MCAT Institute
Final Report 91-003

GRID GENERATION ABOUT COMPLEX THREE-DIMENSIONAL AIRCRAFT CONFIGURATIONS

GOETZ H. KLOPFER

April 1991

NCC2-616

MCAT Institute
3933 Blue Gum Drive
San Jose, California 95127

MCAT Institute
Final Report 91-003

GRID GENERATION ABOUT COMPLEX THREE-DIMENSIONAL AIRCRAFT CONFIGURATIONS

GOETZ H. KLOPFER

NCC2-616

April 1991

MCAT Institute
3933 Blue Gum Drive
San Jose, California 95127

Grid Generation About Complex Three-Dimensional Aircraft Configurations

The study of the aerodynamics or radar cross-section signatures about complete aircraft by numerical methods is still limited by two (or more) problems. The first involves the lack of adequate turbulence models applicable to a wide range of flow regimes and the second has to do with obtaining three dimensional grids with sufficient resolution to resolve all the flow or other physical features of interest. This study is concerned with the latter problem.

The generation of a computational grid involves a series of compromises to resolve several conflicting requirements. On one hand one would like the grid to be fine enough and not too skewed to reduce the numerical errors and to adequately resolve the pertinent physical features of the flow field about the aircraft. On the other hand the capabilities of present or even future supercomputers are finite and the number of mesh points must be limited to a reasonable number, one which is usually much less than desired for numerical accuracy.

One technique to overcome this limitation is the 'zonal' grid approach. In this method the overall field is subdivided into smaller zones or blocks in each of which an independent grid is generated with enough grid density to resolve the flow features in that zone. The zonal boundaries or interfaces require special boundary conditions such that the conservation properties of the governing equations are observed. Much work has been done in 3-D zonal approaches with nonconservative zonal interfaces. A 3-D zonal conservative interfacing method that is efficient and easy to implement was developed during the past year.

As originally envisioned, the conservative interfacing was to be developed for finite difference codes which were the most prevalent codes at NASA/Ames. However during the course of the work it became apparent that such a procedure would become rather complicated and that it would be much more feasible to do the conservative interfacing with cell-centered finite volume codes. Accordingly, the CNS code was converted to finite volume form. This new version of the code is named CNSFV. The original multi-zonal interfacing capability of the CNS code was enhanced by generalizing the procedure to allow for completely arbitrarily shaped zones with no mesh continuity between the zones. While this zoning capability works well for most flow situations it is, however, still nonconservative. The conservative interface algorithm has also been implemented, but has not been completely validated yet.

The results of this work has been reported in reference 1, a paper to be presented at the 10th AIAA CFD conference to be held 24-27 June 1991. This paper essentially covers all the details of the work done under this contract and is included as an appendix.

References

1. Klopfer, G. H. and Molvik, G. A.; "Conservative Multizonal Interface Algorithm for the 3-D Navier-Stokes Equations", AIAA Paper 91-1601, 24-27 June 1991, Honolulu, Hawaii.

Appendix

Preprint of AIAA Paper 91-1601, AIAA 10th Computational Fluid Dynamics Conf., 24-27 June 1991, Honolulu, Hawaii.

Conservative Multizonal Interface Algorithm for the 3-D Navier-Stokes Equations

G. H. Klopfer* and G. A. Molvik*
NASA Ames Research Center
Moffett Field, CA 94035

Abstract

One method of solving the Navier-Stokes equations about complex and realistic aerodynamic configurations is to use a zonal method. In this method the overall flow field domain is subdivided into smaller blocks or zones. In each of these zones, the flow field is solved separately of the other zones. The boundary data for each zone is provided by the neighboring zones. The major difficulty of the zonal methods has been how to maintain overall conservation for arbitrarily shaped zones. A new method of conservative patched zones has been developed. It uses structured meshes in the individual zones. The interface between the zonal block faces is defined by the union of the face points of adjoining blocks. An unstructured grid is generated upon which the interface fluxes can be determined. Flux balancing of the interface fluxes is then easily achieved to obtain global conservation. The method has been implemented into two Navier-Stokes codes. The use of the procedure is easily implemented into other finite volume codes. There are no topological restrictions on the zonal boundaries; e.g. the zonal interfaces can be curved surfaces for ease in constructing structured meshes in each of the zones. Several examples are presented to demonstrate the viability of the interfacing procedure.

Introduction

There are two basic approaches of numerically simulating the Navier-Stokes equations about complex and realistic aerodynamic configurations. One is based on structured meshes in which the neighbors of a mesh point are known implicitly. The other approach is based on an unstructured mesh in which the neighbors of a mesh point are not known implicitly and this information must be stored for each point. Numerical methods based on structured meshes are well developed, but suffer limitations when dealing with complex configurations in that it is difficult, if not impossible, to generate a single mesh about such a configuration and still have the required mesh qualities for stable and accurate numerical solutions. The generation of unstructured grids about complex configurations is, in principle, much easier; however, the numerical methods for such grids are not yet mature enough to compete against structured grid numerical methods.

The structured grid generation problem for complex configurations can be alleviated by zonal methods. In this method the overall domain is subdivided into smaller blocks or zones. In each of these zones, a grid is generated and the flow field is solved independently of the other zones. The boundary data for each zone is provided by the neighboring zones. The major difficulty of the zonal methods has been how to maintain overall conservation for arbitrarily shaped zones. The resolution of this difficulty is the purpose of this paper.

There are two types of zonal methods in common usage today. One is the overlaid zones in which both zones share a common interface region, as used in the Chimera [1] approach. The other is the patched zonal technique where the two zones share only a common boundary. Examples of this approach are the zonal method of Rai [2] and Thomas [3]. Both methods have advantages and disadvantages depending on the application. With patched grids it is much easier to maintain conservation since there is only one boundary across which the two zones communicate. On the other hand for moving multiple bodies the overlaid grid approach has certain advantages if conservation is not important, i.e., the flow field is continuous with no shock waves or shear surfaces. If conservation is important, as it is in this investigation, then we are restricted to patched zones.

Most methods using conservative patching methods are restricted to interface surfaces which are planar surfaces due to the minor gaps and overlaps that occur at a curved interface if the two zones are not mesh continuous. Furukawa et al [4] attempted to resolve this problem by using only one of the zones to determine the zonal boundary for both zones. The open question that remains is which zone determines the interface boundary. Furukawa et al chose to use the zone which has the better resolution. This can result in loss of accuracy if the mesh ratio changes in the interface, for example at a viscous boundary layer.

In this paper the zonal interface boundary is determined by the union of all the face mesh points of both adjoining zones. The interface surface is now unique and determined much more accurately than either one of the individual face surfaces. The collection of interface points is in general no longer structured and readily available unstructured grid generation techniques can be used to construct (triangulate) an interface grid. With the triangulated interface grid, the metrics (i.e., surface area normals) and cell volumes can be determined for each of the interface cells.

The development of the interface algorithm is discussed in the following sections. The procedure has been implemented into two finite volume three-dimensional Navier-Stokes codes, namely TUFF [5] and a finite volume version of CNS [6] henceforth called CNSFV. We will limit the exposition to the diagonal version of the Beam and Warming scheme [7] as used in the CNSFV code, but results from the TUFF code will also be presented.

