A VECTORIZED, FINITE-VOLUME, ADAPTIVE-GRID ALGORITHM FOR NAVIER-STOKES CALCULATIONS

PETER A. GNOFFO

Aero-Space Technologist, Aerothermodynamics Branch, Space Systems Division, NASA Langley Research Center, Hampton, Virginia 23665, Graduate Student, Department of Mechanical and Aerospace Engineering, Princeton University.

ABSTRACT

An adaptive grid, finite-volume method has been used to solve the Navier-Stokes equations for complete (forebody and afterbody) flowfields around blunt bodies. The code, which is applicable for axisymmetric or two-dimensional flows, allows the mesh to adjust during the computation to provide a closer spacing of mesh points in regions of high gradients, thus minimizing the number of required computational points. The solution technique is explicit, utilizing a maximum time-step advancement at each grid point to accelerate convergence to the steady state. The code has been fully vectorized for efficient solution on the CYBER 203 computer. A very flexible rezoning routine is used to concentrate mesh points anywhere in the field, either by a user-defined weighting function or by allowing high gradient regions to adjust the grid. The grid adjustment routine is implicit in nature and represents a very small portion of the total computational cost. Currently, the code runs in approximately $1.6 \times 10^{-6}$ seconds per grid point per iteration.

INTRODUCTION

The finite-volume method of numerically solving systems of conservation laws has been successfully applied to a wide variety of problems in fluid mechanics. Its ability to maintain conservation of mass, momentum, and energy from cell to cell, even in rather complex nonorthogonal coordinates, makes it particularly attractive for use with adaptive grid techniques. For the purpose of this paper, a finite-volume formulation (FVF) is defined as a discrete approximation to a conservation law written in integral form which (1) uniquely defines control volumes in such a way that control volumes (cells) do not overlap nor are gaps left in physical space and (2) uniquely defines fluxes and forces acting through cell walls so that summability without residue (conservation) is guaranteed. It differs from a finite difference formulation (PDF) only in the way a problem is approached. For example, given a system of conservation laws, we might consider the PDF as a discrete approximation to the
differential form of the laws while the FVF is a discrete approximation to the integral form of the laws. A review of various FVF’s suggests that all FVF’s can be written as FDF’s (and usually appear that way in the literature) but not all FDF’s can be written as FVF’s. Furthermore, discretizing the divergence form of the conservation laws in general coordinates to obtain the FDF does not guarantee that the formulation can be expressed as a FVF because conservation is not maintained due to some confusion of how to properly define the metric coefficients. Various ways of defining the metric coefficients to overcome this problem have been given in References 5 and 9. The discussion there leads one to conclude that while there may be a certain amount of ambiguity in how to properly define or discretize the metrics the problems can be overcome without a great deal of difficulty. It should also be noted that the FDF of a conservation law written in nondivergence form can in fact be a conservative FVF. An example of such a scheme is presented in Reference 10.

The discussion up to this point has been concerned with taking a differential form of a conservation law, approximating it by an FDF, and testing if it satisfies the requirements of an FVF (or forcing it to satisfy the requirements by suitably defining the metrics). The approach taken in the presentation which follows is to start with the integral form of the conservation law, approximate it with an FVF on some generalized grid, and expand the formulation to see what FDF results. It will be shown that no special treatment of metrics is required and in fact no terms which are readily recognizable as metric coefficients ever appear. It is only when the FVF is expanded into a format more familiar as an FDF do these metrics appear as ratios of the dimensions of cell walls to cell volumes. Furthermore, the unexpanded form of the FVF is ideally suited for vectorization.

Within this framework a grid adaptation routine has been developed which greatly facilitates placement of new points where needed. A description of how the adaptation algorithm has evolved from the groundwork established in references 11 and 12 to the current implicit adaption scheme will be presented. Finally, sample calculations of supersonic flow over a Viking Aeroshell, including flow in the wake, demonstrate the versatility of both the FVF and the adaption algorithm through its ability to concentrate points not only in the boundary layer but also in the free shear layer.

