INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids **31**: 175–187 (1999)

TETRAHEDRAL FINITE-VOLUME SOLUTIONS TO THE NAVIER–STOKES EQUATIONS ON COMPLEX CONFIGURATIONS[1]

N.T. FRINK* AND S.Z. PIRZADEH

Configuration Aerodynamics Br., Mail Stop 499, NASA Langley Research Center, Hampton, VA 23681-0001, USA

SUMMARY

A review of the algorithmic features and capabilities of the unstructured-grid flow solver USM3Dns is presented. This code, along with the tetrahedral grid generator, VGRIDns, is being extensively used throughout the USA for solving the Euler and Navier–Stokes equations on complex aerodynamic problems. Spatial discretization is accomplished by a tetrahedral cell-centered finite-volume formulation using Roe's upwind flux difference splitting. The fluxes are limited by either a Superbee or MinMod limiter. Solution reconstruction within the tetrahedral cells is accomplished with a simple, but novel, multidimensional analytical formula. Time is advanced by an implicit backward-Euler time-stepping scheme. Flow turbulence effects are modeled by the Spalart–Allmaras one-equation model, which is coupled with a wall function to reduce the number of cells in the near-wall region of the boundary layer. The issues of accuracy and robustness of USM3Dns Navier–Stokes capabilities are addressed for a flat-plate boundary layer, and a full F-16 aircraft with external stores at transonic speed.

KEY WORDS: finite-volume; Navier–Stokes; complex configurations

1. INTRODUCTION

This paper is offered in response to an invitation to present to the 'finite element' community[2] an alternate approach for solving the Navier–Stokes equations using a tetrahedral-based 'finite volume' formulation. The focus will be on the features and application of the cell-centered upwind flow solver USM3Dns [1–3] which is a component of the NASA Tetrahedral Unstructured Software System (TetrUSS[3]) [4].

The primary attractiveness of tetrahedral-based schemes hinges on a demonstrated capability for rapid grid generation on a wide variety of complex geometries by a broad range of users [5–7]. The VGRIDns code [7], another component of TetrUSS, is widely used in the USA to generate inviscid and viscous tetrahedral grids on geometries ranging from aircraft to heart pumps. As with inviscid tetrahedral grids in the past, viscous tetrahedral grids are generated on complex geometries by a range of users in a matter of days.

---

* Correspondence to: Configuration Aerodynamics Br., Mail Stop 499, NASA Langley Research Center, Hampton, VA 23681-0001, USA. Tel.: +1 757 864 2864; e-mail: n.t.frink@larc.nasa.gov

[1] This article is a US Government work and is in the public domain in the United States.

[2] Tenth International Conference on Finite Elements in Fluids, Tucson, Arizona, USA, January 5–8, 1998.

[3] URL: http://ad-www.larc.nasa.gov/tsab/tetruss

The finite-volume approach is based on the physical concept of using macroscopic control volumes to numerically solve the conservation laws of fluid motion. Jameson *et al.* [8] reported one of the earliest successful implementations of this approach for solving the Euler equations on tetrahedral grids. The finite element method [9] is more mathematically based and uses a weak variational form of the governing equations, along with polynomial shape functions, for discretization. While there are inherent differences in the two methodologies, there are similarities between the Galerkin finite element procedure with piece-wise-linear shape functions and the finite-volume approach [8].

Tetrahedral finite-volume Navier–Stokes methodology is maturing along two tracks: node-centered and cell-centered schemes, each with their relative merits. Node-centered schemes exploit an efficient edge-based data structure and more readily facilitate general polyhedral cell volumes [10–12], but typically require large tetrahedral grids. Cell-centered schemes exploit geometric features of tetrahedra for constructing accurate spatial reconstruction schemes and provide comparable accuracy with fewer tetrahedra[4], but they are not as easily extendable to general elements. While the underlying Navier–Stokes methodologies have advanced rapidly in recent years, they are still less mature than their more established 'structured' grid counterparts. Legitimate questions still remain regarding the solution accuracy of unstructured Navier–Stokes schemes [13,11], and the user community does not yet have a sufficient experience base from which to derive full confidence. Thus, there is a strong need for more fundamental analyses and systematic application studies which address the key issues of solution accuracy, robustness, and efficiency on a range of configuration and flow classes.

There is an ongoing discussion among the computational fluid dynamics community regarding the use of pure tetrahedra versus mixed elements for Navier–Stokes computations. While there has been no definitive resolution to this issue, the authors are generally pleased with the (cell-centered) tetrahedral approach. Grid generation is rapid and robust with VGRIDns. Grid sizes are manageable, and the number of tetrahedra needed to resolve the boundary layer is comparable to that for structured or prismatic grid methods. The primary short comings arise from the large memory requirements of general-indexed schemes using implicit time integration. Work is currently underway to resolve these problems using zonal decomposition techniques and adaptive gridding.

The scope of this paper is to review the underlying Navier–Stokes methodology of the VGRIDns grid generator and USM3Dns finite-volume flow solver, and to demonstrate this emerging capability on a very complex configuration. An assessment of solution accuracy is presented for the flat-plate boundary layer problem. The issues of accuracy and robustness are examined with a transonic computation on a realistic complex configuration using a full F-16 aircraft with external stores.

2. TETRAHEDRAL GRID GENERATOR, VGRIDns

VGRIDns is based on the advancing-front (AFM) [14] and the advancing-layers (ALM) [15] methods. The generation of a 'Navier–Stokes' grid is accomplished automatically in three main steps: (1) generation of triangular surface grid by AFM and ALM; (2) generation of thin-layered tetrahedral grid in the boundary layer by ALM; and (3) generation of regular (inviscid) tetrahedral grid outside the boundary layer by AFM.

---

[4] A given tetrahedral grid has between 5 and 5.5 more tetrahedra than nodes. Additional spatial resolution is achieved by the cell-centered scheme on a given grid by virtue of resolving the flow solution at > 5 times more spatial locations than the node-centered scheme.