---

* Research Scientist, MCAT Institute, San Jose, CA 95127

Numerical Scheme

Navier-Stokes Equations

The three-dimensional thin-layer Navier-Stokes equations in strong conservation law form in curvilinear coordinates are

$$\partial_\tau \hat{V} Q + \partial_\xi F + \partial_\eta G + \partial_\zeta H = Re^{-1} \partial_\zeta S \qquad (1)$$

where

$$Q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ e \end{pmatrix}, F = \begin{pmatrix} \rho U \\ \rho u U + \xi_x \hat{V} p \\ \rho v U + \xi_y \hat{V} p \\ \rho w U + \xi_z \hat{V} p \\ (e+p)U - \xi_t \hat{V} p \end{pmatrix},$$

$$G = \begin{pmatrix} \rho V \\ \rho u V + \eta_x \hat{V} p \\ \rho v V + \eta_y \hat{V} p \\ \rho w V + \eta_z \hat{V} p \\ (e+p)V - \eta_t \hat{V} p \end{pmatrix}, H = \begin{pmatrix} \rho W \\ \rho u W + \zeta_x \hat{V} p \\ \rho v W + \zeta_y \hat{V} p \\ \rho w W + \zeta_z \hat{V} p \\ (e+p)W - \zeta_t \hat{V} p \end{pmatrix} \qquad (2)$$

The contravariant velocity components are defined as

$$\begin{aligned} U &= \hat{V}(\xi_t + \xi_x u + \xi_y v + \xi_z w), \\ V &= \hat{V}(\eta_t + \eta_x u + \eta_y v + \eta_z w), \\ W &= \hat{V}(\zeta_t + \zeta_x u + \zeta_y v + \zeta_z w) \end{aligned} \qquad (3)$$

and the viscous flux is given by

$$S = \mu \hat{V} \begin{pmatrix} 0 \\ m_1 u_\zeta + m_2 \zeta_x / 3 \\ m_1 v_\zeta + m_2 \zeta_y / 3 \\ m_1 w_\zeta + m_2 \zeta_z / 3 \\ m_1 m_3 + m_2 (\zeta_x u + \zeta_y v + \zeta_z w)/3 \end{pmatrix} \qquad (4)$$

with

$$\begin{aligned} m_1 &= {\xi_x}^2 + {\xi_y}^2 + {\xi_z}^2, \\ m_2 &= \zeta_x u_\zeta + \zeta_y v_\zeta + \zeta_z w_\zeta, \\ m_3 &= (u^2 + v^2 + w^2)/2 + (Pr(\gamma - 1))^{-1} (a^2)_\zeta \end{aligned} \qquad (5)$$

The pressure is given by the equation of state

$$p = (\gamma - 1)(e - \rho(u^2 + v^2 + w^2)/2) \qquad (6)$$

Metric Terms

The metrics used above have a different meaning for a finite volume formulation compared to the finite difference formulation of [6]. Referring to a typical finite volume cell as shown in figure 1, the finite volume metrics are defined (see, for example, Vinokur [8]) as

$$\begin{aligned} \mathbf{s}_{j+\frac{1}{2}} &= s_{x,j+\frac{1}{2}} \mathbf{i} + s_{y,j+\frac{1}{2}} \mathbf{j} + s_{z,j+\frac{1}{2}} \mathbf{k} \\ &= \frac{1}{2}[(\mathbf{r}_7 - \mathbf{r}_4) \times (\mathbf{r}_8 - \mathbf{r}_4) + (\mathbf{r}_3 - \mathbf{r}_4) \times (\mathbf{r}_7 - \mathbf{r}_4)] \\ \mathbf{s}_{k+\frac{1}{2}} &= s_{x,k+\frac{1}{2}} \mathbf{i} + s_{y,k+\frac{1}{2}} \mathbf{j} + s_{z,k+\frac{1}{2}} \mathbf{k} \\ &= \frac{1}{2}[(\mathbf{r}_7 - \mathbf{r}_2) \times (\mathbf{r}_3 - \mathbf{r}_2) + (\mathbf{r}_6 - \mathbf{r}_2) \times (\mathbf{r}_7 - \mathbf{r}_2)] \\ \mathbf{s}_{l+\frac{1}{2}} &= s_{x,l+\frac{1}{2}} \mathbf{i} + s_{y,l+\frac{1}{2}} \mathbf{j} + s_{z,l+\frac{1}{2}} \mathbf{k} \\ &= \frac{1}{2}[(\mathbf{r}_7 - \mathbf{r}_5) \times (\mathbf{r}_6 - \mathbf{r}_5) + (\mathbf{r}_8 - \mathbf{r}_5) \times (\mathbf{r}_7 - \mathbf{r}_5)] \end{aligned} \qquad (7)$$

The finite volume metrics represent the cell face area normals in each of the curvilinear coordinates (ξ, η, ζ). They are related to the metrics introduced in equations (1 - 5) as follows

$$\begin{aligned} \xi_x \hat{V} &= s_{x,j+\frac{1}{2}} \\ \xi_y \hat{V} &= s_{y,j+\frac{1}{2}} \\ \xi_z \hat{V} &= s_{z,j+\frac{1}{2}} \\ \eta_x \hat{V} &= s_{x,k+\frac{1}{2}} \\ \eta_y \hat{V} &= s_{y,k+\frac{1}{2}} \\ \eta_z \hat{V} &= s_{z,k+\frac{1}{2}} \\ \zeta_x \hat{V} &= s_{x,l+\frac{1}{2}} \\ \zeta_y \hat{V} &= s_{y,l+\frac{1}{2}} \\ \zeta_z \hat{V} &= s_{z,l+\frac{1}{2}} \end{aligned} \qquad (8)$$

The volume of the computational cell is given by

$$\hat{V} = \frac{1}{6}[(\mathbf{r}_4 - \mathbf{r}_1) \times (\mathbf{r}_3 - \mathbf{r}_1) \cdot (\mathbf{r}_7 - \mathbf{r}_3)$$

$$+(\mathbf{r}_8 - \mathbf{r}_1) \times (\mathbf{r}_4 - \mathbf{r}_1) \cdot (\mathbf{r}_7 - \mathbf{r}_8)$$

$$+(\mathbf{r}_3 - \mathbf{r}_1) \times (\mathbf{r}_2 - \mathbf{r}_1) \cdot (\mathbf{r}_7 - \mathbf{r}_3)$$

$$+(\mathbf{r}_2 - \mathbf{r}_1) \times (\mathbf{r}_6 - \mathbf{r}_1) \cdot (\mathbf{r}_7 - \mathbf{r}_6)$$

$$+(\mathbf{r}_6 - \mathbf{r}_1) \times (\mathbf{r}_5 - \mathbf{r}_1) \cdot (\mathbf{r}_7 - \mathbf{r}_6)$$

$$+(\mathbf{r}_5 - \mathbf{r}_1) \times (\mathbf{r}_8 - \mathbf{r}_1) \cdot (\mathbf{r}_7 - \mathbf{r}_8)] \qquad (9)$$

and is the finite volume equivalent of the inverse Jacobian of the coordinate transformation in the finite difference formulation of [6].