FINITE-VOLUME FORMULATION

Consider a system of two-dimensional conservation laws which can be expressed by the vector relation
\[ \frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0 \]  

where Equations (1a) and (1b) are the differential and integral forms of the conservation laws respectively, and $\mathbf{i}$ and $\mathbf{j}$ are unit vectors in the $x$ and $y$ directions, respectively.

A control volume in two-dimensional (or axisymmetric) space is determined by a quadrilateral whose vertices are defined by adjacent mesh points. The $[i,j]$ cell refers to the cell with vertices $(i,j), (i+1,j), (i,j+1), (i+1,j+1)$. Points of constant $j$ index define $\xi$ coordinate lines, increasing $\xi$ in the direction of increasing $i$. Points of constant $i$ index define $\eta$ coordinate lines, increasing $\eta$ in the direction of increasing $j$ (see Fig. 1). The flux through the wall defined by the points $(i,j), (i,j+1)$ is calculated using information from the $(i,j)$ vertex on the predictor step and from the $(i,j+1)$ vertex on the corrector step for odd time steps. The order is reversed for even time steps. The flux through the wall defined by the points $(i,j), (i+1,j)$ follows a similar pattern of definition, using information from the $(i,j)$ vertex on the predictor step. The FVF for the $[i,j]$ cell is written:

\[ \frac{\partial u}{\partial t} \begin{bmatrix} n+1 \\ i,j \end{bmatrix} A_{i,j} + F_{i+1,j}^n (y_{i+1,j+1} - y_{i+1,j}) - G_{i+1,j}^n (x_{i+1,j+1} - x_{i+1,j}) 
- F_{i,j}^n (y_{i,j+1} - y_{i,j}) + G_{i,j}^n (x_{i,j+1} - x_{i,j}) 
- F_{i,j+1}^n (y_{i+1,j+1} - y_{i,j+1}) + G_{i,j+1}^n (x_{i+1,j+1} - x_{i,j+1}) 
+ F_{i,j}^n (y_{i+1,j} - y_{i,j}) - G_{i,j}^n (x_{i+1,j} - x_{i,j}) = 0 \]  

(2a)
\[
\frac{\partial u^{n+1}_{i,j}}{\partial t} = \frac{u^{n+1}_{i+1,j+1} - u^{n+1}_{i-1,j-1}}{4\Delta t} + A_{i,j} \left( y_{i+1,j+1} - y_{i,j} \right) + c_{i,j+1}(x_{i+1,j+1} - x_{i,j})
\]

where \( A_{i,j} \) is the area of the \([i,j]\) cell. Equation (2) must approximate the integral form of Equation (1). The first term of Equation (2) must approximate the integral of \( \partial u/\partial t \) over the entire cell. In the FVF we get

\[
\frac{\partial u^{n+1}_{i,j}}{\partial t} = \frac{u^{n+1}_{i+1,j+1} - u^{n+1}_{i,j}}{4\Delta t} + A_{i,j} \left( y_{i+1,j+1} - y_{i,j} \right) + c_{i,j+1}(x_{i+1,j+1} - x_{i,j})
\]

Note the use of \( \Delta t \) which varies from point to point in Equation (4).

Now consider a finite-difference formulation, FDF, of Equation (1) which can be transformed to

\[
\frac{\partial u}{\partial t} + \frac{\partial f(x,y)}{\partial x} + \frac{\partial g(x,y)}{\partial y} + \frac{\partial h(x,y)}{\partial z} + \frac{\partial j(x,y)}{\partial z} = 0
\]

where \( x = x(\xi,\eta), \ y = y(\xi,\eta). \) The FDF of (5) is written:
\( \bar{u}^{n+1}_{i,j} = \bar{u}^n_{i,j} - \Delta t_{i,j} \left[ \frac{(F_{i+1,j} - F_{i,j})^n}{\Delta \xi} \xi^+_{x_{i,j}} + \frac{(G_{i+1,j} - G_{i,j})^n}{\Delta \xi} \xi^+_{y_{i,j}} + \frac{(F_{i,j} - F_{i-1,j})^n}{\Delta n} \eta^+_{x_{i,j}} + \frac{(G_{i,j} - G_{i-1,j})^n}{\Delta n} \eta^+_{y_{i,j}} \right] \) (6a)

