JOURNAL OF AIRCRAFT
Vol. 30, No. 2, March–April 1993

Solution-Adaptive and Quality-Enhancing Grid Generation

Phu V. Luong,* Joe F. Thompson,† and Boyd Gatlin‡
Mississippi State University, Mississippi State, Mississippi 39762

A straightforward and effective procedure for converting any CFD code that operates on a block-structured grid into a dynamically adaptive code is presented. Coupling between the grid generation system and the flow solver is through external files, so that only minimal changes are required in the source codes. Adaptive grids are generated using elliptic equation techniques. The mechanism of grid movement is control functions, which form the right side of the system of generating equations. These control functions can be specified to include grid quality measures, flow solutions, solution gradients, solution curvature, or any combination of these. Adaptation can be static, dynamic, or periodic. Typical results for adaptation in compressible and incompressible flows are presented and compared with solutions obtained on static grids.

Introduction

THE requirements of accuracy and efficiency for obtaining numerical solutions to PDEs have always been in conflict. While increasing the number of grid points implies decreasing the local truncation error, this results in long computation times and large memory demands. On the other hand, reducing the number of points produces a concomitant reduction in cost, but at the expense of accuracy. Adaptive gridding is a means of resolving this conflict and can lead to better flow solutions at reduced cost. Several techniques for adaptive grid generation techniques are discussed in Ref. 1.

In this work, the control functions are the mechanism for adapting grids, as detailed in Refs. 2 and 3, in which preliminary development of the present effort is described in detail. In contrast to these earlier efforts, the elliptic grid generation procedure here has been separated from the main grid code, which is derived from the EAGLE[4] grid system. The elliptic grid routine now can be called either by a flow code to generate a new adaptive grid based on flow variables and quality measures, or by the grid code itself to produce a grid based on quality measures through static adaptation. Arrays of flow variables can be read into the EAGLE grid code for use in static adaptation as well. In any case, communication between the grid system and the flow solver is through scratch files so that a given CFD code can be turned into an adaptive code with minimal changes to the source code.

The adaptive mechanism has also been extended to include adaptation to either the gradient of a variable, as in the original case,[2,3] to the curvature of a variable, or to the variable itself. In addition, the ability to calculate the weight functions as weighted averages of weight functions from several flow variables, as well as quality measures, is now provided. This allows the adaptation to take into account the effect of many of the flow variables instead of just one, and provides for different weight functions in each coordinate direction. The construction of the weighted average of flow variables and quality measures, and the choice of adaptation to gradient, curvature, or variable are all controlled in each coordinate direction through user input. The quality measures currently available in the system are skewness, aspect ratio, arc length, and smoothness of the grid as measured by the grid Laplacian. These grid quality measures and the resulting control function and weight function values can be output for graphical contouring.

The adaptive EAGLE system has been coupled with the INS3D[5] incompressible flow solver developed at NASA Ames Research Center and the MISSE Euler codes,[6] among others. In principle, it can be linked with any finite-difference or finite-volume PDE solver which operates on a block-structured grid.

Elliptic Grid Generation

Among the various techniques for generating grids with partial differential equations, sets of elliptic equations derived from the Laplace, or Poisson equations, are the most common. As can be shown from the variational calculus, the Laplace system produces the smoothest possible grid, since when it is satisfied the grid is uniformly spaced. Therefore, even with nonuniform boundary distributions, the coordinate lines in the interior of the field tend to be equally spaced. Control of the coordinate line distribution in the field can be obtained with an elliptic system derived from the Poisson equations

$$\nabla^2 \xi^i = P^i, \quad i = 1, 2, 3 \tag{1}$$

where the functions P^i serve to control the coordinate line spacing.

Warsi[7] has shown that if a curvilinear coordinate system $\tilde{\xi}^i$, which satisfied the Laplace system $\nabla^2 \tilde{\xi}^i = 0$, is transformed to another coordinate system ξ^i, then the new curvilinear coordinates ξ^i satisfy the inhomogeneous elliptic system as defined by Eq. (1) with the control functions

$$P^i = \sum_{j=1}^{3} \sum_{k=1}^{3} g^{jk} P^i_{jk}, \quad i = 1, 2, 3 \tag{2}$$

with the P^i_{jk} defined by the transformation from $\tilde{\xi}^i$ to ξ^i by

$$P^i_{jk} = \sum_{m=1}^{3} \sum_{n=1}^{3} \frac{\partial \tilde{\xi}^m}{\partial \xi^j} \frac{\partial \tilde{\xi}^n}{\partial \xi^k} \frac{\partial^2 \xi^i}{\partial \tilde{\xi}^m \partial \tilde{\xi}^n}$$
$$g^{ik} = \frac{1}{g}(g_{jm} g_{ln} - g_{jn} g_{lm}) \tag{3}$$

with (i, j, l), (k, m, n) cyclic. Here g is the square of the Jacobian of the transformation and $g_{ij} = \boldsymbol{r}_{\xi^i} \cdot \boldsymbol{r}_{\xi^j}$ are the elements of the covariant metric tensor.

Received April 25, 1991; presented as Paper 91-1593 at the AIAA 10th Computational Fluid Dynamics Conference, Honolulu, HI, June 24–27, 1991; revision received Feb. 20, 1992; accepted for publication March 10, 1992.

*Mathematician, Naval Oceanographic Office, Code MSA, Stennis Space Center, Mississippi 39522.

†Distinguished Professor of Aerospace Engineering and Director, NSF Engineering Research Center for Computational Field Simulation, P.O. Box 6176. Member AIAA.

‡Assistant Professor of Aerospace Engineering, P.O. Drawer A. Member AIAA.