Diagonal Beam-Warming Algorithm

The implicit Beam-Warming scheme for the finite volume formulation is given by

$$[I + \hat{V}^{-1}\hat{h}(\delta_\xi A^n + \delta_\eta B^n + \delta_\zeta C^n - Re^{-1}\delta_\zeta M^n)]\delta Q^{-1} = R^n \qquad (10)$$

where

$$R^n = -\hat{V}^{-1}\hat{h}[\delta_\xi F^n + \delta_\eta G^n + \delta_\zeta H^n - Re^{-1}\delta_\zeta S^n] \qquad (11)$$

The convective three-dimensional flux Jacobians A, B, C and the viscous flux Jacobian are defined in the appendix of [5]. With the use of approximate factorization and diagonalization of the flux Jacobian matrices, a scalar pentadiagonal algorithm [7] can be derived as

$$T_\xi[I + \hat{V}^{-1}\hat{h}\delta_\xi\Lambda_\xi]N[I + \hat{V}^{-1}\hat{h}\delta_\eta\Lambda_\eta]P\cdot$$

$$[I + \hat{V}^{-1}\hat{h}\delta_\zeta\Lambda_\zeta]T_\zeta{}^{-1}\Delta Q^n = R^n \qquad (12)$$

where δ_ξ is a central difference operator and $\Delta Q^n = Q^{n+1} - Q^n$ with $Q^{n+1} = Q(t^n + \hat{h})$. The viscous terms are not included in the left-hand (implicit) side. The artificial dissipation is included in both sides and is derived below.

The inviscid flux Jacobians are diagonalized as follows:

$$\partial_Q F = A = T_\xi\Lambda_\xi T_\xi{}^{-1}$$

$$\partial_Q G = B = T_\eta\Lambda_\eta T_\eta{}^{-1}$$

$$\partial_Q H = C = T_\zeta\Lambda_\zeta T_\zeta{}^{-1}$$

The T_ξ, T_η, T_ζ are the eigenvector matrices of A, B, C, respectively with $\Lambda_\xi, \Lambda_\eta, \Lambda_\zeta$ as the respective eigenvalues.

We also have

$$N = T_\xi{}^{-1}T_\eta$$

$$P = T_\eta{}^{-1}T_\zeta$$

Each of the factors of the implicit operator of equation (12) has an artificial term added to stabilize the central difference operator. The dissipation is based on Jameson's nonlinear second and fourth order dissipation and for the $\xi-$ operator takes the form

$$\hat{h}\hat{V}^{-1}\nabla_\xi[\bar{\sigma}_{j+\frac{1}{2}}(\epsilon^{(2)}\Delta_\xi\cdot - \epsilon^{(4)}\Delta_\xi\nabla_\xi\Delta_\xi\cdot)]\Delta Q^n \qquad (13)$$

with

$$\epsilon^{(2)} = \kappa_2 max(\gamma_{j+1}, \gamma_j, \gamma_{j-1}) \qquad (14)$$

$$\gamma_j = \frac{|p_{j+1} - 2p_j + p_{j-1}|}{|p_{j+1} + 2p_j + p_{j-1}|} \qquad (15)$$

$$\epsilon^{(4)} = max(0, \kappa_4 - \epsilon^{(2)}) \qquad (16)$$

where κ_2, κ_4 are constants of o(1), and Δ_ξ, ∇_ξ are the forward and backward difference operators. $\bar{\sigma}_{j+\frac{1}{2}}$ is a modified spectral radius defined as

$$\bar{\sigma}_{j+\frac{1}{2}} = \bar{\sigma}_j + \bar{\sigma}_{j+1}$$

$$\bar{\sigma}_j = \sigma_j(1 + \sqrt{max(\sigma_k/\sigma_j, \sigma_l/\sigma_j)}), \qquad (17)$$

$$\sigma_j = \left[|U| + a\sqrt{s_x{}^2 + s_y{}^2 + s_z{}^2}\right]_j \qquad (18)$$

and where cell centered surface areas are used, e.g.

$$s_{x,j} = \frac{1}{2}(s_{x,j+\frac{1}{2}} + s_{x,j-\frac{1}{2}})$$

The modified form of the spectral radius, equation (17), is suggested by Turkel [9] to account for large aspect

ratio computational cells as for example in a viscous layer. Similar dissipation terms are obtained for the $\eta-$ and $\zeta-$ operators. The dissipation terms added to the right hand side of equation (10) are identical to those given above except that ΔQ^n is replaced by Q^n.

To show that the above scheme is conservative and has the telescoping property, the right hand side of (12) can be written as

$$R^n = -\hat{h}\hat{V}^{-1}\left[\hat{F}^n_{j+\frac{1}{2}} - \hat{F}^n_{j-\frac{1}{2}} + \hat{G}^n_{k+\frac{1}{2}} - \hat{G}^n_{k-\frac{1}{2}}\right.$$

$$\left.+\hat{H}^n_{l+\frac{1}{2}} - \hat{H}^n_{l-\frac{1}{2}} - Re^{-1}\hat{S}^n_{l+\frac{1}{2}} + Re^{-1}\hat{S}^n_{l-\frac{1}{2}}\right] \qquad (19)$$

where $\hat{F}, \hat{G}, \hat{H}, \hat{S}$ are the numerical fluxes and are defined, for example for $\hat{F}$, as

$$\hat{F}^n_{j-\frac{1}{2}} = \bar{F}^n_{j+\frac{1}{2}} + \bar{\sigma}_{j+\frac{1}{2}}(\epsilon^{(2)}\,\Delta_\xi\, Q^n - \epsilon^{(4)}\,\Delta_\xi\,\nabla_\xi\,\Delta_\xi\, Q^n)_{j+\frac{1}{2}} \qquad (20)$$

with

$$\bar{F}^n_{j+\frac{1}{2}} = \frac{1}{2}\left[s_{x,j+\frac{1}{2}}(f_{j+1} + f_j)^n + s_{y,j+\frac{1}{2}}(g_{j+1} + g_j)^n\right.$$

$$\left.+s_{z,j+\frac{1}{2}}(h_{j+1} + h_j)^n\right] \qquad (21)$$

Similar terms are obtained for the numerical fluxes in the other two coordinate directions.

Boundary Conditions

To complete the equation set, boundary conditions must be specified. With the use of curvilinear coordinates, the physical boundaries have been mapped into computational boundaries, which simplifies the application of boundary conditions. The boundary conditions to be implemented for external viscous or inviscid flows include (1) inflow or far field, (2) outflow, (3) inviscid and (4) viscous impermeable wall, and (5) symmetry conditions. For external three-dimensional flow fields about closed bodies, the topology of the grid usually introduces (6) grid singularities which require special boundary conditions. The use of zonal methods can avoid the generation of grid singularities, but requires (7) special zonal interface boundary conditions. For compressible flows these zonal boundary conditions should be conservative to maintain global conservation.

In the finite volume approach, the specification of boundary conditions reduces to specifying the appropriate numerical fluxes at the boundaries. The details of implementing boundary conditions (1) through (5) are well known and are given in [5] and [10]. The grid singularity boundary condition is described below and the interface boundary conditions are given in the next section.