\( \bar{v}^{n+1}_{i,j} = \bar{v}^n_{i,j} - \Delta t_{i,j} \left[ \frac{(F_{i,j} - F_{i,j-1})^{n+1}}{\Delta \xi} \xi^-_{x_{i,j}} + \frac{(F_{i,j} - F_{i,j-1})^{n+1}}{\Delta \xi} \xi^-_{y_{i,j}} + \frac{(G_{i,j} - G_{i,j-1})^{n+1}}{\Delta n} \eta^-_{x_{i,j}} + \frac{(G_{i,j} - G_{i,j-1})^{n+1}}{\Delta n} \eta^-_{y_{i,j}} \right] \) (6b)

\( \bar{u}^{n+1}_{i,j} = \frac{1}{2} \left( \bar{u}^n_{i,j} + \bar{u}^{n+1}_{i,j} \right) \) (6c)

The FVF (Eqs. (2) and (4)) and the FDF (Eq. (6)) are equivalent if

\( \xi^+_{x_{i,j}} \equiv \frac{y_{i+1,j+1} - y_{i+1,j}}{\Delta \xi} \Delta n / A_{i,j} \) (7a)

\( \eta^+_{x_{i,j}} \equiv \frac{y_{i,j} - y_{i+1,j+1}}{\Delta n} \Delta \xi / A_{i,j} \) (7b)

\( \xi^+_{y_{i,j}} \equiv \frac{x_{i+1,j+1} - x_{i+1,j}}{\Delta \xi} \Delta n / A_{i,j} \) (7c)

\( \eta^+_{y_{i,j}} \equiv \frac{x_{i,j} - x_{i+1,j+1}}{\Delta n} \Delta \xi / A_{i,j} \) (7d)

\( \eta^-_{x_{i,j+1}} \equiv \frac{x_{i,j+1} - x_{i,j}}{\Delta \xi} \Delta n / A_{i,j+1} \) (7e)

\( \eta^-_{y_{i,j}} \equiv \frac{y_{i,j} - y_{i,j+1}}{\Delta n} \Delta \xi / A_{i,j+1} \) (7f)
A linearized analysis of the FOF (Eq. (6)), after it has been expanded into a one-step scheme shows the following: (1) Equation (6) is a consistent representation of Equation (5), (2) the truncation error for variable \( \Delta t_{ij} \) is first order in time and can be expressed

\[
E_1 = \frac{-\Delta t_{ij}}{2} \left( \frac{\partial h_{ij}}{\partial \xi} C + \frac{\partial h_{ij}}{\partial \eta} D \right) \left( C \frac{\partial U}{\partial \xi} + D \frac{\partial U}{\partial \eta} \right)
\]

where

\[
C = \frac{\partial F}{\partial U} \xi_x + \frac{\partial G}{\partial U} \xi_y, \quad D = \frac{\partial F}{\partial U} \eta_x + \frac{\partial G}{\partial U} \eta_y
\]

and

\[
h_{ij}(\xi, \eta) = \Delta t(\xi, \eta)/\Delta t(\xi_{ij}, \eta_{ij})
\]

Note that \( E_1 = 0 \) if \( \Delta t_{ij} = \text{constant} \), and (3) the spatial truncation error is of \( O(\Delta^2) \) where \( \Delta \) is proportional to a cell wall dimension if all \( \xi_x \), etc., are first-order-accurate approximations to \( \xi_x \), etc., and \( (\xi_x + \xi_y)/2 \), etc., are second-order-accurate approximations to \( \xi_x \), etc. These conditions are formally satisfied by Equation (7). For example

\[
(\xi_x + \xi_y)/2 = \xi_x + \Delta \xi/\Delta \xi \left[ (Jy\xi\eta)\xi + Jy\xi\xi\eta \right] + \Delta \eta(Jy\xi\eta)/2
\]

where \( J = \frac{x\gamma\eta - x\gamma_y}{x_x} \cdot \)