In these relations, $\boldsymbol{r} = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}$ is the Cartesian position vector of a grid point, and ξ^i, $i = 1, 2, 3$, are the three curvilinear coordinates. The combination of Eqs. (1) and (2) gives

$$\nabla^2 \xi^i = \sum_{j=1}^{3} \sum_{k=1}^{3} g^{jk} P^i_{jk}, \quad i = 1, 2, 3 \tag{4}$$

These results show that a coordinate system obtained by application of a stretching transformation to one generated as the solution of the Laplace system, can be generated directly by solving Eq. (4) with appropriate control functions P^i_{jk} as defined by Eq. (3). Therefore, the Poisson system [Eq. (1)] can be taken as the generation system with the control functions considered to be specified. Among these control functions, P^i_{ii}, $(i = 1, 2, 3)$ are the most important since they correspond to one-dimensional stretching in each coordinate direction. By taking all the other control functions to be zero, i.e, $P^i_{jk} = \delta^i_j \delta^i_k P_i$, Eq. (4) becomes

$$\nabla^2 \xi^i = g^{ii} P_i, \quad i = 1, 2, 3 \tag{5}$$

Transformation of Eq. (5) into curvilinear space then yields

$$\sum_{i=1}^{3} \sum_{j=1}^{3} g^{ij} \boldsymbol{r}_{\xi i \xi j} + \sum_{k=1}^{3} g^{kk} P_k \boldsymbol{r}_{\xi k} = 0 \tag{6}$$

which is the form commonly used. The spacing and orientation of grid lines in the field are controlled by the control functions P_i. For example, a negative value of P_i causes the coordinate lines to concentrate in the direction of decreasing ξ^i. Control functions can be evaluated such that the grid generated by Eq. (6) reflects the spacing of some initial grid, generated perhaps by algebraic methods. Or they may be based upon boundary properties such as spacing and curvature, so that these are projected into the field. Procedures for the determination of control functions in the EAGLE code are discussed in Ref. 4.

Adaptive Grid Generation

Variational Approach

Minimization of the integral of some grid property over the computational domain is known as the variational approach. The resulting Euler variational equations from the calculus of variations then constitute the grid generation system. The choice of what property is to be minimized depends upon what is expected from the grid. For example, Saltzman and Brackbill[8] developed adaptive grids by minimizing a weighted combination of integrals which emphasize smoothness, orthogonality, and point concentration. A similar approach developed by considering smoothness, a measure of the grid cell area, and the orthogonality of the grid lines can be found in Ref. 9. Several other grid properties that might be considered, such as the square of cell volume or inverse cell volume, are discussed by Thompson and Warsi.[10]

However, due to the complexity of the Euler equations they are difficult to solve and solution algorithms may not converge. Kennon and Dulikravich[11] and Carcaillet[12] have developed algorithms for the direct solution of the variational problem. In these efforts, different discrete problems were formulated as unconstrained optimization problems and then solved by a conjugate gradient method. A survey of the types of integrals that may be included in a variational problem and the geometric properties that each integral imposes upon the grid can be found in Ref. 13.

Control Function Approach

The control function approach to adaptation is developed by noting the correspondence between the one-dimensional form of Eq. (6)

$$x_{\xi\xi} + P x_\xi = 0 \tag{7}$$

and the differentiated form of the equidistribution principle $W_\xi = \text{const}$

$$W x_{\xi\xi} + W_\xi x_\xi = 0 \tag{8}$$

where P is the function to control the coordinate line spacing, and W is some weight function.

From Eqs. (7) and (8), the control function can be defined in terms of the weight function and its derivative as

$$P = (W_\xi / W) \tag{9}$$

This equation can be expressed in a general three-dimensional form as

$$P_i = (W_{\xi^i} / W) \tag{10}$$

This approach was developed by Anderson,[14,15] and has been applied with success in two-dimensional configurations by Johnson and Thompson[16,17] and in three-dimensional configurations by Kim and Thompson,[2] and by Tu and Thompson.[3]

The complete generalization of Eq. (10) was proposed by Eiseman[18] as

$$P_i = \sum_{j=1}^{3} \frac{g^{ij}}{g^{ii}} \frac{(W_i)_{\xi j}}{W_i} \tag{11}$$

where W_i is the weight function chosen for the ξ_i direction. This definition of the control functions provides a convenient means of specifying three separate functions, with one in each coordinate direction.

In order to preserve the geometric characteristics of the existing grid the control functions are constructed so that those defined by Eq. (11) can be added to the initial set, thus giving the grid a memory. To wit

$$P_i = (P_i)_g + C_i (P_i)_W, \quad i = 1, 2, 3 \tag{12}$$

where

$(P_i)_g$ = control function based on geometry
$(P_i)_W$ = control function based on weight function
(C_i) = weight coefficient to be specified

In these equations the weight function W is defined according to what one wishes to adapt to. The geometric contribution $(P_i)_g$ can be based upon either the starting grid or the previous adapted grid, with the latter approach bringing greater deviation from the original grid. For adaptation to

$$\text{Variable;} \quad W = 1 + |V| \tag{13a}$$

$$\text{Gradient;} \quad W = 1 + |\nabla V| \tag{13b}$$

$$\text{Curvature;} \quad W = (1 + \beta |K|) \sqrt{1 + \alpha |\nabla V|^2} \tag{13c}$$

where V can be any scalar component of the solution, some derivative of the solution, such as the magnitude of vorticity, a grid quality measure, or some measure of truncation error. Here $\beta, \alpha \in [0, 1]$ and

$$K = \frac{\nabla^2 V}{(1 + |\nabla V|^2)^{3/2}} \tag{14}$$

is the usual definition of the curvature of V.

With the control functions defined in this way, the elliptic generation system given by Eq. (6) becomes an adaptive system. It is then solved iteratively by point SOR to generate the adapted grid. The control function approach to grid adaptation can be obtained directly from a variation principle, as is shown in Ref. 10.