The grid singularity boundary condition is similar to the symmetry boundary condition for the inviscid and viscous fluxes in that there is no flux through that boundary. If, however, that is all that is done the results shown in figure 2a are obtained. These results are the density contours of a Mach 8 viscous blunt body flow. As shown in the figure, a nonphysical behavior appears at the singular line. The nonphysical results are due to a local violation of the entropy condition [11]. For central difference schemes, the artificial dissipation is the only stabilizing (entropy producing) mechanism available. At the grid singularity, the spectral radius, and hence the artificial dissipation, vanishes due to the vanishing of the metrics. The introduction of Harten's entropy correction [12] resolves the difficulty and the results are shown in figure 2b.

Interface Method

Most methods using conservative patching methods are restricted to interface surfaces which are planar due to the minor gaps and overlaps that occur at a curved interface if the two zones are not mesh continuous. This is demonstrated in figure 3a. Furukawa et al [4] attempted to resolve this problem by using only one of the zones to determine the zonal boundary for both zones. This is shown for the two-dimensional example in figure 3b. The open question that remains is which zone determines the interface boundary. Furukawa et al chose to use the zone with the better resolution. This, however, can result in loss of accuracy if the mesh ratio changes in the interface as for example at a viscous boundary layer.

In this paper the zonal interface boundary is determined by the union of all the face mesh points of both adjoining zones as shown in figure 3c for the two-dimensional case. The interface surface is now unique and determined much more accurately than by either one of the individual face surfaces. This presumes that the individual points from both zones lie in the interface surface and that the interface surface is itself smooth and continuous. The collection of interface points is, in general, no longer structured, and readily available unstructured grid generation techniques can be used to construct (triangulate) an interface grid. With the triangulated interface grid, the metrics (i.e., surface area normals) and cell volumes can be determined for each of the interface cells, i.e., the computational cells that touch the interface surface.

The redefinition of the interface surface has modified all the interface cells and they are no longer hexahedral, but rather multifacetted. Determining the surface area normals and cell volumes now becomes more complicated. There are three types of interface cells, namely, face cells, edge cells, and corner cells. The face cells have only one face bordering other zones, whereas the edge and corner cells have two or three faces in contact with other zones, respectively. A typical cell at the interface is shown in figure 4.

The modification introduced by the interface surface requires that the metrics and cell volumes of the interface cells be corrected to account for the changed shape of the interface cells. Corrections are required for the area normals of the cell face touching the interface surface, the four sidewalls, which may no longer be quadrilaterals, and the cell volume. For inviscid steady flows, the cell volume corrections are not needed since the volume has no effect on the steady solution as shown by equation (11). However,

for all viscous flows and inviscid unsteady flows the volume corrections are necessary.

There are two ways that the interface grids can be triangulated. The first involves eliminating all the grid lines from the structured grid cell faces and constructing the triangulated mesh. Since in general the new grid lines will not be aligned with the original grid lines, a clipping algorithm is used to clip those triangles which lie outside a particular individual face cell of either zone. An example of this triangulation procedure is shown in figure 5. Here two zones have square faces each consisting of a Cartesian grid of 15 × 15 uniformly spaced points. The two faces are oriented at an angle of 45° to each other (see figure 5a). The union of both sets of face points results in an unstructured collection of points which are then triangulated with an advancing front unstructured mesh generation procedure [13]. The resulting mesh is shown in figure 5b.

The second procedure retains the original grid lines and triangulates any set of points which form a polygon of more than three sides. This approach avoids the use of a clipping algorithm since each face cell contains an integral number of interface triangular cells. An example of this procedure is shown in figure 10b for two zones with polar grids.

The unstructured interface grid requires that a set of pointers be defined. These pointers indicate which two interface cells share a common interface area or section. These sections need not be triangular even though the surface has been triangulated. A common section (or polygon) may be composed of several triangles if the surface is triangulated by the second procedure described above or it may be composed of several clipped triangles if the first procedure is used. For a more efficient interface flux computation it is convenient to define another set of cross-reference pointers. These cross-reference pointers identify which interface polygons are in contact with each of the interface cells.

Numerical Interface Flux

The final step required for the conservative interface algorithm is to determine the numerical fluxes at the interface. Originally it was anticipated that the interface flux could be determined by any stable numerical scheme, not necessarily the one used in the interior of the zone. However, it was found that if the numerical schemes differ, then "glitches" always appear at the zonal interface. It was also necessary to maintain the same order of accuracy for the interface scheme as for the interior scheme. An example of this case is shown in figure 6. Here the interface flux is determined by a first order upwind scheme and the interior by a second order upwind scheme. The unfortunate results are the discrepancies at the interface boundary as shown by the pressure contours. The use of the same second order upwind scheme for the interface flux eliminates these discrepancies.

The above example shows that it is not possible to completely generalize a conservative interface algorithm. The numerical flux must be determined by the same numerical scheme as used in the interior of the zone. The geometric aspects of the interface algorithm can be generalized, which is essentially the most difficult part of the interface procedure. The numerical fluxes at the interface must (and should) be computed by the particular flow solver involved.

The computation of the interface numerical flux is relatively straightforward. Since it is scheme dependent it is given is general form first, valid for many schemes up to second-order accuracy. A specific form will be given for the CNSFV code. Figure 7 shows two zones, j1 and j2, and the interface with the individual surface polygons labelled by "i". The flux at the interface section "i" is given by

$$\vec{F}_{j1,j2=\frac{1}{2}} = \Phi[\vec{s}^{\,i}, \vec{q}_{j1=1}, \vec{q}_{j1=2}, \vec{q}_{j2=1}, \vec{q}_{j2=2}]$$

where Φ is the function defining the particular scheme under consideration, $\vec{s}^{\,i}$ is the area normal of the interface section "i", and the $\vec{q}_{j1}$ and $\vec{q}_{j2}$ are the cell-centered values in zones 1 and 2, respectively.

The numerical flux at the opposite face of the interface cell is slightly more complicated. It is

$$\vec{F}_{j2=\frac{3}{2}} = \Phi[\vec{s}_{j2=\frac{3}{2}}, \vec{q}_{j2=1}, \vec{q}_{j2=2}, \vec{q}_{j2=3}, \tilde{q}_{j1=1}]$$

where $\tilde{q}_{j1=1}$ is the area (i.e., $|\vec{s}^{\,i}|$) weighted average of all the interface cells of the zone "1" sections that comprise the face cell of zone "2" touching the interface surface. If higher order schemes are considered, then special care must also be taken for the fluxes at the next level of cell faces, ie. $\vec{F}_{j2=\frac{1}{2}}$. The above formulation is valid for all cell-centered finite volume schemes.

