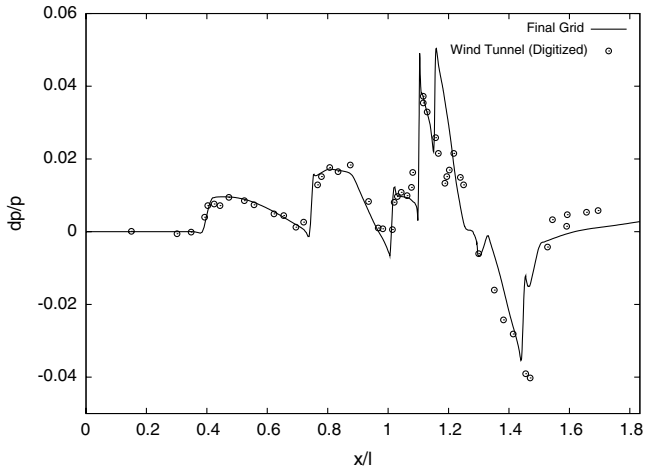


Fig. 32 LBWT final adapted pressure signature at 1.16 body lengths.

VI. Conclusions

A cut-cell approach to CFD is described that uses the median dual of tetrahedral background grid with a topologically consistent cut-cell geometry calculation method. The discrete adjoint is also calculated, which permits adaptation based on improving the calculation of a specified output (offbody pressure signature). Output-based adaptation simulations are presented for a cone-cylinder and low-boom configurations. These predicted signatures are compared to wind-tunnel measurements to validate the method for sonic boom prediction. These predicted signatures are compared to wind-tunnel measurements up to 10 body lengths below the model, providing simultaneous on- and offcenterline predictions.

The use of cut cells with an output-based adaptive scheme automates the volume grid generation task after the triangular surface mesh is generated. This robust adaptation scheme allows extremely coarse isotropic initial grids that can be generated without a priori knowledge of shock locations or Mach angles. The heuristic limiter dramatically improves the iterative convergence of the flow solver and robustness of the adjoint solution scheme. This heuristic limiter produces similar signatures to the existing Venkatakrishnan and Barth–Jespersen limiters. The general anisotropy of the adapted background grids allows for accurate centerline and offcenterline signal prediction. The forward portion of the signatures is well predicted for all cases. The aft signatures have the most difference between measurements and simulation. These differences may be due to differences in geometry or a result of viscous flow phenomena that is missing from the Euler calculations.

Acknowledgment

A sincere thank you to Jeffery White of the NASA Langley Research Center Computational Aerosciences Branch for his assistance with shock capturing by providing a limiter that enables excellent iterative convergence.

References

[1] Plotkin, K. J., "State of the Art of Sonic Boom Modeling," *Journal of the Acoustical Society of America*, Vol. 111, No. 1, Jan. 2002, pp. 530–536. doi:10.1121/1.1379075

[2] Carlson, H. W., "Experimental and Analytic Research on Sonic Boom Generation at NASA," *Sonic Boom Research*, edited by A. R. Seebass, NASA SP-147, April 1967, pp. 9–23.

[3] Campbell, R., Carter, M., Deere, K., and Waithe, K. A., "Efficient Unstructured Grid Adaptation Methods for Sonic Boom Prediction," AIAA Paper 2008-7327, 2008.

[4] Laflin, K. R., Klausmeyer, S. M., and Chaffin, M., "A Hybrid Computational Fluid Dynamics Procedure for Sonic Boom Prediction," AIAA Paper 2006-3168, 2006.

[5] Kandil, O., and Ozcer, I. A., "Sonic Boom Computations for Double-Cone Configuration Using CFL3-D, FUN3D and Full-Potential Codes," AIAA Paper 2006-414, 2006.

[6] Waithe, K. A., "Application of USM3-D for Sonic Boom Prediction by Utilizing a Hybrid Procedure," AIAA Paper 2008-129, 2008.

[7] Loseille, A., Dervieux, A., Frey, P., and Alauzet, F., "Achievement of Global Second Order Mesh Convergence for Discontinuous Flows with Adapted Unstructured Meshes," AIAA Paper 2007-4186, 2007.

[8] Ozcer, I. A., and Kandil, O., "FUN3D/OptiGRID Coupling for Unstructured Grid Adaptation for Sonic Boom Problems," AIAA Paper 2008-61, 2008.