Grid Quality Measures

Grids can be evaluated qualitatively and, to some extent, quantitatively, by the examination of certain computable properties. These properties are related to grid quality in that they are directly or indirectly linked to local truncation error or algorithm convergence. Following Kerlick and Klopfer,[19] and Gatlin,[20] the grid quality measures considered here are skew angle, aspect ratio, grid Laplacian, and spacing arc length. In three-dimensions, three values can be computed for each property at each point. The approach taken here is to treat each surface of constant ξ separately for the sake of clarity.

In adapting grids to these computed properties, no control law formulation is applied to directly reduce or increase the property being adapted to. This was intentionally avoided because of the high computational penalty. Instead, grid points are moved to regions which have relatively more skewness or less smoothness, for example, so that overall truncation error may be reduced. In any case, it is not intended that the grid quality measures be used alone, but in conjunction with solution properties.

Skew Angle

The minimum skew angle between intersecting grid lines is one of the most important measurable grid properties. This angle can be expressed in terms of the covariant metric elements as

$$\theta_{ij} = \cos^{-1}\left(\frac{g_{ij}}{\sqrt{g_{ii}g_{jj}}}\right) \tag{15}$$

Since $g_{12} = g_{21}$, $g_{13} = g_{31}$, and $g_{23} = g_{32}$, the three skew angles associated with each grid point in a three-dimensional grid are θ_{12}, θ_{23}, and θ_{31}. The choice of the minimum as opposed to the maximum angle is arbitrary.

Aspect Ratio

Clearly aspect ratio can be defined in two different ways on any coordinate surface. For example, on a surface of constant ξ^k the ratio can be expressed in terms of metric elements g_{ii} and g_{jj} as

$$AR_{ij} = \sqrt{g_{ii}/g_{jj}} \tag{16}$$

Large changes in aspect ratio of grids from one part of the field to another are known to inhibit convergence.

Laplacian

A measure of the smoothness of a grid is the Laplacian of the curvilinear system, which is simply the rate of change of grid point density. For a perfectly uniform grid, the grid Laplacian would vanish everywhere, but exceedingly large values may arise in highly stretched grids. When a coordinate transformation is applied so that the Cartesian coordinates are the dependent variables, the grid Laplacian is given in terms of the contravariant metric elements g^{ij}, the contravariant base vectors $\boldsymbol{a}^l$, and the position vector $\boldsymbol{r}$ as

$$\nabla^2\xi^l = -\sum_{i=1}^{3}\sum_{j=1}^{3} g^{ij}\boldsymbol{a}^l\cdot\boldsymbol{r}_{\xi^i\xi^j}, \quad l = 1, 2, 3 \tag{17}$$

Arc Length

Another important measure of grid quality is the local rate at which grid spacing changes. On a coordinate surface of constant ξ^k, and along a coordinate line of constant ξ^j, the grid spacing is just

$$d_i = [(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2(z_{i+1} - z_i)^2]^{1/2} \tag{18a}$$

The rate at which grid spacing changes (ARCL) is then just

$$(\text{ARCL})_i = \frac{d_i - d_{i-1}}{\frac{1}{2}(d_i + d_{i-1})} \tag{18b}$$

Results and Discussion

The adaptive grid generation system based on the control function approach as described above has been applied in static and periodic modes. In static mode, the initial grids were adapted to either grid quality measures or to existing flow solution variables. In periodic mode, multiple adaptations were applied to several different configurations with a compressible flow code[6] for inviscid flow, and with INS3D for incompressible viscous flow.

Adaptation to Quality Measures

Some examples of the grid quality adaptation are shown in Fig. 1. In each case, the grid was adapted 21 times, with updating of the quality measures before each adaptation. Geometric control functions, as defined in Eq. (12), were

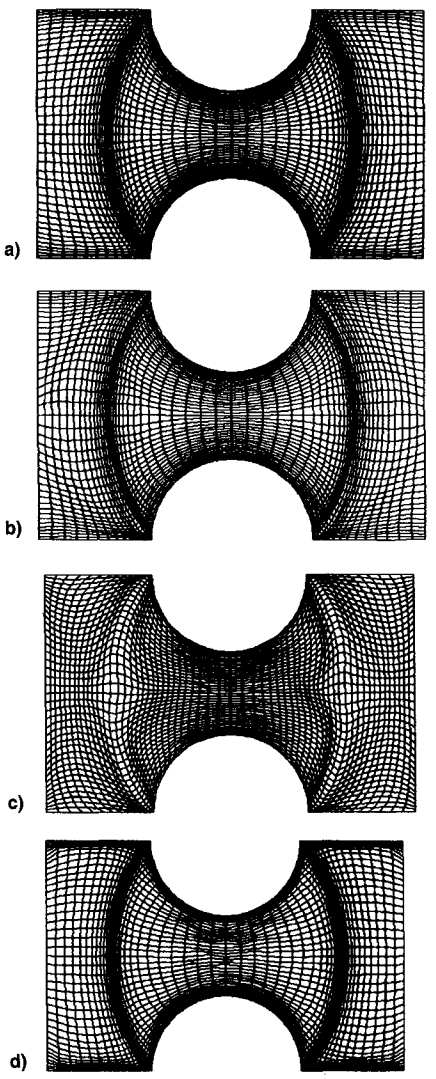


Fig. 1 Examples of adaptation to grid quality measures: a) elliptic grid with boundary orthogonality; b) adaptation to grid skewness; c) adaptation to aspect ratio; and d) adaptation to grid Laplacian, a measure of smoothness.

based upon the previous grid rather than upon the initial grid. An unadapted grid generated elliptically with boundary orthogonality control is shown in Fig. 1a. In Figs. 1b–d results of adaptation to skew angle, aspect ratio, and smoothness are shown.