For the particular case of the CNSFV code, the interface numerical fluxes are

$$\hat{F}^n_{\frac{1}{2}} = \bar{F}^n_{\frac{1}{2}} + \bar{\sigma}_{\frac{1}{2}}(\epsilon^{(2)}\,\Delta_\xi\,Q^n - \epsilon^{(4)}\,\Delta_\xi\,\nabla_\xi\,\Delta_\xi\,Q^n)_{\frac{1}{2}}$$

where

$$\bar{F}^n_{\frac{1}{2}} = \frac{1}{2}\left[s^i{}_x(f_{j2=1} + f_{j1=1})^n + s^i{}_y(g_{j2=1} + g_{j1=1})^n\right.$$

$$\left. + s^i{}_z(h_{j2=1} + h_{j1=1})^n\right]$$

and

$$\bar{\sigma}_{\frac{1}{2}} = \bar{\sigma}_{j1=1} + \bar{\sigma}_{j2=1}$$

The flux at $j2 = \frac{3}{2}$ is similarily determined, taking care to use the appropriate area averaged values of all cell-centered values.

Results

To validate the interface algorithm, several tests were conducted. The first test is the freestream preserving test.

In this test, the inflow and and permeable wall boundary conditions (ie. conditions (1).(3) or (4) from the boundary conditions section) are set to the freestream conditions and the solution is converged. If the initial conditions were also set to the freestream condition, then the residual should be at machine zero (R= $O(10^{-14})$) and remain there for all subsequent iterations provided that the flow field is discretized properly with no gaps or overlaps in any of the computational cells and interface boundaries. Indeed, for the CNSFV code, this was the case.

However, the freestream test is not a good indication of the accuracy of the scheme. Because of the telescoping property of the scheme, the surface area normals can be computed inaccurately (even erroneously) and the scheme will still pass the freestream test. To test for accuracy, a single cell residual, equation (19), is computed with the same freestream conditions imposed as above. In this case, the maximum residual was of the $O(10^{-8})$ on a 64 bit machine (Cray YMP). If the grid and the metric terms are computed in double precision (ie. 128 bits), then the maximum residual reduces to $O(10^{-12})$. This indicates that machine roundoff errors are not yet a problem, but can be if the meshes are refined much further.

Three different flow cases covering the entire Mach regime from incompressible, to supersonic, and to hypersonic flows with finite rate chemistry were computed with the TUFF code. The same basic conservative interface algorithm described above was used in all three cases, however the conservation law equations differed for each of the three cases.

The first case involved an incompressible inviscid flow about a cylinder. The two zone mesh is shown in figure 8a. The pressure contours are given in figure 8b and the surface pressures in 8c. The results across the zonal boundaries are smooth and continuous.

The second set of results are for supersonic blunt body flow. This case is an inviscid Mach 2, axisymmetric blunt body flow computed on the four-zone mesh shown in figure 9a. Figure 9b shows the solution and the bow shock position. For these results, the analytic shock location at steady state [14] is shown by the solid squares. The solution is shown in terms of mach contours on a background grid which is cell centered for plotting purposes only. As can be seen the multizonal computed and analytic shock shapes compare quite well.

The final flow results obtained are for a viscous hypersonic flow about a hemisphere at $M_\infty = 15.3$ and $Re = 2.2 \times 10^5/m$. The interface triangulation for the two zones containing the grid polar singularity is depicted in figure 10b. The flow results in terms of Mach and atomic oxygen concentration contours are shown in figures 10c and d, respectively. Again, the results indicate that the solution contours are smooth and continuous across the zonal boundaries. Although not shown, the computed shock stand-off distance agreed well with the experimental data of reference 15.

Closing Remarks

A conservative zonal interface algorithm has been presented. It uses some of the best features of both the structured and unstructured mesh CFD technology. The interface surface grid is unstructured from which the metrics and interface fluxes can be readily constructed to obtain the proper conservative interface algorithm. For efficiency and rapid convergence, the flow solver within each of the zones is based on structured mesh CFD technology. The ordering between the zones can be unstructured for maximum flexibility in constructing zones and grids about complex and arbitrary configurations. The interface algorithm has been implemented into two three-dimensional Navier-Stokes finite volume codes (TUFF and CNSFV) and has shown to yield good results. Further testing is being conducted for more complex and realistic aerodynamic configurations. The procedure is general and can be easily implemented into other finite volume codes.

References

1. Benek, J.A., Buning, P.G., and Steger, J.L.; "A 3-D Chimera Grid Embedding Technique", AIAA Paper No. 85-1523CP, 1985.

2. Rai, M. M.; "A Conservative Treatment of Zonal Boundaries for Euler Equations Calculations", Journal of Computational Physics, Vol. 62, No. 2, Feb. 1986, pp. 472-503.

3. Walters, R.W., Reu,T., McGregory, W., Thomas, J. L., and Richardson, P.F.; "A Longitudinally-Patched Grid Approach with Applications to High Speed Flows", AIAA Paper No. 85-0295, 1985.

4. Furukawa, M., Yamasaki, M., and Inoue, M.; "A Zonal Approach for Solving the Navier-Stokes Equations Using a TVD Finite Volume Method", International Symposium on Computational Fluid Dynamics - Nagoya, Nagoya Trade & Industry Center, Nagoya, Japan, August 28-31, 1989.

5. Molvik, G. and Merkle, C. L.; "A Set of Strongly Coupled Algorithms for Computing Flows in Chemical Nonequilibrium", AIAA Paper 89-0199, Jan. 1989.

6. Ryan, J.S., Flores, J., and Chow, C.-Y.; "Development and Validation of a Navier-Stokes Code for Hypersonic External Flow", Journal of Spacecraft and Rockets, Vol. 27, No. 2, March-April 1990, pp. 160-166.

7. Pulliam, T.H. and Chaussee, D.S.; "A Diagonal Form of an Implicit Approximate Factorization Algorithm", Journal of Computational Physics, Vol. 39, 1981, p. 347.

8. Vinokur, M.; "An Analysis of Finite-Difference and Finite-Volume Formulations of Conservation Laws", NASA CR-177416, June 1986.

9. Turkel, E. and Vatsa, V.N.; "Effect of Artificial Viscosity on Three Dimensional Flow Solutions", AIAA Paper 90-1444, AIAA 21st Fluid Dynamics, Plasma Dynamics and Lasers Conference, Seattle, WA, June 1990.

10. Atwood, C. A.; "An Upwind Approach to Unsteady Flowfield Simulation", AIAA Paper 90-3100, AIAA 8th Applied Aerodynamics Conference, August 1990.

11. Merriam, M.L.; "An Entropy-Based Approach to Nonlinear Stability", NASA TM-101086, March 1989.

12. Harten, A.; "On a Class of High Resolution Total-Variation-Stable Finite-Difference Schemes", SIAM

J. Num. Anal., Vol. 21, pp. 1-23, 1984.

13. Merriam, M.L.; "A Fast Robust Algorithm For Delaunay Triangulation", Unpublished Paper, NASA Ames Research Center, Moffett Field, CA, October 1989.

14. Lyubimov, A.N. and Rusanov, V.V.; "Gas Flows Past Blunt Bodies", NASA TT-F 715. 1973.

15. Lobb, R. K.;"Experimental Measurements of Shock Distance on Spheres Fired in Air at Hypervelocities", The High Temperature Aspects of Hypersonic Flow, ed. W. C. Nelson, Pergammon Press, McMillan Co., New York, 1964.