Consequently, it is seen that while the FDF or FVT is second-order-accurate (for \( \Delta t_{ij} = \text{constant} \)) excessive stretching or skewness of the grid can introduce large factors of second-order terms. Furthermore, these factors multiply terms like \( \partial^2 U/\partial \xi^2 \), etc., and they contribute to the overall artificial viscosity of the method. At present, grid-induced errors are
checked by comparing results obtained for the same problem on the different
grids constructed from more points or from different adjusting parameters (see
section on ADAPTIVE GRID). It should also be noted that while Equation (5) is
not in strict conservation form, the FVF which models Equation (1b) is con-
servative in the strict sense of summation without residus. For example,
regardless of the grid used, the FVF returns an exact uniform flow if a uniform
flow is used as an initial condition.

It is noted that the equations and analysis in this section are for two-
dimensional flow. An axisymmetric extension introduces only minor complica-
tions. Finally we point out that the FVF (Eqs. (2) and (4)) reduces to
MacCormack's method\(^7\) in Cartesian coordinates. There are an infinity of other
ways to define cells and flux through cells on a general grid, each of which
lead to a unique FVF and unique definitions of metric coefficients for that
particular FVF.

BOUNDARY CONDITIONS

Only a cursory description of boundary conditions can be provided due to
space limitations.

Wall: \( u = v = 0 \),
Adiabatic wall or constant wall temperaturas. \( \partial p/\partial n = 0 \).

Shock: Rankine-Hugoniot relations for a discrete moving shock are
applied. Pressures behind shock obtained from extrapolation
from interior points.

Symmetry: Limiting form of differential form of governing equations
solved using MacCormack's method.

Outflow: Primitive variables obtained by extrapolation.

ADAPTIVE GRID

The present grid adjustment scheme has evolved from one which imparted
velocities to grid points based on gradients in the field as in Reference\(^11\)
and\(^12\) to one which rezones the computational mesh using an implicit algorithm
at any desired frequency (i.e., once per time step or once per 1000 time steps).
Originally every point \((i,j)\) in the computational plane was connected to the
four adjacent points \((i+1,j), (i,j+1), (i-1,j), (i,j-1)\) by springs whose spring constants
\(k_i, k_j\) were determined by a function of the gradient of some dependent
variable between the points. For example, the spring along the ith row
connecting points \((i,j)\) and \((i,j+1)\) was defined
\[ k_j^i = 1 + c|f_{i,j} - f_{i,j+1}|/\Delta r \]  

where \( c \) is a constant and \( \Delta r \) is the distance between the points. Grid points on the boundaries were free to move along the boundary as if on a frictionless rod. Grid points on the corners of the domain were held fixed. Small adjustments in the \( x \) and \( y \) directions were assigned to grid points as determined by the net forces on the grid points in the respective directions. After every time step, the grid was updated and checked to make sure that no grid overlap was imminent.

In a noninteractive test case where \( p(x,y) \) was prescribed to model the pressure field of an oblique shock crossing a uniform flowfield (\( p(t) = 0 \)), the grid evolved from one of equally spaced rectangular cells to the one shown in Figure 2. In an interactive test case where the solution \( p(x,y,t) \) was allowed to evolve with time, the final grid distribution (Fig. 3) is seen to be much more erratic than that of the previous case. The pressure field for this case is shown in Figure 4.

By sacrificing the column to column (or row to row) influence of the spring system (i.e., no springs between column \( i \) and column \( i+1 \)) a superior algorithm in terms of computational costs and smoothness of grid distribution can be constructed. (Smoothness of grid distribution is a subjective judgment. A coordinate line whose direction or length changes erratically from point to point is judged nonsmooth.)