Grid clustering is controlled by a prescribed stretching function within the 'viscous' layers, and a 'transparent' Cartesian background grid [16] overlaying the entire domain. Included in the background grid are a number of prescribed 'point' and 'line' sources for defining local cell spacings. The grid characteristics are smoothly diffused from the sources onto the background grid nodes by solving an elliptic equation. The problem is analogous to the transfer heat in a conducting medium.

Two main operations are involved in ALM: (1) computation of surface vectors along which the grid points are distributed and (2) construction of a pattern for a compatible cell connectivity within the thin layers. Thin layers of tetrahedra are formed by inserting new points along the surface vectors and connecting the points according to the predetermined connectivity pattern. The individual layers (see Figure 1) continue advancing independent of each other until either the background grid information or an approaching front warrants them to stop. When all layers are complete, the ALM process automatically switches to the AFM to generate regular tetrahedra outside the boundary layer. With a common background grid controlling both methods, the transition from thin layers to the regular grid becomes gradual and continuous.

Another feature of VGRIDns which is of practical importance for Navier–Stokes is its ability to generate multi-directional anisotropically stretched grids [17]. This results in at least a factor of three reduction in overall number of cells. With such a capability, fewer points are distributed in the directions of reduced flow gradient with no loss of grid resolution in other essential direction(s).

3. FINITE-VOLUME FLOW SOLVER, USM3Dns

The fluid motion is governed by the time-dependent Reynolds-averaged Navier–Stokes equations for an ideal gas which express the conservation of mass, momentum, and energy for a compressible Newtonian fluid in the absence of external forces. The equations are prescribed in integral form for a bounded domain Ω with the boundary $\partial\Omega$:

$$\frac{\partial}{\partial t}\iiint_{\Omega}\mathbf{Q}\,\mathrm{d}V+\iint_{\partial\Omega}\mathbf{F}(\mathbf{Q})\cdot\hat{n}\,\mathrm{d}S=\iint_{\partial\Omega}\mathbf{G}(\mathbf{Q})\cdot\hat{\mathbf{n}}\,\mathrm{d}S,\tag{1}$$

where the state variables are $\mathbf{Q}=[\rho,\ \rho u,\ \rho v,\ \rho w,\ e_{\mathrm{o}}]^{\mathrm{T}}$, and the inviscid and viscous fluxes, $\mathbf{F}(\mathbf{Q})$ and $\mathbf{G}(\mathbf{Q})$, respectively, are defined in [3].

A finite-volume discretization is applied to Equation (1) which results in a consistent approximation to the conservation laws. The spatial domain is divided into a finite number of

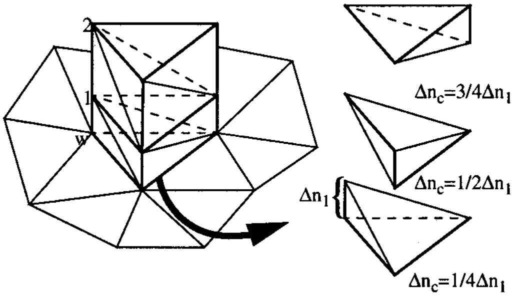

Figure 1. Thin-layered tetrahedra formed by ALM; (Δn_1, height of first node; Δn_c, height of cell centroid).

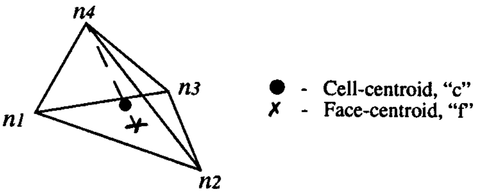

Figure 2. Reconstruction stencil for tetrahedral cell-centered scheme.

tetrahedral cell volumes. The discretized solution to Equation (1) results in a set of volume-averaged state variables which are in balance with the area-averaged fluxes (inviscid and viscous) across the cell faces.

3.1. Inviscid fluxes

The primary challenge in designing a tetrahedral cell-centered finite-volume scheme is the accurate reconstruction of cell-averaged data within non-isotropic tetrahedra to the perimeter triangular faces for flux computation as illustrated in Figure 2. A novel cell reconstruction process was derived in [2], which is based on an analytical formulation for computing the gradient term of a Taylor series expansion within tetrahedral cells. The scheme consists of simple, universal formula for tetrahedral cells

$$\mathbf{q}_{\mathrm{f}(1,2,3)} = \mathbf{q}_{\mathrm{c}} + \frac{1}{4}\left[\frac{1}{3}(\mathbf{q}_{n_1} + \mathbf{q}_{n_2} + \mathbf{q}_{n_3}) - \mathbf{q}_{n_4}\right], \tag{2}$$

where $\mathbf{q} = [\rho,\ u,\ v,\ w,\ p]^{\mathrm{T}}$ are the primitive variables of density, velocity, and pressure. Its derivation is based on exploiting several invariant features of tetrahedra in combination with the mid-point trapezoidal rule for numerical integration. This gives rise to the analytical expression of the Taylor series expansion of the cell-averaged solution to the cell face in Equation (2). With an accurate reconstruction of data at the cell faces, the inviscid flux quantities are computed in a straightforward manner across each face using the Roe [18] flux-difference splitting (FDS) approach. Limiting of the reconstructed solution is generally necessary for FDS to eliminate overshoots of the expansion in high-gradient regions, such as shocks. The well known Superbee or MinMod limiters have been implemented in USM3Dns.

A key component of the scheme is the secondary reconstruction of surrounding cell-averaged data to a common vertex or node by a weighted averaging procedure. The current averaging scheme, originally proposed in [1], is based on an inverse-distance weighting of the primitive variables from the cell centroid to the cell vertices:

$$q_n = \left(\sum_{i=1}^{N} \frac{q_{c,i}}{r_i}\right) \Bigg/ \left(\sum_{i=1}^{N} \frac{1}{r_i}\right), \tag{3}$$

where $r_i = [(x_{c,i} - x_n)^2 + (y_{c,i} - y_n)^2 + (z_{c,i} - z_n)^2]^{1/2}$. Reconstruction at boundary nodes is accomplished through ghost-cells as described in [2].