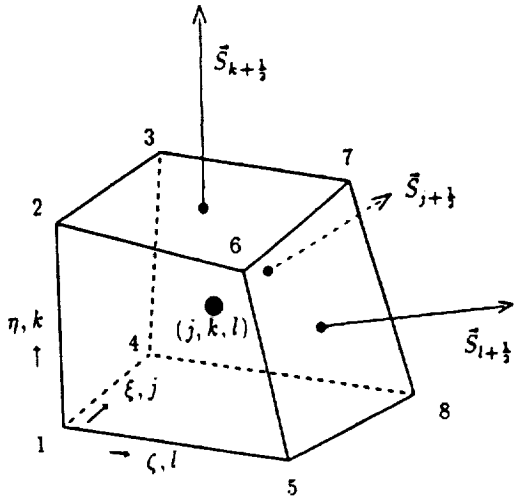
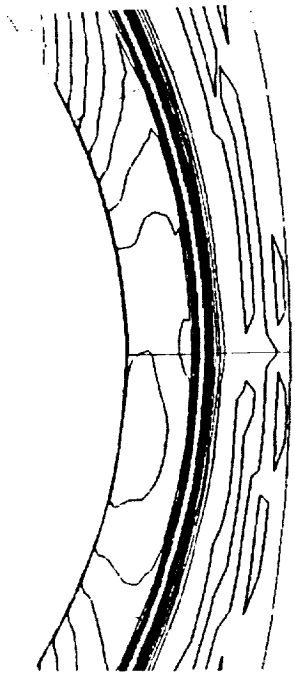
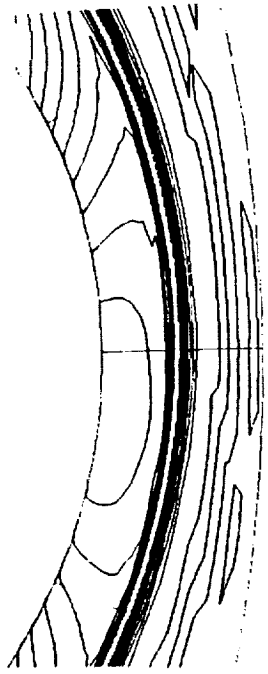


Figure 1. Finite volume cell nomenclature.

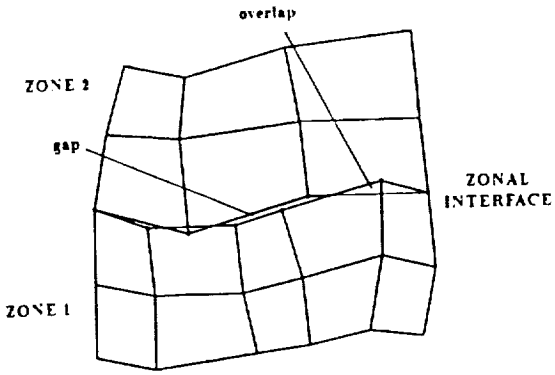


a) grid singularity problem.



b) grid singularity problem resolved by Harten's entropy correction [12].

Figure 2. Density contours for a viscous blunt body flow $M_\infty = 8.0, Re_D = 2.19 \times 10^5, \alpha = 5.0°$



a) "classical"
works only for planar interfaces,
otherwise conservation errors occur.

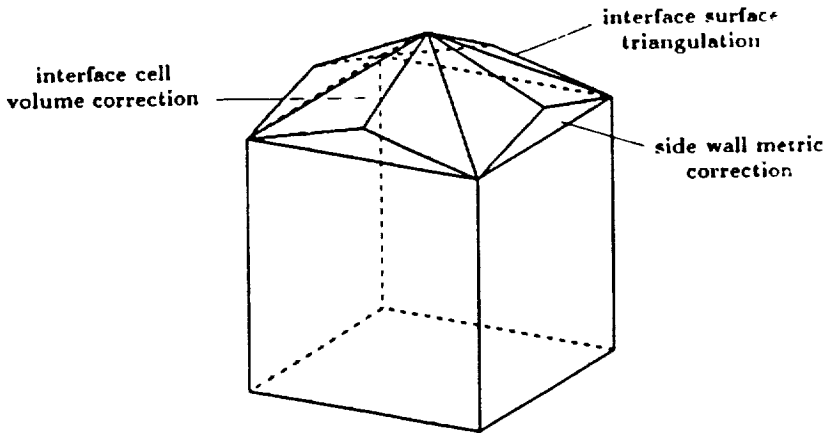
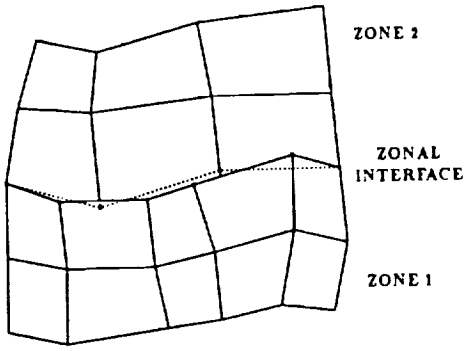


Figure 4. A typical zonal interface finite volume cell.



b) Furukawa et al [4] method
uses boundary of finer mesh
zone to determine the interface boundary.

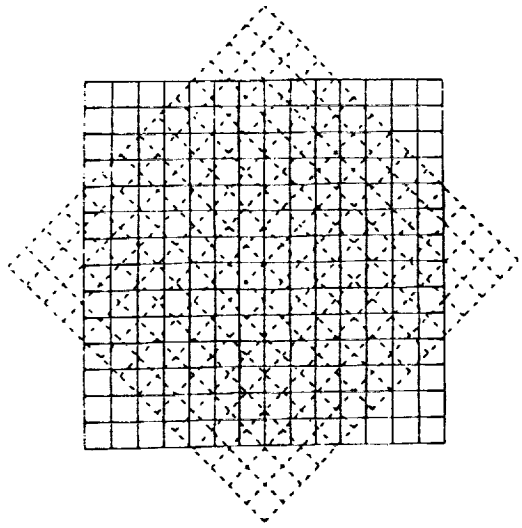
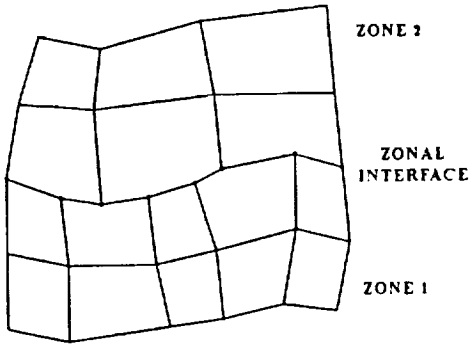


Figure 5a. The two structured face grids
of a typical interface surface.



c) present procedure
uses all the interface points
to define the interface boundary.

Figure 3. Zonal Boundary on a Curved Interface

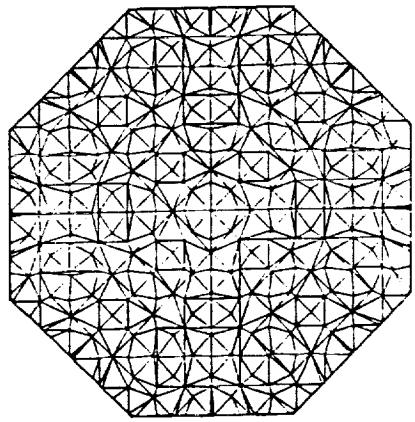


Figure 5b. The unstructured grid
of a typical interface surface.