Although not obvious from Fig. 1, the color contour plots of Ref. 21 show that adaptation to the skewness is effective in reducing the skewness in one region, while increasing the skewness in other regions of the grid. A small improvement in aspect ratio occurs, but the smoothness of the grid is decreased. Adaptation to aspect ratio improves both aspect ratio and smoothness of the grid; the skewness is increased, however. Adaptation to smoothness as measured by the grid Laplacian improves the skewness and aspect ratio of the grid effectively, but the adaptive grid is not as smooth as the initial elliptic grid. The most pleasing results were obtained by adapting to a combination of several quality measures simultaneously.

The geometry chosen of this demonstration case represents a generic blade passage or tube-bank passage, with no symmetry assumed. While adaptations in the sense of local grid refinement through point migration seem to have little value when used alone, they can be coupled with adaptations to solution properties, thus providing finer resolution where errors are likely to occur due to other grid deficiencies.

In addition to static adaptation to grid properties, there is also provision for adaptation to an existing flow solution. Since it is often not clear what weight functions or how many adaptive iterations will produce an efficient grid for a particular class of flow, the user can use the static adaptive mode to help determine these parameters. It enables one to observe the effects of adaptation parameters before solving the problem of interest.

Adaptation to Solution Properties

True dynamic adaptation requires that additional terms be added to the discretized fluid flow equations to account for grid motion relative to the flowfield. While this is certainly a reasonable thing to do, it is a clear impediment to rapid conversion of an existing code into an adaptive solver. The addition of the necessary terms generally requires changes to the FORTRAN code far greater than those needed to link the solver with the adaptive EAGLE system described here. Multiple, periodic adaptation has shown to be an efficient alternative to dynamic adaptation, since a new grid need not be produced each time step, and no significant code modifications are required. In this periodic mode, the solution can be interpolated from the old grid to the new grid if desired, but one may also simply use the existing solution as starting values on the new grid without interpolation. Such is the approach in the results given here.

Supersonic Flow over Double Wedge

Results obtained from flow at Mach 2 over a double wedge in a wind tunnel are shown in Figs. 2–7. Both fine (121 × 41) and coarse (81 × 31) grids were used, and the resulting solutions can be compared. Figure 2 shows the pressure contours obtained with 300 time steps on both the initial grid and on the adapted grid. The grid was adapted to the density gradient in the streamwise direction, with weight coefficient $C_1 = 0.7$, and to the pressure gradient in the normal direction, with $C_2 = 0.5$. Four adaptations were performed for this case with control functions evaluated from the most recent grid. The pressure contours are clearly packed more tightly at shock waves in the adapted case, but it is also clear that such a solution must be more costly. A record of total CPU time on a Silicon Graphics IRIS 4D/440VGX shows that the solution on the static grid required 1480 CPU s, while the adaptive solution consumed 1600 CPU s. However, shock resolution has been increased dramatically at only an 8% increase in CPU time.

In Fig. 3 coarse grid solutions for the same flow conditions are shown. Both the static solution and the adaptive solution

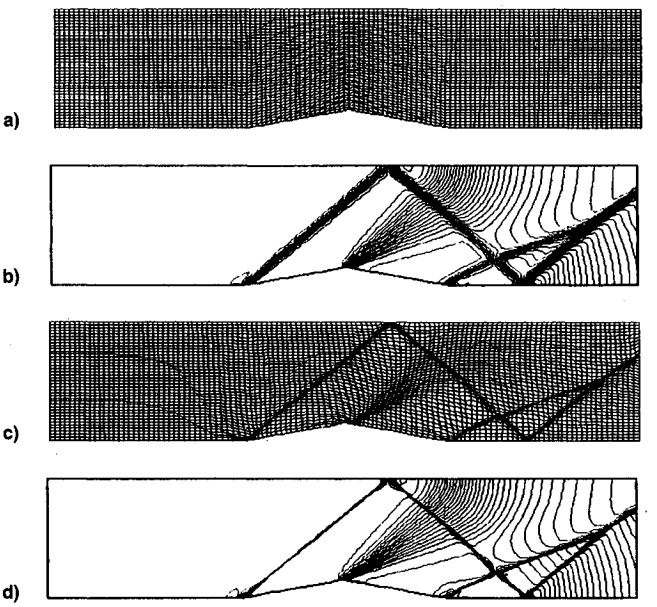


Fig. 2 Comparison of pressure contours at Mach 2 on (121 × 41) static and adaptive grids after 300 time steps. First adaptation at 100 time steps, the last at 200: a) unadapted elliptic grid; b) pressure contours for static grid solution; c) grid adapted four times to density gradient in streamwise direction and pressure gradient in normal direction; and d) pressure contours for adapted grid solution.

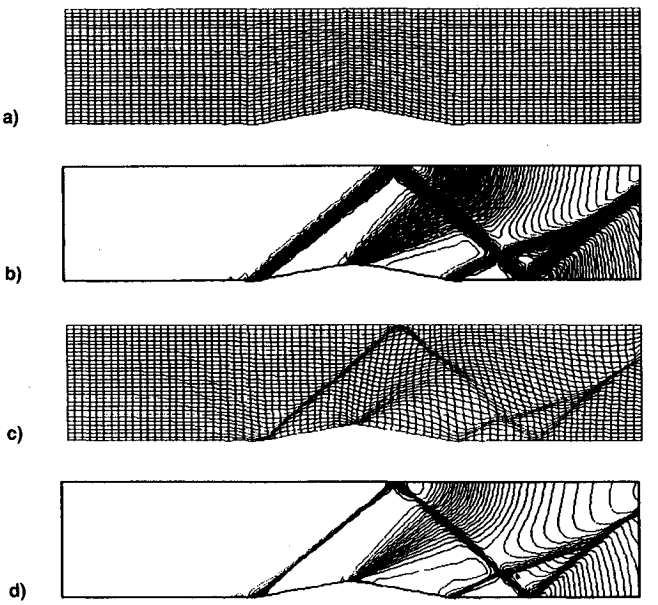


Fig. 3 Pressure contours at Mach 2 on (81 × 31) static and adaptive grids after 300 time steps. First adaptation at 100 time steps, the last at 200: a) unadapted elliptic grid; b) pressure contours for static grid solution; c) grid adapted four times to density gradient in streamwise direction and pressure gradient in normal direction; and d) pressure contours for adapted grid solution.