Until recently, the sole approach for secondary reconstruction was a pseudo-Laplacian averaging scheme presented in [2]. This scheme offers the advantage of second-order accuracy in reconstructing data from surrounding cells to a node. However, there is a need to artificially 'clip' the weighting factors between 0 and 2 [19] to avert a violation of the positivity principle which is necessary for solution stability. This artificial 'clipping' process does, unfortunately, compromise the formal second-order accuracy of the scheme to some extent. Recent experiences with applying the pseudo-Laplacian scheme to Navier–Stokes computations have surfaced some anomalous behavior which needs further investigation. Meanwhile, for the present work, we are temporarily reverting to the inverse-distance averaging of Equation (3) which is less accurate, but will never violate the principle of positivity.

3.2. *Viscous fluxes*

The viscous fluxes **G**(**Q**) are approximated at the cell-face centroids by linear reconstruction which provides a continuous representation of the solution variables across the cell faces. A modified version of the stencil presented by Mitchell [20], sketched in Figure 3, utilizes the averaged solution quantities at the three vertices of a cell face, $\mathbf{q}_{n_1}$, $\mathbf{q}_{n_2}$ and $\mathbf{q}_{n_3}$, and the cell-centered values of the two cells sharing the face, $\mathbf{q}_{c_1}$ and $\mathbf{q}_{c_2}$ where $\mathbf{q} \equiv (\rho, u, v, w, p)$. The required derivatives u_x, u_y, etc. for **G**(**Q**) are derived from a Cramer's rule solution to:

$$\begin{bmatrix} x_{c_2}-x_{c_1} & y_{c_2}-y_{c_1} & z_{c_2}-z_{c_1} \\ (0.5(x_{n_2}+x_{n_3})-x_{n_1}) & (0.5(y_{n_2}+y_{n_3})-y_{n_1}) & (0.5(z_{n_2}+z_{n_3})-z_{n_1}) \\ (0.5(x_{n_3}+x_{n_1})-x_{n_2}) & (0.5(y_{n_3}+y_{n_1})-y_{n_2}) & (0.5(z_{n_3}+z_{n_1})-z_{n_2}) \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$$

$$= \begin{bmatrix} u_{c_2}-u_{c_1} \\ (0.5(u_{n_2}+u_{n_3})-u_{n_1}) \\ (0.5(u_{n_3}+u_{n_1})-u_{n_2}) \end{bmatrix}. \tag{4}$$

3.3. *Time integration*

The computations are advanced to steady state by the implicit time advancement strategy of Anderson [21]. The scheme uses the linearized, backward Euler time differencing approach to update the solution at each time step. The linear system of equations are solved at each time step with a point-Jacobi subiteration on groups of tetrahedral cells which are separated into 'colors' (different from face-coloring) such that no two cells share a common face. While the point-Jacobi method is in itself not very efficient, convergence rate is accelerated by using the latest values of the updated solution variables as soon as they are available after subiteration of prior 'colors'. This produces a Gauss–Seidel-like effect, and the method has the advantage of being completely vectorizable. Typically, ten subiterations are used for Navier–Stokes computations, and 20 subiterations for Euler.

Because of the number of operations required to invert a matrix depends on the matrix bandwidth, the left-hand side of the system of linear equations is evaluated with first-order differencing to reduce both required storage and computer time. Convergence of the subiterations is further accelerated by using Van Leer's flux vector splitting (FVS) [22] on the left-hand side. Thus, in the present study, first-order differencing and FVS are applied to the left-hand side, and higher order differencing and FDS to the right-hand side. The viscous Jacobian terms are included in the left-hand side of the equation.

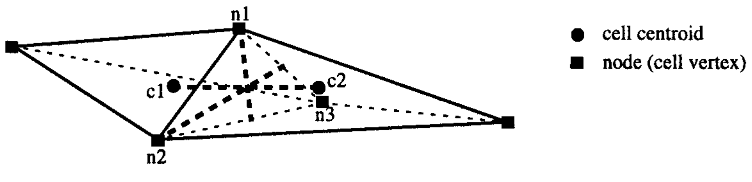

Figure 3. Modified tetrahedral viscous stencil.

USM3Dns requires 175 words/cell of core memory, and runs at a speed of 27 μs/cell/cycle on a CRAY C-90 with higher-order differencing. Multitasking on the CRAY is utilized for reducing the wall-clock time of a computation. Multitask efficiencies of six out of ten processors on a 16-processor machine are typically achieved for most applications.

3.4. Turbulence model

3.4.1. Spalart–Allmaras. Closure of the Reynolds stress is provided by the one-equation Spalart–Allmaras (S-A) turbulence model. This model is derived 'using empiricism and arguments of dimensional analysis, Galilean invariance, and selective dependence on the molecular viscosity' [23]. The model solves a partial differential equation (PDE) over the entire field for a transformed working variable, $\tilde{\nu}$, from which the eddy viscosity, μ_t, can be extracted. The PDE is solved separately from the flow equations using the same backward Euler time integration scheme, which results in a loosely coupled system. The production and destruction terms have been modified as recommended in [23] to ensure positive eddy viscosity throughout the computation.

On 'no-slip' surfaces, the dependent variable, $\tilde{\nu}$, is set to zero. For tangent-flow surfaces, a zero gradient of the variable is applied. Far field boundary conditions are applied by extrapolating $\tilde{\nu}$ from the interior for outflow boundaries, and taken from the free stream for the inflow.

The S-A model requires that the distance of each cell to the nearest wall be provided for the near-wall damping terms for cells which are in proximity to 'viscous' surfaces. These distances are determined prior to code execution for cells in the 'viscous' layers. A very efficient preprocessing code is developed which exploits the ordering of the nodes in these layers (see Figure 1) to complete the distance computation within a few seconds.

3.4.2. Wall function. The S-A model has been coupled with a wall function formulation to eliminate the need for resolving the flow in the sublayer portion of a turbulent boundary layer. Details are presented in [3]. With this approach, the inner region of the boundary layer is modeled by an analytical function which is matched with the numerical solution in the outer region. This has the advantage of (1) significantly reducing memory requirement by eliminating a large portion of cells normally required to resolve the sublayer, and (2) improving overall convergence by removing the thinner, more highly stretched cells which add stiffness to the solution process.