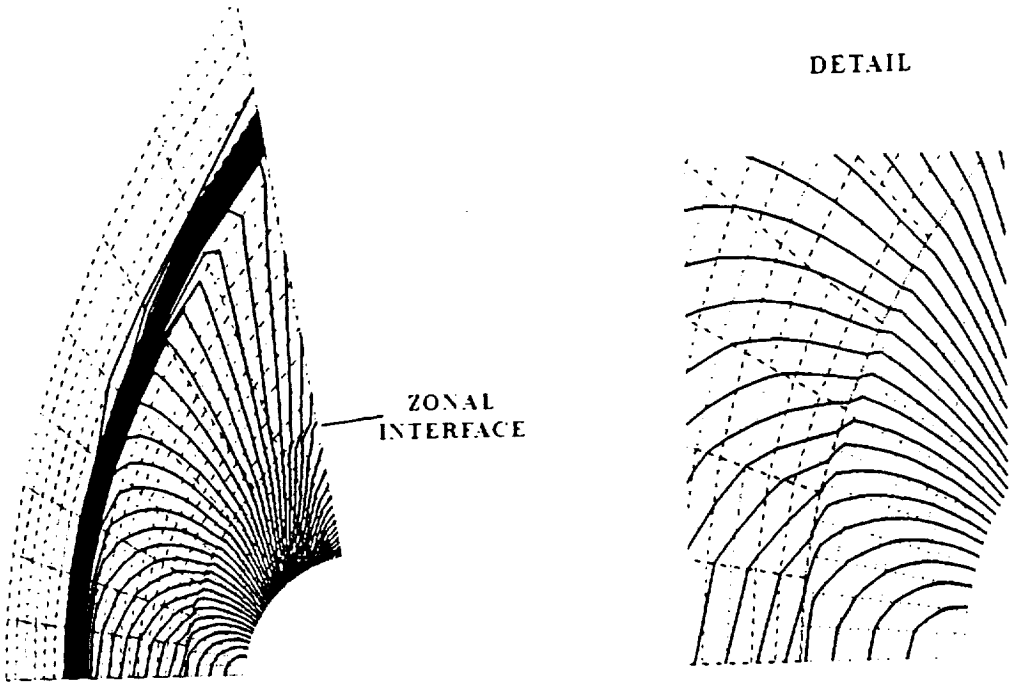


Figure 6. Pressure contours for inviscid blunt body flow at $M_\infty = 2.0$.

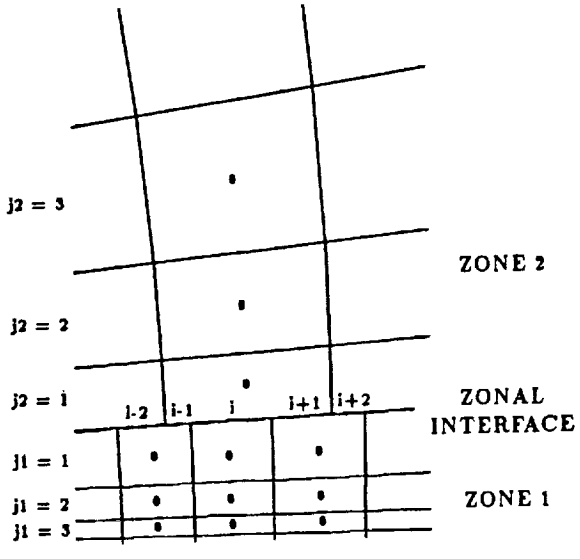
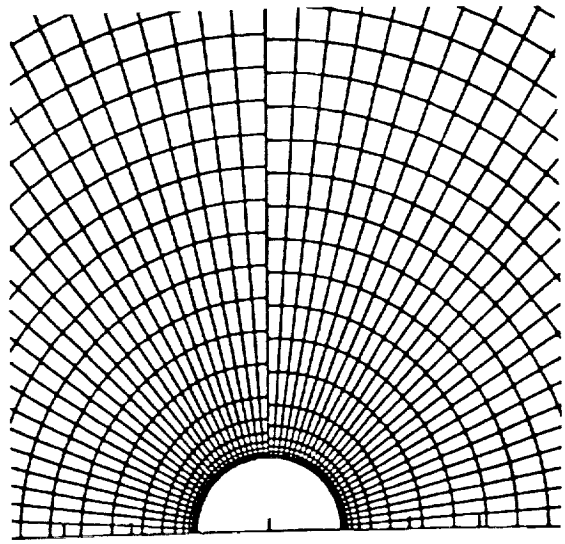
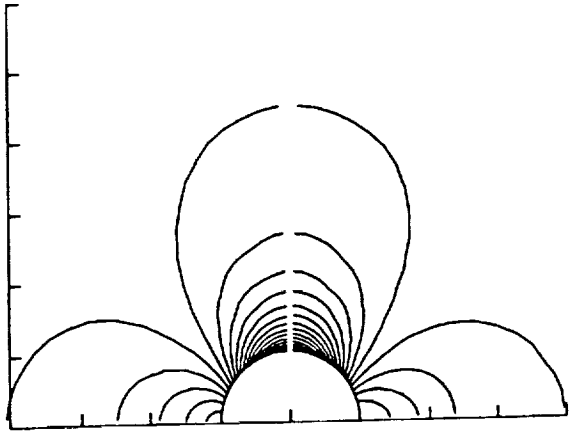


Figure 7. Mesh and nomenclature for numerical flux computation at zonal interface.

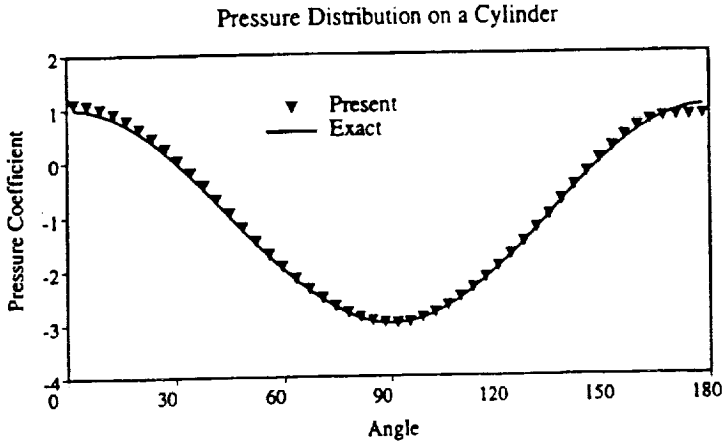


a) Two zone grid.

Figure 8. Inviscid incompressible flow about a cylinder.