may be compared with the fine-grid solutions of Fig. 2. This relatively coarse grid was adapted to a combination of density and pressure in the ξ^1 direction, with weight coefficient $C_1 = 0.5$, and to the gradient of the same combination along the ξ^2 coordinate, with $C_2 = 0.5$.In Fig. 4 it is difficult to discern any difference in the wall pressure coefficients, even though the shock wave is sharper in the adapted case.

A variety of different adaptive parameters applied to the coarse grid in the multiple adaptation process are shown in Fig. 5. Figure 5b shows the pressure contours obtained on the adaptive grid of Fig. 5a. The initial grid was adapted to the curvature of density and pressure in both directions. The

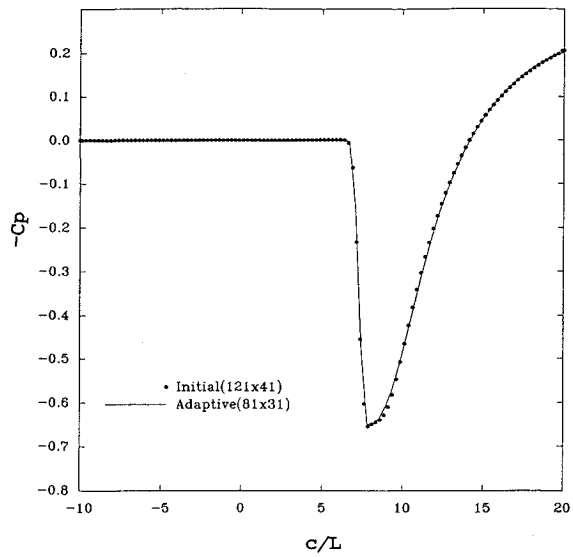


Fig. 4 Wall pressure coefficient for fine static grid solution of Fig. 2 and coarse adapted grid solution of Fig. 3.

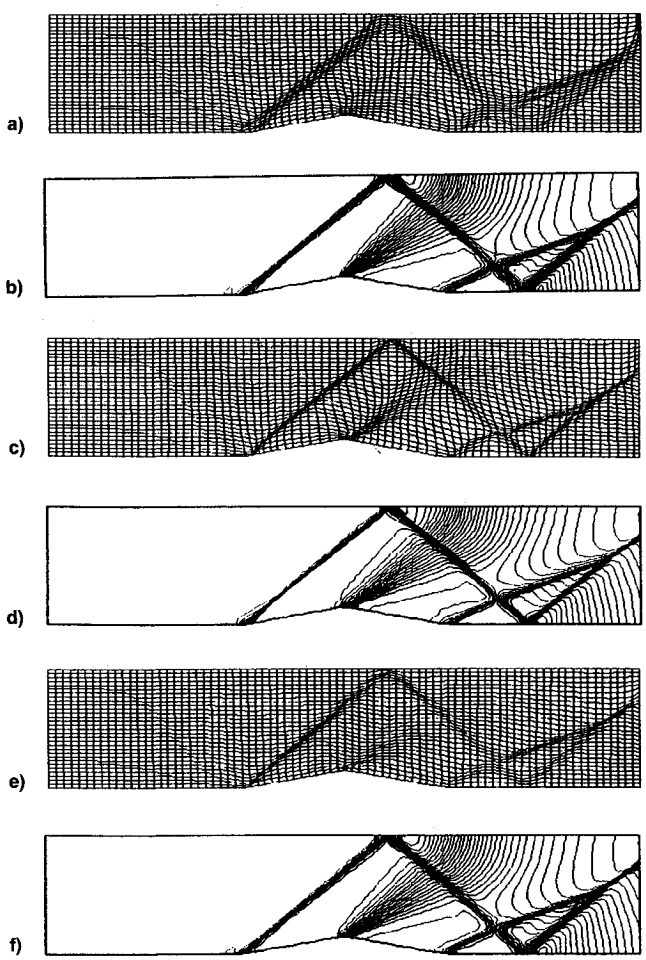


Fig. 5 Adapted grids and resulting pressure contours for (81 × 31) grid for Mach 2 flow. Each solution ran for 300 time steps and four adaptations, the first after 100 time steps. Solutions can be compared with Figs. 2 and 3: a) grid adapted to curvature of pressure and density in both directions, with final adaptation at 180 time steps; b) pressure contours for the solution obtained on a); c) grid adapted to gradient of pressure in both directions, with final adaptation at 160 time steps; d) pressure contours for the solution obtained on c); e) grid adapted to gradient of pressure and density in normal direction only, with final adaptation at 200 time steps; and f) pressure contours for the solution obtained on e).