The selected wall function is a law-of-the-wall expression [24] derived by Spalding in 1961. The present implementation considers no adjustment to adiabatic wall density, which is important to high speed flows.

4. RESULTS AND DISCUSSION

Results are presented in this section for the flat-plate boundary layer problem, and a complete F-16 aircraft with external stores at transonic speeds. Only a cursory summary of the flat-plate problem is included from prior work [3] to review the accuracy of a relatively new wall-function approach. The F-16 computation was chosen to demonstrate the emerging unstructured finite-volume Navier–Stokes capability on a realistic complex aircraft configuration which exhibits strong viscous effects during store separation.

4.1. Grid spacing

The normal grid spacing across the boundary layer is prescribed by the exponential function:

$$\Delta n_j = \Delta n_1(1 + a(1 + b)^{j-1})^{j-1}, \tag{5}$$

such that the distance from the jth layer to the surface is given by $n_j = n_{j-1} + \Delta n_{j-1}$.

The parameter Δn_1 is the spacing of the first node above the surface (Figure 1), while a and b are parameters which govern the growth. A small utility program is used to determine an initial estimate of Δn_1, a, and b on an assumed 1/7th law velocity profile for prescribed values of Reynolds number, n^+, and number of points across the boundary layer.

4.2. Flat-plate boundary layer

The flat-plate boundary layer solution is used to assess the accuracy of the wall function in predicting the flat-plate turbulent skin friction. The computations were made on quasi-2D tetrahedral 'channel' grids for $M_\infty = 0.5$ and $Re_L = 2 \times 10^6$ where L is the length of the plate.

Grid 1 was generated by constructing a 49 × 12 H-topology structured grid with a normal spacing defined by $\Delta n_1 = 0.001L$, $a = 0.3$, and $b = 0.07$ in Equation (5), which yields roughly five nodes across the boundary layer at $x/L = 0.5$ and an approximate n^+ of 80 at the first node. The resulting upper domain boundary ($k = 12$) is located at $0.22L$. The 2D grid was stacked spanwise in $0.02L$ increments to form three planes resulting in a 3D structured dual-channel grid (49 × 3 × 12) of H-H topology. Each hexahedral cell was subdivided into two prismatic cells, which were further subdivided into three tetrahedra each to form the 3D tetrahedral grid with 6336 cells. The 'flat plate' was discretized by a cosine clustering between the 'structured' indices $15 \leq i \leq 49$ along the $k = 1$ boundary with inviscid flow prescribed on the $k = 1$ boundary ahead of the plate. Boundary conditions of constant entropy and constant total pressure were prescribed on the inflow plane, while an extrapolation condition was applied to the upper and exit domain boundaries. A constant freestream pressure was also imposed on the exit plane.

A second grid was generated in a similar manner as the first to explore the lower limits of grid coarseness on solution accuracy. Grid 2 was constructed from a 49 × 6 H-topology with the Equation (5) parameters of $\Delta n_1 = 0.001L$, $a = 2.0$, and $b = 0.07$. This resulted in a 3D channel grid (49 × 3 × 6) with 2880 cells, and an upper domain boundary ($k = 6$) also at $0.22L$.

Figure 4 portrays the effect of normal grid density on skin-friction coefficient and the law-of-the-wall behavior at $x/L = 0.5$, $Re_x = 1 \times 10^6$, for the two grids. The plotted nodal solutions were reconstructed from the surrounding tetrahedral cells using the weighted averaging procedure of [2]. *The plotted nodal data points effectively have three layers of tetrahedra between each pair, as sketched in Figure 1.* Note that the first nodal value is matched with the log layer at approximately $n^+ = 80$ for both grids. Grid 1 has five nodes (15 tetrahedra) across the boundary layer, while Grid 2 has three nodes (nine tetrahedra). Grid 1 displays excellent agreement over $0.2 < x/L \leq 1.0$ with the theoretical coefficient for fully turbulent flow, $C_f = 0.0583\ (Re_x)^{-1/5}$, which is based on the 1/7th power law assumption. Grid 2 does not exhibit the same level of agreement, but is remarkably close considering its extreme grid coarseness across the boundary layer.

These results highlight a common misconception about tetrahedral grid resolution. For a *cell-centered* scheme, it is important to think in terms of *cell*-resolution rather than *node*-resolution. It is the number of cells across the boundary layer that are important to solution accuracy for the present method.

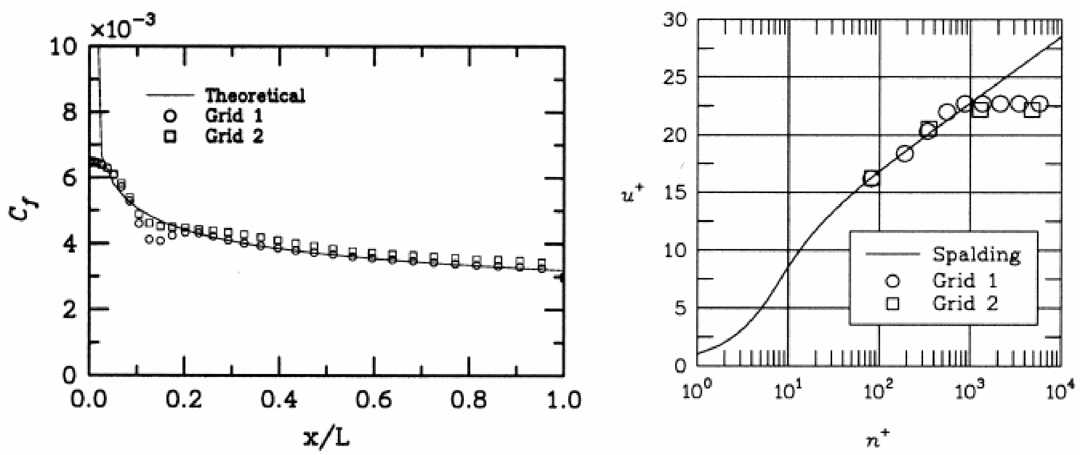

Figure 4. Effect of normal grid density on skin friction (left), and law-of-the-wall at $x/L = 0.5$ (right), for flat-plate boundary layer flow; Mach $= 0.5$ and $Re_L = 2$ million.