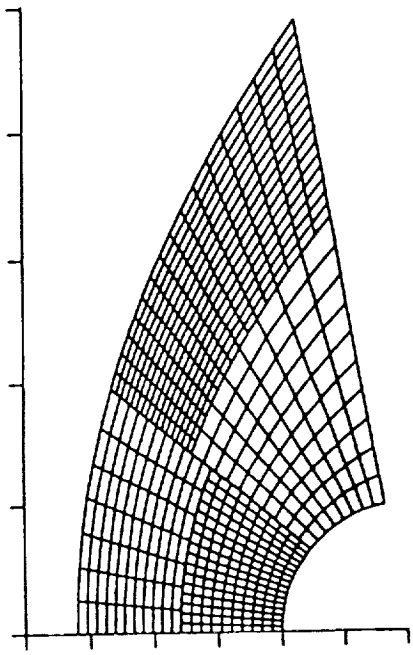


b) Pressure contours

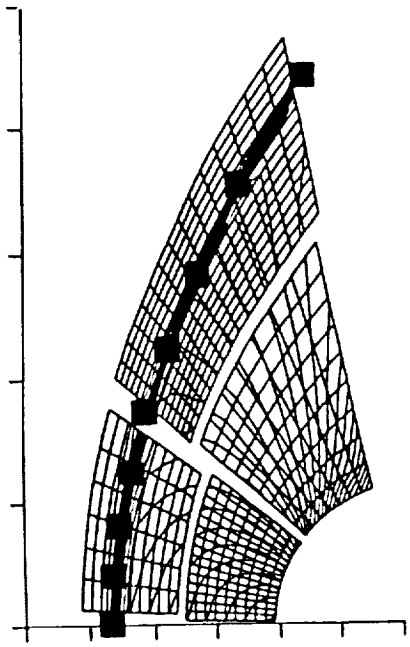


c) Comparison with analytic data.

Figure 8. Inviscid incompressible flow about a cylinder.

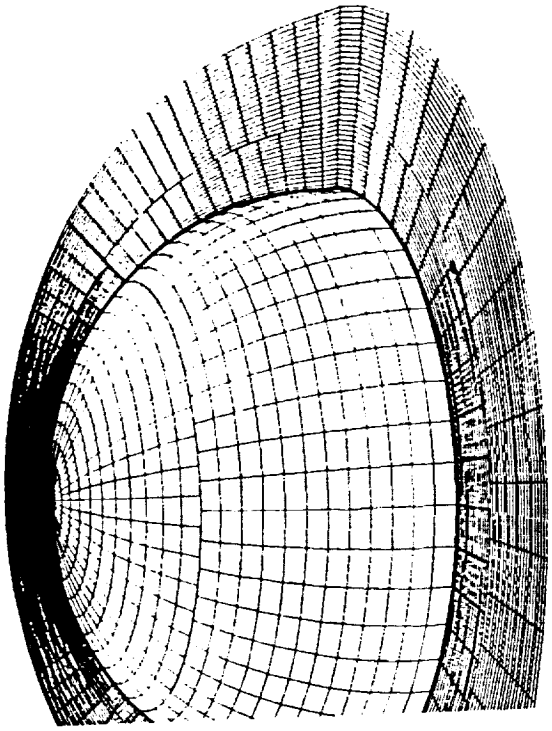


a) Four zone grid.

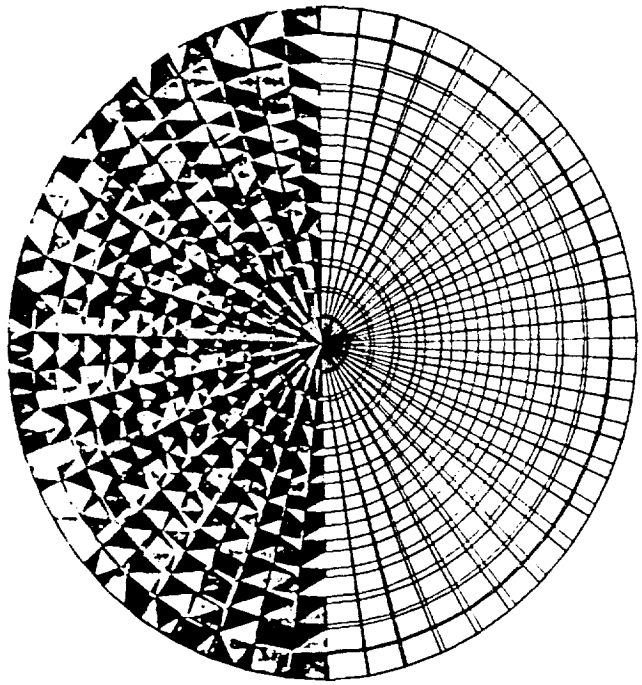


b) Mach contours.
■ - analytic bow shock location
(Lyubimov, NASA TT-F715, 1973).

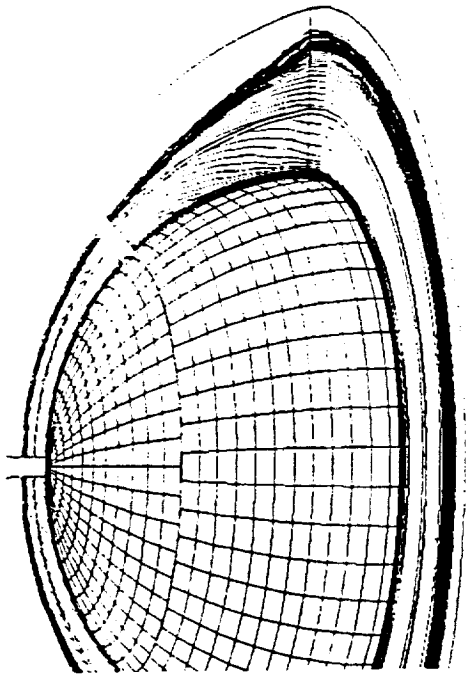
Figure 9. Axisymmetric inviscid blunt body flow at $M_\infty = 2.0$.



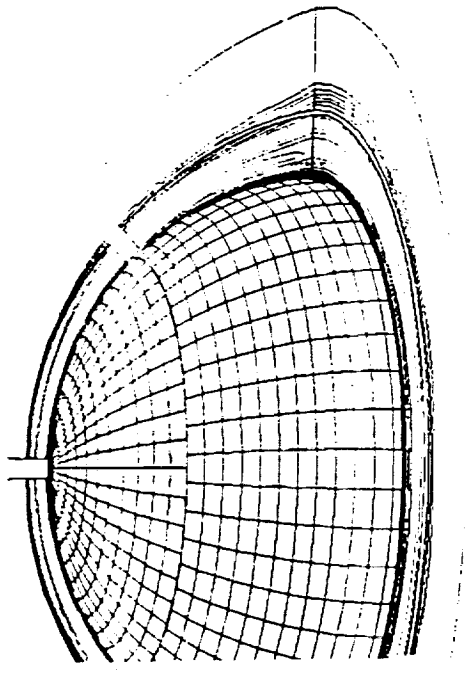
(a) Four Zone Grid



(b) The Two Face Grids and Triangulation



(c) Mach Contours



(d) Atomic Oxygen Concentration Contours

Figure 10. Viscous Hypersonic Blunt Body Flow with Finite Rate Chemistry $M_\infty = 15.3$, $Re = 2.2 \times 10^6/m$.