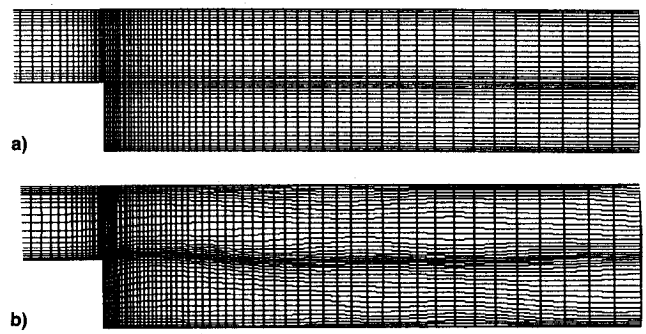
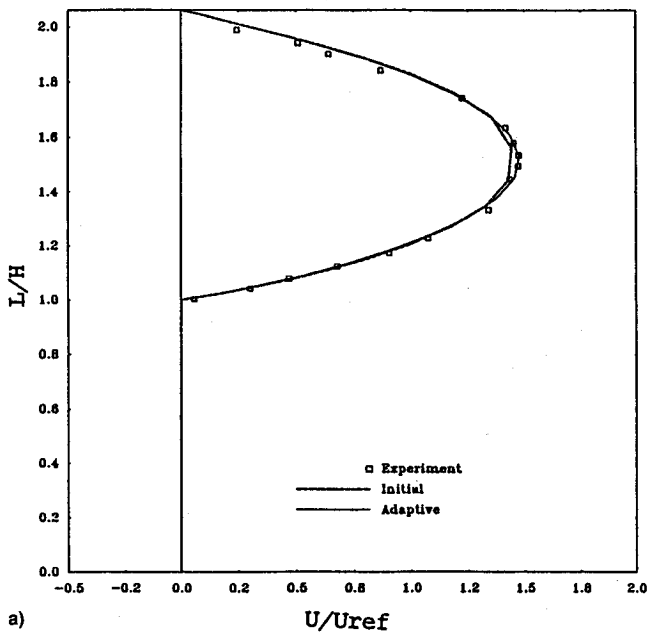


Fig. 6 Two-block grid for laminar flow over back-facing step. Block boundary is at step face, with (35 × 21) points in first block and (81 × 41) in second. Reynolds number is 183: a) initial algebraic grid, and b) grid after two adaptations to vorticity in direction normal to mean flow at 500 and 1000 time steps.



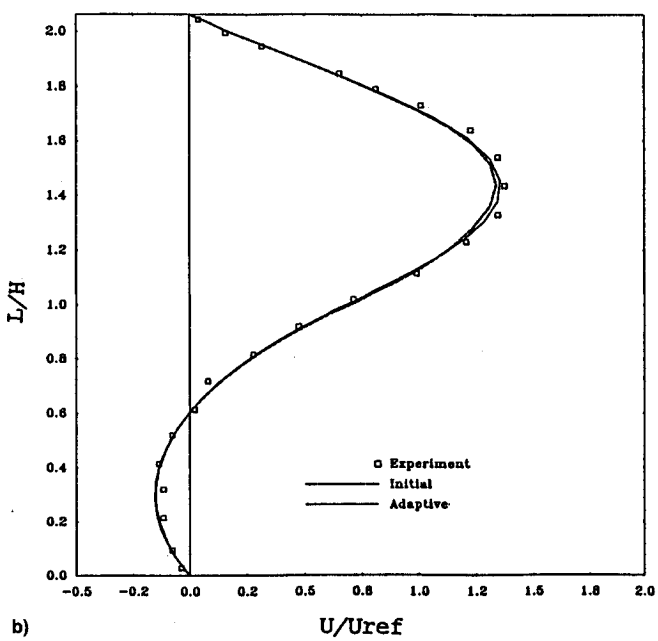


Fig. 7 Velocity profiles from experimental data compared with those computed on static and adapted grids of Fig. 6 for laminar flow over back-facing step: a) profiles at the step, which corresponds to block interface in the computations, and b) profiles at 2.77 step heights downstream.

total number of adaptations was four, with $C_1 = C_2 = 0.7$. The coefficients of the gradient and curvature were $\alpha = 1$ and $\beta = 0.5$, respectively. Geometric control functions were based upon the initial grid.

Figure 5d shows the pressure contours obtained on the adaptive grid of Fig. 5c. The adaptive mechanism for this case was pressure gradient in both directions, with $C_1 = C_2 = 0.7$ and a total of four adaptations. Figure 5e shows the result of adaptation to the gradient of the combination of density and pressure in the ξ^2 direction (normal to the mean flow) only. Total number of adaptations was five, with $C_1 = 0.0$, and $C_2 = 0.9$, with geometric contributions to the control functions coming from the previous grid. The resulting pressure contours are shown in Fig. 5f.

Incompressible Flow over Backward Step

Results of multiple adaptations performed with INS3D are shown in Figs. 6 and 7. These results are for incompressible laminar flow over a two-block backward-facing step. Grid size for the first block is (35 × 21), and for the second block, (81 × 41). For purposes of comparison with experimental results, the Reynolds number chosen for the computation was 183. However, the grid length downstream of the step is only 30 step heights, while that in the experiment was much greater.

The initial grid is shown in Fig. 6a, while the grid following adaptations after 500 and 1000 time steps is shown in Fig. 6b. Adaptation was to the magnitude of the vorticity in the direction normal to the mean flow. Geometric contributions to the control functions were taken from the initial grid rather than the previous grid. Velocity profiles at the step and downstream (nearest to experimental sample points) are shown in Fig. 7.

Reattachment lengths for both computations are about $7.67H$, compared with the experimental value of $7.9H$, where H is the step height. While this case is clearly a difficult one to compute under any circumstances, the discrepancy in reattachment length may be partly attributable to the shorter recovery length allowed in the computation downstream of the step. The improvement in velocity profile prediction for the adaptive case is very slight, but would undoubtedly have been more dramatic if the grid had been coarser. However, there are clearly limits for grid resolution in which a given flow solver will remain stable. In some situations it appears that a grid which is sufficient to allow the solution to hang together is sufficient to resolve the solution properties of interest, so that adaptive gridding can play only a limited role.