On a side note, the spurious behavior of the computed skin friction near the plate leading edge ($0 \leq x < 0.2$) in Figure 4 is not fully understood at this time and is presently under investigation. However, the principal interest for the present study is in the fully developed turbulent flow over the remaining region of the plate.

4.3. F-16 with generic finned store

Unstructured transonic Navier–Stokes computations are presented for the complete F-16 aircraft configuration as a demonstration and assessment of the present finite-volume methodology for a very complex geometry and flow field. In addition to the base aircraft, the configuration includes a flowing inlet, external fuel tank mounted on the inner wing pylon, and a generic finned-store suspended below the outer wing pylon. Support stings are also attached to the fuselage and the generic finned-store to simulate the wind-tunnel configuration tested in [25]. The horizontal and vertical tails were not present during the wind-tunnel test, but are included on the computational geometry as additional geometric complexities for demonstration purposes. Their influence on the store pressures is expected to be small.

An inviscid Euler study on this configuration is reported by Kern and Bruner [26] where the primary interest was the accurate and efficient prediction of initial loads on the finned-store to drive a separation trajectory analysis. The format of the following results is replicated from [26] to better complement the existing knowledge base.

4.3.1. Grid generation. Tetrahedral viscous and inviscid grids were generated for the F-16 aircraft using VGRIDns. The 'viscous' grid, shown in Figure 5, has 1428779 tetrahedra, 255959 nodes, whereas the 'inviscid' grid (not shown) contained 1111762 cells and 202378 nodes. The farfield boundaries were prescribed as 5.4, 6.2, and 3.6 body lengths ahead, aft, and spanwise of the aircraft, respectively. These boundary locations respectively correspond to 25, 29, and 17 wing mean-aerodynamic-chord (mac) lengths.

The F-16 surface definition is defined in full-scale inches. The normal grid spacing for the Navier–Stokes computation is sized for a wind-tunnel Reynolds number of 2 million, based on mac, assuming a 1/7th law boundary layer profile at the mid-mac location on the wing. The spacing is prescribed for computation with the wall function to yield 18 tetrahedral layers (six nodes) across the mid-chord boundary layer with an $n^+ = 30$ for the first node. The corresponding parameters for Equation (5) are $\Delta n_1 = 0.046$ inches, $a = 0.745$, and $b = 0.07$. The thin-layered tetrahedra are evident on the symmetry plane in Figure 5 and on a plane cutting through the wing, pylon, and store. While the grid was generated over an extended period of time, it is estimated that the cumulative manual labor was in the range of 40 to 60 h.

The 'inviscid' grid was generated from the same 'viscous' grid input file with an initial grid spacing of $\Delta n_1 = 0.2$ which produced a near identical match between the two surface triangulations. Inviscid grids are typically generated entirely by the AFM approach. However, to gain more control of the cells generated within the small gap between the finned-store and its pylon, it was necessary to conduct the inviscid grid generation with the ALM option using an appropriate initial spacing which yielded five layers of tetrahedra within the gap.

4.3.2. Flow solution. Navier–Stokes (N-S) and Euler flow solutions were obtained for angles of attack (AOA) of 0, 2, 4, and 8 degrees at Mach 0.95. The Navier–Stokes cases were run for 1000 iterations starting from freestream conditions with an initial Courant–Friedrichs–Lewy (CFL) number of 0.01 that was ramped up to 100 over 500 cycles. The Euler solutions were run for 800 cycles starting from freestream with an initial $\mathrm{CFL} = 10$ with ramping up to 50 over 50 cycles. Convergence was accelerated to steady state with local time stepping. The inviscid fluxes were limited with the Superbee limiter.

The USM3Dns solver was executed with multitasking over ten processors on a 16-processor CRAY C90. The Navier–Stokes and Euler solutions required 254 and 197 megawords of core memory, respectively. With a multitasking efficiency of six out of 10 processors, each N-S solution required from 10.5 to 11 h of CPU time which was turned around in two wallclock hours. One N-S solution required 14.8 CPU hours and 2.75 wallclock hours due to the particular scheduling load of the multitasking queue when the run was submitted. The Euler solutions each utilized approximately 6.6 CPU hours and 45 wallclock minutes of CRAY time, with a multitasking efficiency of 8.7 out of ten processors.

The history of normalized residual error and lift coefficient for the Navier–Stokes solutions in Figure 6 show a consistent convergence for each case. The leveling off of residual error is common for solutions on complex configurations at transonic speeds which also occurred for the Euler solutions. The oscillatory behavior above 500 iterations is most likely due to local

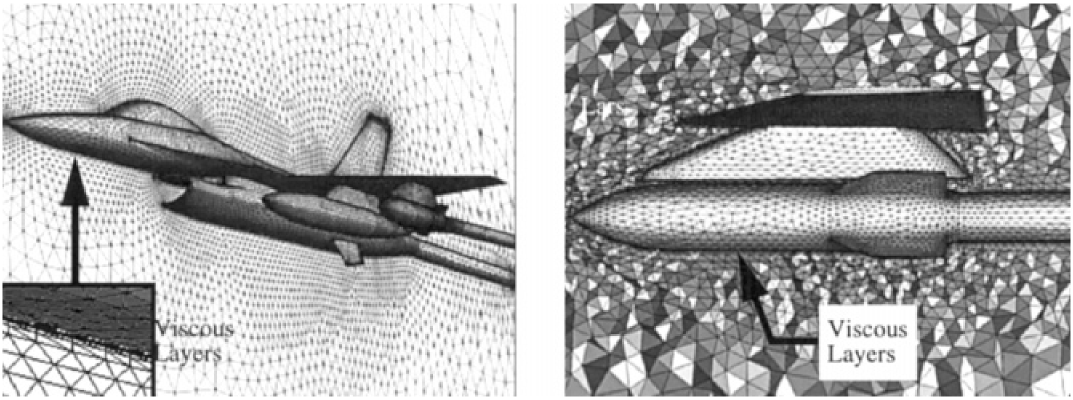

Figure 5. Viscous tetrahedral grid on complete F-16 aircraft with external stores, 1428779 cells. (Left, triangulation of surface and symmetry plane; right, tetrahedra on plane intersecting wing/pylon/finned-store.)