[9] Peraire, J., Peirò, J., and Morgan, K., "Adaptive Remeshing for Three-Dimensional Compressible Flow Computations," *Journal of Computational Physics*, Vol. 103, No. 2, 1992, pp. 269–285. doi:10.1016/0021-9991(92)90401-J

[10] Baker, T. J., "Mesh Adaptation Strategies for Problems in Fluid Dynamics," *Finite Elements in Analysis and Design*, Vol. 25, Nos. 3–4, 1997, pp. 243–273. doi:10.1016/S0168-874X(96)00032-7

[11] Aftosmis, M. J., and Berger, M. J., "Multilevel Error Estimation and Adaptive *h*-Refinement for Cartesian Meshes with Embedded Boundaries," AIAA Paper 2002-0863, 2002.

[12] Rannacher, R., "Adaptive Galerkin Finite Element Methods for Partial Differential Equations," *Journal of Computational and Applied Mathematics*, Vol. 128, 2001, pp. 205–233. doi:10.1016/S0377-0427(00)00513-6

[13] Pierce, N. A., and Giles, M. B., "Adjoint Recovery of Superconvergent Functionals from PDE Approximations," *SIAM Review*, Vol. 42, No. 2, 2000, pp. 247–264. doi:10.1137/S0036144598349423

[14] Venditti, D. A., "Grid Adaptation for Functional Outputs of Compressible Flow Simulations," Ph.D. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, 2002.

[15] Fidkowski, K. J., and Darmofal, D. L., "Output Error Estimation and Adaptation in Computational Fluid Dynamics: Overview and Recent Results," AIAA Paper 2009-1303, 2009.

[16] Barter, G. E., "Shock Capturing with PDE-Based Artificial Viscosity for an Adaptive, Higher-Order Discontinuous Galerkin Finite Element Method," Ph.D. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, 2008.

[17] Wintzer, M., Nemec, M., and Aftosmis, M. J., "Adjoint-Based Adaptive Mesh Refinement for Sonic Boom Prediction," AIAA Paper 2008-6593, 2008.

[18] Lee-Rausch, E. M., Park, M. A., Jones, W. T., Hammond, D. P., and Nielsen, E. J., "Application of a Parallel Adjoint-Based Error Estimation and Anisotropic Grid Adaptation for Three-Dimensional Aerospace Configurations," AIAA Paper 2005-4842, 2005.

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

[20] Young, D. P., Melvin, R. G., Bieterman, M. B., Johnson, F. T., Samant, S. S., and Bussoletti, J. E., "A Locally Refined Rectangular Grid Finite Element Method: Application to Computational Fluid Dynamics and Computational Physics," *Journal of Computational Physics*, Vol. 92, No. 1, 1991, pp. 1–66. doi:10.1016/0021-9991(91)90291-R

[21] Charlton, E. F., and Powell, K. G., "An Octree Solution to Conservation Laws Over Arbitrary Regions (OSCAR)," AIAA Paper 97-198, 1997.

[22] Aftosmis, M. J., Berger, M. J., and Melton, J. E., "Robust and Efficient Cartesian Mesh Generation for Component-Based Geometry," *AIAA Journal*, Vol. 36, No. 6, 1998, pp. 952–960. doi:10.2514/2.464

[23] Domel, N. D., and Karman, S. L., Jr., "Splitflow: Progress in 3-D CFD with Cartesian Omni-Tree Grids for Complex Geometries," AIAA Paper 2000-1006, 2000.

[24] Nemec, M., Aftosmis, M. J., and Wintzer, M., "Adjoint-Based Adaptive Mesh Refinement for Complex Geometries," AIAA Paper 2008-725, 2008.

[25] Löhner, R., Baum, J. D., Mestreau, E., Sharov, D., Charman, C., and Pelessone, D., "Adaptive Embedded Unstructured Grid Methods," *International Journal for Numerical Methods in Engineering*, Vol. 60, 2004, pp. 641–660. doi:10.1002/nme.978

[26] Fidkowski, K. J., and Darmofal, D. L., "A Triangular Cut-Cell Adaptive Method for High-Order Discretizations of the Compressible Navier–Stokes Equations," *Journal of Computational Physics*, Vol. 225, No. 2, Aug. 2007, pp. 1653–1672. doi:10.1016/j.jcp.2007.02.007

[27] Fidkowski, K. J., "A Simplex Cut-Cell Adaptive Method for High-Order Discretizations of the Compressible Navier–Stokes Equations," Ph.D. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, 2007.

[28] Park, M. A., "Anisotropic Output-Based Adaptation with Tetrahedral Cut Cells for Compressible Flows," Ph.D. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, Sept. 2008.

[29] Li, X., Shephard, M. S., and Beall, M. W., "Accounting for Curved Domains in Mesh Adaptation," *International Journal for Numerical Methods in Engineering*, Vol. 58, No. 1, 2000, pp. 247–276.