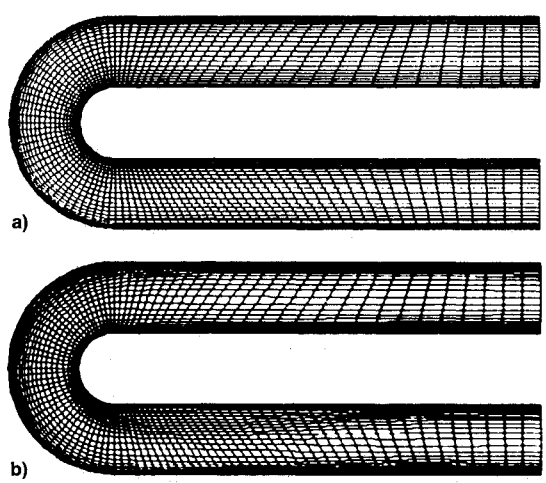


Fig. 8 Coarse grids with (111 × 31) points for computation of laminar flow at Reynolds number of 500 through a two-dimensional turnaround duct. A total of 6000 time steps were run on both static and adaptive grids: a) elliptic grid with orthogonal boundaries, and b) grid adapted four times to gradient of velocity magnitude in direction normal to the flow. The first adaptation was at 1000 time steps and the last at 2500.

However, the adaptive run here cost only about 1.2% more CPU time since only two adaptations were made.

Incompressible Flow in Two-Dimensional Turn-Around Duct

For this study, laminar flow was assumed at a Reynolds number of 500 through a two-dimensional duct with a 180-deg turn. A reference solution was obtained on a fixed grid with dimensions of (111 × 51) and off-wall spacing of $0.002H$, where H is the duct height. Both static and adaptive solutions were obtained for comparison on a coarse grid with only 31 points in the cross-stream coordinate, an off-wall spacing of $0.004H$, and 111 streamwise points. The coarse elliptic and adaptive grids are shown in Fig. 8. Boundary orthogonality was enforced through control functions on both grids.

Shown in Fig. 9 is the skin friction coefficients for both the inner and outer walls of the duct. Some improvement is seen

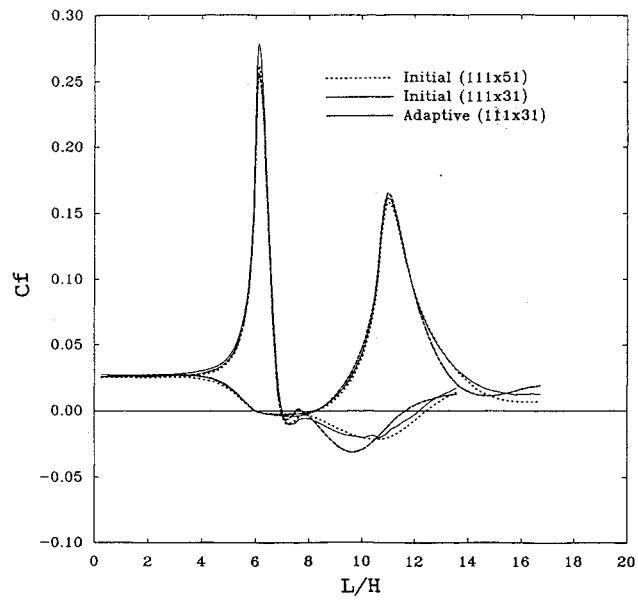


Fig. 9 Skin friction coefficient for fine-grid solution, coarse grid solution, and adaptive grid solution for flow through two-dimensional turn-around duct. The (111 × 31) grid was adapted four times to gradient of velocity magnitude.

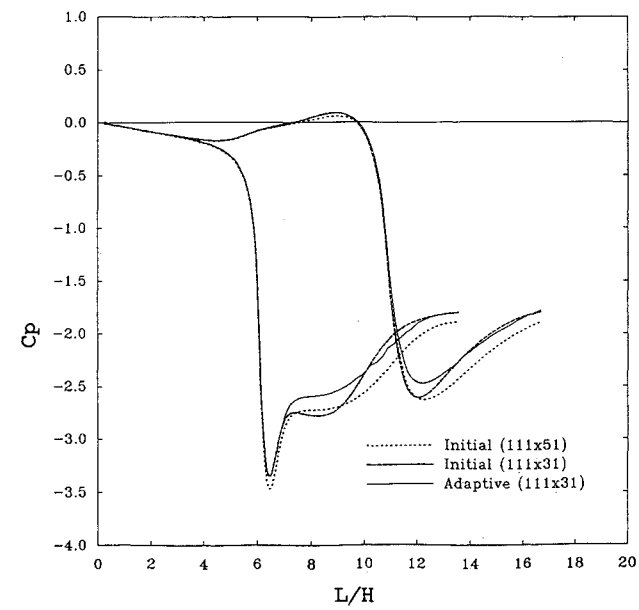


Fig. 10 Pressure coefficient for reference solution, coarse grid solution, and adaptive grid solution for laminar flow through a two-dimensional turn-around duct. The (111 × 31) grid was adapted four times to the gradient of velocity magnitude.