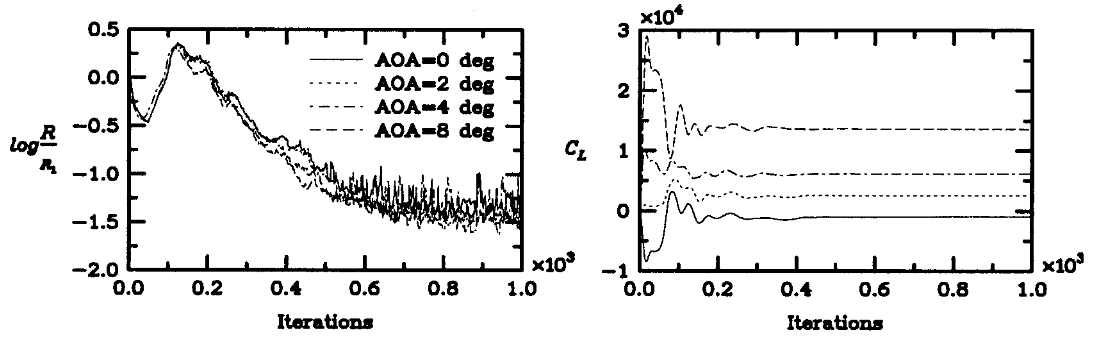

Figure 6. History of L2-norm residual error and lift coefficient of Navier–Stokes flow solutions.

recirculatory flow oscillations. Convergence is usually assessed from the force and moment coefficients.

Figure 7 depicts the surface pressure contours and store/pylon flow patterns for the N-S solution at $\mathrm{AOA} = 4$ degrees. The global pressure field is characterized by strong shock systems and interactions, both between store components and in the farfield. The surface flow patterns reveal shock-induced separations on the store afterbody and aft-pylon region.

4.3.3. Force and moments. Figure 8 presents a comparison of the experimental force and moments of the store with N-S and Euler results from USM3Dns. The plots depict the variation of store normal and side force coefficients (CN and CY, respectively) and the pitch (Cm) and yaw (Cn) moments, in the store body axis system, with angle of attack. Forces are positive upward or inboard from a cockpit perspective. Moments are positive for nose pitching upward or yawing inboard relative to the aircraft. The N-S results are generally in better agreement with the experimental data [25] than the Euler solutions, with the exception of the yawing moment, Cn. The largest benefit to the N-S approach occurs at AOA of 0 degrees. Reference [26] reported difficulties at this AOA with the Euler modeling. Very strong shock interactions were produced by the inviscid equations between the store's aft upper fins and the support pylon which may not be present in real flow. The present N-S solutions produce a more physically correct result by modeling the aft-store-body shock-induced flow separation, as will be shown in the next section.

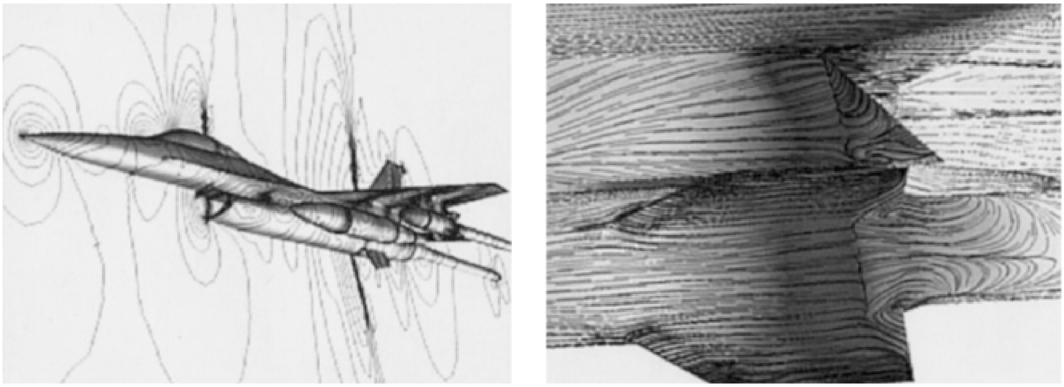

Figure 7. Navier–Stokes flow solution on complete F-16 aircraft with stores for Mach 0.95, angle of attack 4 degrees, and $Re_{\mathrm{mac}} = 2$ million. (Left, isobars of pressure; right, surface flow patterns on aft finned-store and pylon.)

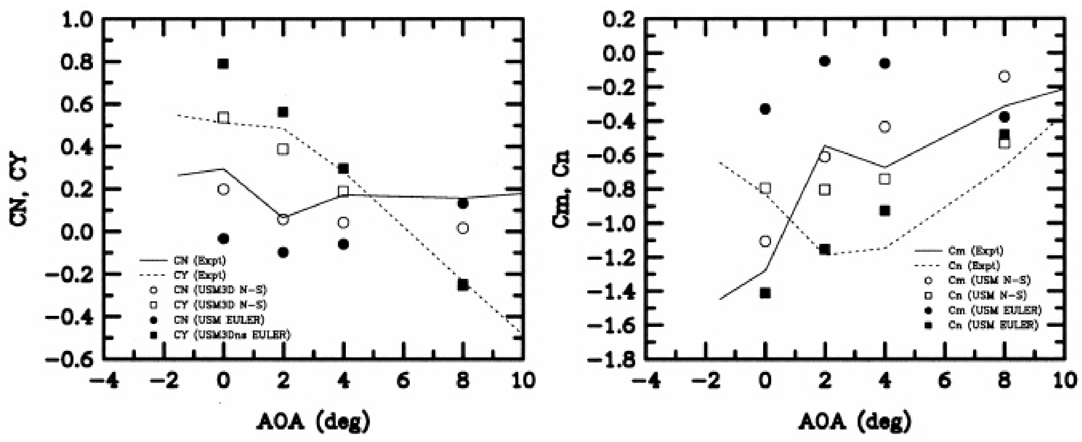

Figure 8. Variation of stores forces and moments with angle of attack, Mach = 0.95.