Consider the coordinate line of constant index \( i \) as shown in Figure 5.

\[ f_{j,6} = x_j, f_{j,2} = y_j, f_{j,1} = 0_j, f_{j,4} = u_j, f_{j,5} = v_j, f_{j,6} = e_j \]  

The grid points in physical \((x,y)\) space are mapped onto a straight line in \( s \) space. Let each grid point be connected to its neighbor in \( s \) space by a spring with spring constant \( K_j \) between the \( j \) and \( j+1 \) points to be defined later.
Let $\Delta n_j$ be the distance between the $j$ and $j+1$ points in space. Then $K_j \Delta n_j$ equals a constant. Therefore,

$$\Delta n_j = K_j \Delta n_1 / K_j$$  \hspace{1cm} (13)

The known total length of $s$ can be expressed as

$$S_{\text{JN}+1} = \sum_{j=1}^{\text{JN}} \Delta n_j = \Delta n_1 K_1 \sum_{j=1}^{\text{JN}} 1/K_j = \Delta n_1 K_1 \text{SUM}$$  \hspace{1cm} (14)

where $\text{JN} = \text{total number of intervals}$. Consequently,

$$\Delta n_1 = S_{\text{JN}+1} / (K_1 \text{SUM})$$

$$\Delta n_j = S_{\text{JN}+1} / (K_j \text{SUM})$$  \hspace{1cm} (15)

The constants $K_j$ can be defined explicitly or implicitly. An explicit definition of $K_j$ assigns a value to the spring constant independent of the position of the spring in space. For example,

$$K_j = \exp(-c_1 \cdot j / \text{JN})$$  \hspace{1cm} (16)

provides an exponentially increasing spacing of grid points from $j = 1$ to $j = \text{JN}$ where the constant $c_1$ is used to control stretching. This definition provides no adaptive capability but does provide a very quick way of concentrating points near boundaries.

An implicit definition of $K_j$ assigns a value to the spring constant which is a function of the position of the spring in space. For an implicit case, the algorithm is implemented as follows:

1. Start at the first column (or row) of data and compute and store all the values of $s_j$ and $f_j,m$.
2. Initialize $\hat{n}_j$ and $n_j$ equal to $s_j$ for all $j$, ITER = 0.
4. Using the known stored values of independent variable $s_j$ and dependent variables $f_j,m$, compute the new values of the dependent variables $d_j,m$ at all $n_j$ with a univariate interpolation routine where $d_j,1 = x_j$, $d_j,2 = y_j$, etc.
5. Compute $K_j = K_j(d_j,m)$. 

6) Compute the new coordinates \( n_j \) from Equation (15); set \( n_{JN+1} = e_{JN+1} \) to keep nodal point values unchanged.

(7) Compute an error norm to determine convergence

\[
L_2 = \sum_{j=2}^{JN} (n_j - \bar{n}_j)^2 \quad \text{(17)}
\]

If \( L_2 < 0.0001 \), proceed to step (9).
If \( \text{ITER} > 100 \), stop the procedure.

(8) Set \( \bar{n}_j = n_j \) for all \( j \) and go to step (3).

(9) Replace \( x_j, y_j, \rho_j, \) etc., with \( d_{j,1}, d_{j,2}, d_{j,3}, \) etc., go to next column of data and repeat all steps.

Typical definitions of \( K_j \) employ gradients of velocity, internal energy, or Mach number. For example,

\[
K_j = 1 + c_3 \left[ e(n_{j+1}) - e(n_j) \right] / (n_{j+1} - n_j) \quad \text{(18)}
\]

where \( c_3 \) can be used to control the concentration of mesh points in regions of high gradient. Examples of grid distribution using these gradient adaptive mechanisms will be presented in the RESULTS section.

In cases where a high gradient develops rapidly, as in the vicinity of an expansion corner of a planetary probe, the adaption algorithm can have trouble converging. This problem is overcome as follows.

(1) After step (4) in the adaption algorithm, the overall change in the new grid distribution \( n_j \) is damped by writing

\[
n_j = c_4 n_j + (1 - c_4) \bar{n}_j \quad \text{(19)}
\]

where \( 0 < c_4 \leq 1 \) \( (c_4 \) is typically of order \( 1/2 \))

(2) After step (5) in the adaption algorithm, the values of the spring constants are filtered by writing

\[
K_j^{n+1} = (K_j^{-1} + 2K_j + K_{j+1})^{7/4} \quad j = 2, JN+1 \quad \text{(20)}
\]

Often the convergence problems can be overcome by keeping the rezoning interval small or by avoiding large changes of \( c_3 \) in Equation (18) between rezonings.