[30] Haimes, R., and Crawford, C., "Unified Geometry Access for Analysis and Design," *12th International Meshing Roundtable*, Sandia National Labs., Albuquerque, NM, Sept. 2003, pp. 21–31.

[31] Aftosmis, M. J., Delanaye, M., and Haimes, R., "Automatic Generation of CFD-Ready Surface Triangulations from CAD Geometry," AIAA Paper 99-0776, 1999.

[32] Jones, W. T., "GridEx—An Integrated Grid Generation Package for CFD," AIAA Paper 2003-4129, 2003.

[33] Kalantari, I., and McDonald, G., "A Data Structure and an Algorithm for the Nearest Point Problem," *IEEE Transactions on Software Engineering*, Vol. SE-9, No. 5, Sept. 1983, pp. 631–634. doi:10.1109/TSE.1983.235263

[34] Andrews, L., "A Template for the Nearest Neighbor Problem," C/C^{++} *Users Journal*, Vol. 19, No. 11, Nov. 2001, pp. 40–49.

[35] Edelsbrunner, H., and Mücke, E. P., "Simulation of Simplicity: a Technique to Dope with Degenerate Cases in Geometric Algorithms," *ACM Transactions on Graphics*, Vol. 9, No. 1, Jan. 1990, pp. 66–104. doi:10.1145/77635.77639

[36] Green, P. J., and Sibson, R., "Computing Dirichlet Tessellations in the Plane," *Computer Journal*, Vol. 21, No. 2, 1978, pp. 168–173.

[37] Shewchuk, J. R., "Triangle: Engineering a 2-D Quality Mesh Generator and Delaunay Triangulator," *Applied Computational Geometry: Towards Geometric Engineering*, edited by M. C. Lin and D. Manocha, Lecture Notes in Computer Science, Vol. 1148, Springer–Verlag, Berlin, May 1996, pp. 203–222.

[38] George, P.-L., Hecht, F., and Saltel, E., "Automatic Mesh Generator with Specified Boundary," *Computer Methods in Applied Mechanics and Engineering*, Vol. 92, No. 3, 1991, pp. 269–288. doi:10.1016/0045-7825(91)90017-Z

[39] Anderson, W. K., and Bonhaus, D. L., "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids," *Computers and Fluids*, Vol. 23, No. 1, 1994, pp. 1–22. doi:10.1016/0045-7930(94)90023-X

[40] Anderson, W. K., Rausch, R. D., and Bonhaus, D. L., "Implicit/Multigrid Algorithm for Incompressible Turbulent Flows on Unstructured Grids," *Journal of Computational Physics*, Vol. 128, No. 2, 1996, pp. 391–408. doi:10.1006/jcph.1996.0219

[41] van Leer, B., "Flux Vector Splitting for the Euler Equations," *Lecture Notes in Physics*, Vol. 170, 1982, pp. 507–512. doi:10.1007/3-540-11948-5_66

[42] Nielsen, E. J., and Anderson, W. K., "Recent Improvements in Aerodynamic Design Optimization on Unstructured Meshes," *AIAA Journal*, Vol. 40, No. 6, 2002, pp. 1155–1163. doi:10.2514/2.1765; also AIAA Paper 2001-596, 2001.

[43] Nielsen, E. J., "Aerodynamic Design Sensitivities on an Unstructured Mesh Using the Navier–Stokes Equations and a Discrete Adjoint Formulation," Ph.D. Thesis, Virginia Polytechnic Inst. and State Univ., Blacksburg, VA, 1998.

[44] Giles, M., Duta, M., Müller, J.-D., and Pierce, N., "Algorithm Developments for Discrete Adjoint Methods," *AIAA Journal*, Vol. 41, No. 2, 2003, pp. 198–205; also AIAA Paper 2001-2596, 2001.

[45] Nielsen, E. J., Lu, J., Park, M. A., and Darmofal, D. L., "An Implicit, Exact Dual Adjoint Solution Method for Turbulent Flows on Unstructured Grids," *Computers and Fluids*, Vol. 33, No. 9, 2004, pp. 1131–1155; also AIAA Paper 2003-272, 2003.

[46] Barth, T. J., and Jespersen, D. C., "The Design and Application of Upwind Schemes on Unstructured Meshes," AIAA Paper 89-366, 1989.

[47] Barth, T. J., "A 3-D Upwind Euler Solver for Unstructured Meshes," AIAA Paper 91-1548, 1991.

[48] Mavriplis, D. J., "Revisiting the Least-Squares Procedure for Gradient Reconstruction on Unstructured Meshes," AIAA Paper 2003-3986, 2003.

[49] Elliott, J. K., "Aerodynamic Optimization Based on the Euler and Navier–Stokes Equations Using Unstructured Grids," Ph.D. Thesis, Massachusetts Inst. of Technology, Cambridge, MA, 1998.