4.3.4. Surface pressures. The computed longitudinal distributions of surface pressure coefficient for the finned-store body are compared with experimental data from [25] in Figure 9 for 4 degrees AOA. The top, bottom, inboard, and outboard longitudinal stations correspond to radial locations of 5, 185, 275 and 95 degrees, respectively, measured from the top of the store body and rotating outward away from the fuselage. Values for x/L_{store} of 0.0 correspond to the store nose, and 1.0 to an aft-body station slightly behind the store fins. The correlation of pressure coefficient is generally good for the bottom, outboard, and inboard longitudinal stations. The poor correlation along the top station is caused by the interference of sway braces used in the wind-tunnel test, but not modeled in the computation, which induce a local suction at store body stations of 43 and 59%. Consistent with results from [26], the Euler solutions predict a rapid expansion followed by a strong shock at the boat-tail of the body. As expected from the N-S solution, the boat-tail separation, evident in the surface flows in Figure 7, results in a softened expansion in Figure 9 and brings the aft-shock into better agreement. There is also a generally better agreement of the mid-body compressions for the Navier–Stokes solutions.

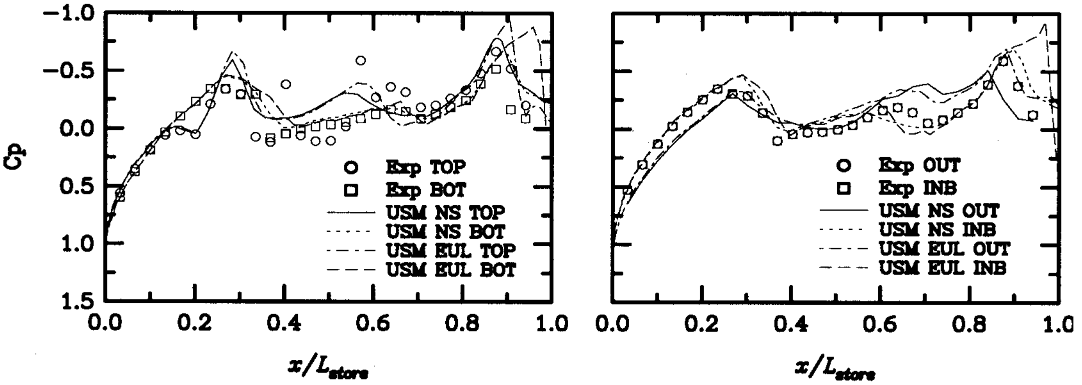

Figure 9. Longitudinal distribution of surface pressure coefficient on generic finned-store body at Mach 0.95 and angle of attack 4 degrees. (Left, top and bottom; right, outboard and inboard.)

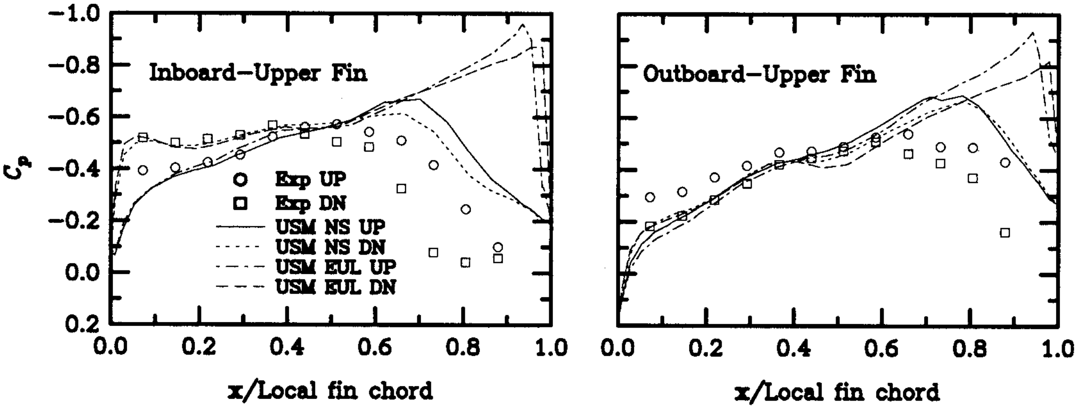

Figure 10. Longitudinal distribution of surface pressure coefficient on upper inboard and outboard fins on generic store at Mach 0.95 and angle of attack 4 degrees. (Left, inboard-upper fin; right, outboard-upper fin.)

A similar, but more dramatic effect can be observed in the fin pressure coefficient distributions in Figure 10. These longitudinal distributions are plotted along the 50% exposed-fin span station for the inboard-upper and outboard-upper fin components. The N-S methodology yields a significant improvement in predicting the shock strength and location on the aft 60% of the fin. A further assessment of the effect of grid refinement on the pressure distributions of Figures 9 and 10 is warranted in future work. However, the level of agreement of the present Navier–Stokes comparisons is good considering the relatively small number of tetrahedra (1428779 cells) used to resolve this complex full aircraft configuration.

5. CONCLUDING REMARKS

A review has been presented of the algorithmic features and capabilities of the tetrahedral-based finite-volume Euler and Navier–Stokes flow solver USM3Dns. This code along with the tetrahedral grid generator VGRIDns are primary components of the NASA TetrUSS package which is being used extensively throughout the USA for solving complex aerodynamic problems. The Navier–Stokes capability is maturing rapidly as a result of many focused customer applications on a range of configurations.