The major advantage of the rezoning procedure described herein is that it provides a very quick way of providing an adaptive capability to the
finite-volume formulation. The option to control rezoning frequency allows the grid to dynamically adapt to developing features of the flow (i.e., shock, shear layers) and then to be held fixed as steady state is approached. The rezoning procedure can also be used as a separate program to study the effects of monitoring various gradients in the field or using different definitions of $K_j$.

RESULTS AND DISCUSSION

The finite-volume, adaptive grid algorithm described in the previous sections has been used to compute supersonic flowfields over spheres, cylinders and two planetary probe configurations. The most comprehensive tests to date for testing the adaptation algorithm have been on the Viking Aeroshell (Fig. 6), a configuration which protected the Viking lander for its descent through the Martian atmosphere. This section will deal exclusively with those results. The freestream conditions, $M_\infty = 2, \gamma = 1.285, Re_\infty = 5000$, have been chosen to permit comparison with the results of Reference 14. Subsequent calculations at higher Reynolds numbers have been generated to demonstrate the capabilities of the adaptation algorithm.

The grid shown in Figure 6 was achieved with an explicit definition of $K_j$ (Eq. (16)) with $c_1/\sqrt{\gamma} = 0.15, 90$ points around the body and down the wake centerline and 31 points between the body/wake centerline and the bow shock. The pressure distribution around the body/wake centerline is presented in Figure 7. This core was run to convergence in approximately 55 minutes of computing time with a timing of $3 \times 10^{-5}$ sec/iteration/grid point. Recent changes in the code effecting the manner in which data are stored have decreased timing to $1.6 \times 10^{-5}$ sec/iteration/grid point.

An adaptive grid calculation which monitored velocity gradients was applied to this case problem. In the previous cases, the grid was not sufficiently stretched to adequately resolve the boundary layer at the separation point. Since the separation region is very important in determining the nature of the flow in the wake, it was decided to run this case with 91 points across the shock layer. The converged grid for $c_3 = 0.5$ is shown in Figure 8. In defining the values of $K_j$ near the wall, a test was included that would make $K_j = 3, j \leq 4$ if $K_j$ was previously less than 3. From $j = 5$ to 9, the minimum allowable value of $K_j$ was decreased linearly to zero. This forced a concentration of points near the body and in the wake. Pressure distribution is plotted in Figure 7. Comparison with numerical results of
Reference 14 for shock shape and pressure distribution down the wake centerline show generally good agreement in Figures 9 and 7, respectively.

Solutions for the higher Reynolds number cases ($5 \times 10^4$, $5 \times 10^6$) were generated sequentially using the previous results as initial conditions. All of these results are for laminar flow and so while the highest ones are physically in error they test the adaptation routine's ability to resolve the large gradients associated with such flows.

Some interesting results of these high Reynolds number studies are presented in Figures 10 through 17. The converged grid showing a concentration of points in the free shear layer is presented in Figures 10. The streamlines pattern around the expansion corner of the vehicle and associated Mach number contours are presented in Figures 11 and 12. The outer extent of these figures is defined by the physical location of the 25th mesh point. The velocity and grid point distributions along the $\eta$ coordinate line (i=22) located just ahead of the expansion corner are given in Figure 13 for $Re_{\infty} = 10^4$ and $10^6$.