[50] Nielsen, E. J., and Kleb, W. L., "Efficient Construction of Discrete Adjoint Operators on Unstructured Grids Using Complex Variables," *AIAA Journal*, Vol. 44, No. 4, 2006, pp. 827–836. doi:10.2514/1.15830; also AIAA Paper 2005-324, 2005.

[51] Venkatakrishnan, V., "Convergence to Steady State Solutions of the Euler Equations on Unstructured Grids with Limiters," *Journal of Computational Physics*, Vol. 118, No. 1, 1995, pp. 120–130. doi:10.1006/jcph.1995.1084; also AIAA Paper 93-880, 1993.

[52] Balasubramanian, R., and Newman, J. C., III, "Discrete Direct and Discrete Adjoint Sensitivity Analysis for Variable Mach Flows," *International Journal for Numerical Methods in Engineering*, Vol. 66, No. 2, 2006, pp. 297–318. doi:10.1002/nme.1558

[53] Berger, M. J., Aftosmis, M. J., and Murman, S. M., "Analysis of Slope Limiters on Irregular Grids," AIAA Paper 2005-0490, 2005.

[54] Aftosmis, M. J., Gaitonde, D., and Tavares, T. S., "Behavior of Linear Reconstruction Techniques on Unstructured Meshes," *AIAA Journal*, Vol. 33, No. 11, Nov. 1995, pp. 2038–2049. doi:10.2514/3.12945

[55] Aftosmis, M. J., Berger, M. J., and Melton, J. E., "Adaptation and Surface Modeling for Cartesian Mesh Methods," AIAA Paper 95-1725, 1995.

[56] Nemec, M., and Aftosmis, M. J., "Adjoint Error Estimation and Adaptive Refinement for Embedded-Boundary Cartesian Meshes," AIAA Paper 2007-4187, 2007.

[57] Ilinca, C., Zhang, X. D., Trépanier, J.-Y., and Camarero, R., "A Comparison of Three Error Estimation Techniques for Finite Volume Solutions of Compressible Flows," *Computer Methods in Applied Mechanics and Engineering*, Vol. 189, No. 4, 2000, pp. 1277–1294. doi:10.1016/S0045-7825(99)00377-1

[58] Haselbacher, A., "A WENO Reconstruction Algorithim for Unstructured Grids Based on Explicit Stencil Construction," AIAA Paper 2005-879, 2005.

[59] Thomas J. L., Diskin, B., and Rumsey, C. L., "Towards Verification of Unstructured-Grid Solvers," *AIAA Journal*, Vol. 46, No. 12, 2008, pp. 3070–3079. doi:10.2514/1.36655; also AIAA Paper 2008-666, 2008.

[60] Park, M. A., and Darmofal, D., "Parallel Anisotropic Tetrahedral Adaptation," AIAA Paper 2008-917, 2008.

[61] Carlson, H., Mack, R., and Morris, O., "A Wind-Tunnel Investigation of the Effect of Body Shape on Sonic-Boom Pressure Distributions," NASA TN D-3106, 1965.

[62] Mack, R. J., and Kuhn, N., "Determination of Extrapolation Distance with Measured Pressure Signatures from Two Low-Boom Models," NASA TM-2004-213264, 2004.

[63] Mack, R. J., and Kuhn, N., "Determination of Extrapolation Distance with Pressure Signatures Measured at Two to Twenty Span Lengths from Two Low-Boom Models," NASA/TM-2006-214524, 2006.

[64] Carter, M., and Deere, K., "Grid Sourcing and Adaptation Study Using Unstructured Grids for Supersonic Boom Prediction," AIAA Paper 2008-6595, 2008.

[65] Cliff, S. E., Baker, T. J., and Hicks, R. M., "Design and Computational/Experimental Analysis of Low Sonic Boom Configurations," *High-Speed Research: 1994 Sonic Boom Workshop: Configuration Design, Analysis, and Testing*, CP-1999-209699, edited by D. A. McCurdy, NASA, Dec. 1999, pp. 33–58.

[66] Cliff, S. E., "On the Design and Analysis of Low Sonic Boom Configurations," *High-Speed Research: Sonic Boom, Volume II*, CP-10133, edited by T. A. Edwards, NASA, Feb. 1994, pp. 37–78.

[67] Mendoza, J. P., Hicks, R. M., and Cliff, S. E., *Wind Tunnel Sonic Boom Tests on Four High-Speed Civil Transport Models at Mach Numbers from 1.68 to 2.4*, NASA TM-2008-214561, Sept. 2008.

W. Anderson
Associate Editor