The present work has addressed the issues of accuracy and robustness of the USM3Dns Navier–Stokes feature using the flat-plate turbulent boundary layer problem and a full F-16 aircraft with external stores. The code achieves excellent accuracy in predicting the turbulent skin friction coefficient on the flat plate with as few as 15 tetrahedral layers (five nodes) across the boundary layer using the Spalart–Allmaras one-equation turbulence model with a wall function. The F-16 calculation serves as a good test case for demonstrating robustness and accuracy on a very complex geometry with transonic flow. A reasonably good prediction of force-moments and surface pressures on the generic finned-store body and fins was achieved with as few as 1.4 million tetrahedral cells. and within two wallclock hours per solution on a CRAY C90. While many more application studies are needed to increase user confidence in this capability these results serve to demonstrate the strong potential for tetrahedral-based finite-volume Navier–Stokes methodologies to become a practical computational aerodynamic tool.

ACKNOWLEDGMENTS

The authors would like to thank Mr. Steve Kern and Dr. Christopher Bruner of the Navel Air Systems Command for providing the surface definition and experimental data files for the F-16 configuration. The support of Mr. Javier Garriz, ViGYAN, Inc. and Mr. Derek Leitchy, Purdue University, in generating the F-16 viscous grid is also greatly appreciated. The first author would like to extend a special thanks to Dr. David Whitaker, Cray Research, for efficiently multitasking the USM3Dns code. All computations were performed on the Numerical Aerodynamic Simulation (NAS) facility at NASA Ames Research Center.

REFERENCES

1. N.T. Frink, 'Upwind scheme for solving the euler equations on unstructured tetrahedral meshes', *AIAA J.*, **1**, 70–77 (1992).
2. N.T. Frink, 'Recent progress toward a three-dimensional unstructured Navier–Stokes flow solver', *AIAA 94-0061*, January, 1994.
3. N.T. Frink, 'Assessment of an unstructured-grid method for predicting 3-D turbulent viscous flows', *AIAA 96-0292*, January, 1996.
4. N.T. Frink, S. Pirzadeh and P. Parikh, 'An unstructured-grid software system for solving complex aerodynamic problems', *NASA CP-3291*, May 9–11, 1995, pp. 289–308.
5. O. Hassan, K. Morgan, E.J. Probert and J. Peraire, 'Unstructured tetrahedral mesh generation for three dimensional viscous flows', *Int. J. Num. Methods Eng.*, **39**, 549–567 (1996) (University of Wales Swansea Report CR/846/94).
6. D.L. Marcum, 'Generation of unstructured grids for viscous flow applications', *AIAA-95-0212*, 33rd AIAA Aerospace Sciences Meeting, Reno, NV, January, 1995.
7. S. Pirzadeh, 'Three-dimensional unstructured viscous grids by the advancing front method', *AIAA J.*, **34**(1), 43–49 (1996).
8. A. Jameson, T.J. Baker and N.P. Weatherill, 'Calculation of inviscid transonic flow over a complete aircraft', *AIAA Paper 86-0103*, January, 1986.
9. J. Perair, J. Peiro and K. Morgan, 'Finite element multigrid solution of Euler flows past installed aero-engines', *Comp. Mech.*, **11**, 433–451 (1993).
10. W.K. Anderson, R.D. Rausch and D.L. Bonhaus, 'Implicit/multigrid algorithms for incompressible turbulent flows on unstructured grids', *J. Comp. Phys.*, **128**, 391–408 (1996).
11. T.J. Barth, 'Numerical aspects of computing viscous high Reynolds number flows on unstructured meshes', *AIAA Paper 91-0721*, January, 1991.
12. D.J. Mavriplis and V. Venkatakrishnan, 'A 3D agglomeration multigrid solver for the Reynolds-averaged Navier–Stokes equations on unstructured meshes', *NASA CR 195073*, April, 1995.
13. T.J. Baker, 'Irregular meshes and the propagation of solution errors', *Proceedings 15th International Conference on Numerical Methods in Fluid Dynamics*, Monterey, CA, June, 1996.
14. R. Lohner and P. Parikh, 'Three-dimensional grid generation by the advancing front method', *Int. J. Num. Methods Fluids*, **8**, 1135–1149, (1988).
15. S. Pirzadeh, 'Unstructured viscous grid generation by advancing-layers method', *AIAA J.*, **32**(8), 1735–1737 (1994).
16. S. Pirzadeh, 'Structured background grids for generation of unstructured grids by advancing-front method', *AIAA J.*, **31**(2), 257–265 (1993).
17. S. Pirzadeh, 'Progress toward a user-oriented unstructured viscous grid generator', *AIAA 96-0031*, January, 1996.
18. P.L. Roe, 'Characteristic based schemes for the Euler equations', *Annu. Rev. Fluid Mech.*, **18**, 337–365 (1986).
19. D.G. Holmes and S.D. Connell, 'Solution of the 2D Navier–Stokes equations on unstructured adaptive grids', *Presented at the AIAA 9th Computational Fluid Dynamics Conference*, June, 1989.
20. C.R. Mitchell, 'Improved reconstruction schemes for the Navier–Stokes equations on unstructured meshes', *AIAA Paper 94-0642*, January, 1994.
21. W.K. Anderson, 'Grid generation and flow solution method for euler equations on unstructured grids', *NASA TM-4295*, April, 1992.
22. B. Van Leer, 'Flux–vector splitting for the Euler equations', in E. Krause (ed.), *Eighth International Conference on Numerical Methods in Fluid Dynamics*, Vol. 170, Lecture Notes in Physics, Springer-Verlag, 1982, pp. 507–512.
23. P.R. Spalart and S.R. Allmaras, 'A one-equation turbulence model for aerodynamic flows', *AIAA Paper 92-0439*, January 1992.
24. F.M. White, *Viscous Fluid Flow*, McGraw-Hill, ISBN 0-07-069710-8, 1974.
25. J.C. Hoph, 'Separation characteristics of the DWS-24 dispenser, the MK-84 LDGP, the 370-GAL Tank (E) + Pylon, and the generic missile (metric and pressure-instrumented) in the flow field of the F-16 aircraft', *AEDC-TSR-94-1*, February 1994.
26. S. Kern and C. Bruner, 'External carriage analysis of a generic finned-store on the F-16 using USM3D', *AIAA 96-2456*, June, 1996.