These results were obtained using twenty passes through Equation (20) with $c_3 = 0.5$ and 1.0 respectively. The global internal energy contour plot in Figure 14 shows the high gradient regions in the shear layer behind the expansion corner, the captured recompression shock in the wake and a small, high gradient shock-like region on the symmetry line just behind the base where recirculating velocities rapidly change from supersonic to subsonic. This phenomenon was also observed in the results of Reference 14 using a different numerical approach. The high gradient free shear layer extends approximately one-fourth maximum body radius beyond the corner after which point the gradients rapidly decrease and the flow turns toward the axis. There is a large region just below the shear layer of low density ($0.2 \leq \rho \leq 0.7$) recirculating flow. A Mach number distribution along an $\eta$ coordinate line (i=34), Figures 15, shows the high gradient region end distribution of mesh points through the shear layer. The complexity of the flow in this region end the ability of the adaption process to concentrate mesh points in the high gradient regions away from the well further demonstrate the versatility and sensitivity of the adaption processes.

CONCLUDING REMARKS

The Finite-Volume Formulation (FVF) described herein has been shown to be a consistent and conservative approximation to the Navier-Stokes equations. The formulation has a first order error in wave speed, proportional to the local value of $\Delta t$, when the local maximum time step is used to advance the
solution at every point. The formulation is second order accurate when constant time increment is used to advance the solution. However, at large Reynolds numbers (small cell volumes) the constant $\Delta t$ advancement of the solution, using one value of $\Delta t$ that is stable for all points, is impractical because many more iterations are required to converge the solution. Converged solutions for complete blunt body flows using approximately 3000 mesh points can be obtained within 30 minutes using the variable $\Delta t$ option and adaptive grid rezoning on the CYBER 203 computer.

In all of the problems considered herein adaptation along only one coordinate line is quite sufficient for the purpose of moving mesh points to high gradient regions in the flow. Restricting grid motion along one coordinate direction permits application of a highly efficient implicit grid adaptation procedure. The routine can be implemented at any desired frequency thus permitting dynamic adaptation or cost savings by adapting only after large increments of iteration count. Grid point concentration in the boundary layer and free shear layer have demonstrated the success and versatility of the adaption algorithm.

REFERENCES

The Use of Adaptive Grids in Conjunction with Shock Capturing Methods, AIAA Paper 81-1012.


Fig. 1 Schematic showing relation of indexing system, physical \((x,y)\) coordinates, and computational \((\xi,\eta)\) coordinates.

Fig. 2 Noninteractive example of grid adapting to prescribed pressure distribution with two degrees of adaptive freedom.

Fig. 3 Interactive example of grid adapting to computed pressure distribution across oblique shock with two degrees of adaptive freedom.

Fig. 4 Perspective view of pressure on \((x,y)\) plane for oblique shock calculation on interactive grid.

Fig. 5 Schematic showing spring system for one degree of adaptive freedom along \(\eta\) coordinate line.
Fig. 6 Grid over Viking Aeroshell obtained with explicit definition of $X_j$, $C_{l}/N=15$.

Fig. 8 Grid over Viking Aeroshell obtained with implicit definition of $X_j$, Re=5000.

Fig. 7 Pressure distribution around body/wake centerline of Viking Aeroshell. $M_\infty=2$, Re=5000, $\gamma=1.289$.

Fig. 9 Bow shock over Viking Aeroshell for $M_\infty=2$, Re=5000.

O FINITE VOLUME
--- REF. 14

O ADAPTED GRID 49 POINTS
--- EXPLICITLY STRETCHED GRID 37 POINTS
--- REF. 14, APPROXIMATE BODY INCLUDES SLIP CONDITIONS
Fig. 10 Grid over Viking Aeroshell obtained with implicit definition of $K_{f}$, $Re = 10^6$.

Fig. 11 Streamlines around expansion corner of Viking Aeroshell, $M_\infty = 2$, $Re = 10^6$.

Fig. 12 Mach number contours around expansion corner of Viking Aeroshell, $M_\infty = 2$, $Re = 10^6$.

Fig. 13 Velocity and grid spacing distribution along $\eta$ coordinate line ahead of expansion corner, $Re = 10^8$ and $10^6$. 
Fig. 14 Contour plot of internal energy over Viking A roshell for $M_a=2$, $Re_a=10^5$.

Fig. 15 Mach number and grid spacing distribution through separated shear layer behind expansion corner along $n$ coordinate line.