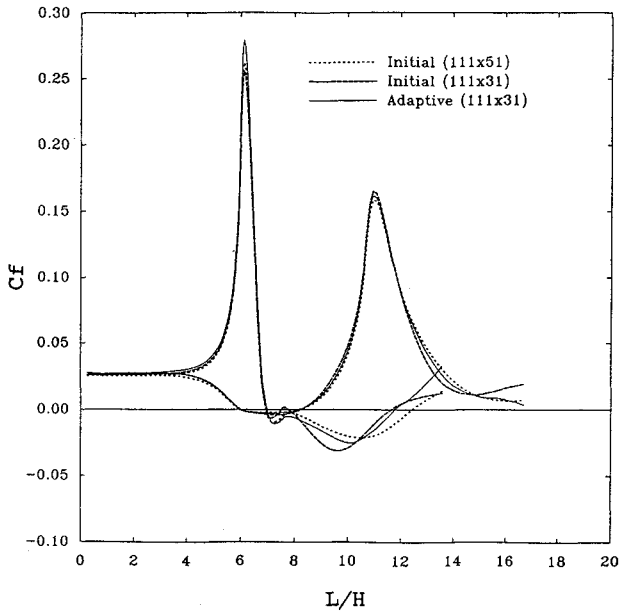


Fig. 11 Friction coefficient for fine grid solution, coarse grid solution, and adaptive grid solution for flow through two-dimensional turn-around duct. The (111 × 31) grid was adapted four times to the gradient of vorticity and grid aspect ratio combined.

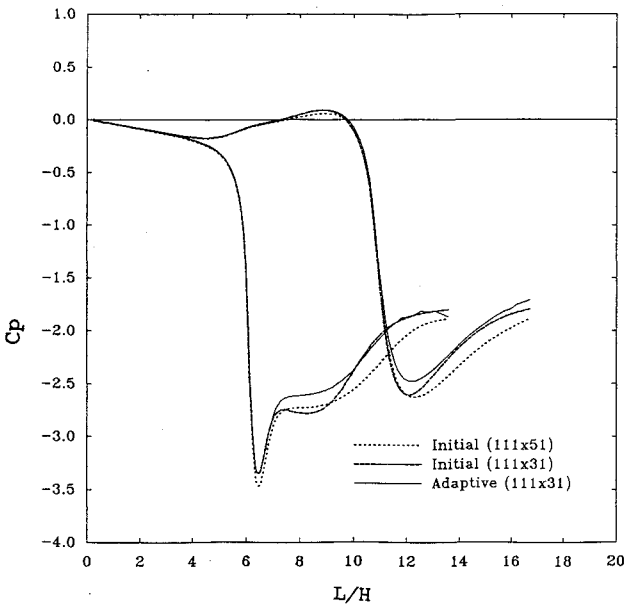


Fig. 12 Pressure coefficient for fine grid, coarse grid, and adaptive grid solutions for flow through two-dimensional turn-around duct. The (111 × 31) grid was adapted four times to the gradient of vorticity and grid aspect ratio combined.

in the adaptive results over the fixed grid results as compared with the reference solution. The separation point is the same for all three computations, but the adaptive results agree more closely on the reattachment point. However, the adaptive solution has little to offer in the prediction of surface pressure, as evidenced by Fig. 10. For this case, four adaptations to the gradient of the velocity magnitude were performed in the direction normal to the flow, with $C_1 = 0.1$ and $C_2 = 0.5$, and with control function updates applied to the initial values rather than to the previous values. The fine-grid solution required nearly twice the CPU time as that obtained on the coarse grid with four adaptations.

In Figs. 11 and 12 are results for adaptation to a combination of a flow variable and a grid quality measure for this case. Here adaptation is to the combination of vorticity and grid aspect ratio in the direction normal to the flow. Weight coefficients were the same as for the previous case.

The adapted grid skin friction coefficient for the outer wall is very nearly identical to that of the fine grid, as can be seen in Fig. 9a. Friction distribution for the inner wall, although smoother, departs significantly from the reference solution near the duct exit. This is attributable, at least in part, to the grid-dependence of the outflow numerical boundary condition which is a second-order extrapolation of flow variables. The discrepancy in solutions is even more pronounced in the surface pressure distribution, as depicted in Fig. 10b.

Conclusions

The widely used EAGLE grid generation system has been extended and enhanced so that it can be readily coupled with an existing PDE solver to provide a flexible adaptive grid capability. Since communication between the solver and the grid system is through external files, minimal changes are required in either computer program. The control function approach to grid adaptation used here enables the user to select from a number of weight functions of which to adapt the grid. A static adaptation capability allows sensitivity studies to be made so that intelligent choices can be made for weight functions and coefficients.

Results from inviscid compressible computations indicate that the adaptive grid system described is effective in providing finer shock wave resolution on grids much coarser than those ordinarily required. This results in significantly lower computational costs, even without dynamic grid terms in the discretized PDEs or interpolation between successive grids.

However, the results for incompressible viscous flows are less clear. Since it is usually necessary to preadapt viscous grids by clustering points in boundary regions, there remains little for the adaptive mechanism to do. Even in flows with large recirculating regions, the gradients in these areas are small compared to those in the boundary layer, and points are removed from the boundary layer at considerable peril. It is also known that separation and reattachment points tend to be influenced by the grid. So adequate resolution is required from the start of computations for these phenomena to evolve, even in laminar flow. The picture is even less clear in turbulent computations.

However, it has been clearly shown here that the control function approach is effective in moving grid points in response to weight functions specified by the user. It should be noted that the static grids used in this study were generated with the elliptic equation system of EAGLE, and already had boundary orthogonality and considerable smoothness. In other words, they were high-quality grids to start with, so that one might expect them to produce good solutions.

However, additional work is needed to identify the weight functions which are most effective for increasing the accuracy and decreasing the cost of flow solutions on nonoptimal grids. This is especially true for incompressible viscous flows, where the dramatic gradients posed by shock waves are not present.

Acknowledgments

This work was supported in part by NASA Marshall Space Flight Center, Contract NAS8-36949, Paul K. McConnaughey, Monitor, and in part by the Air Force Armament Directorate, Eglin AFB, Grant F08635-89-C-0209, Lawrence E. Lijewski, Monitor.

References

[1]Thompson, J. F., "A Survey of Dynamically-Adaptive Grids in the Numerical Solution of PDE," *Journal of Applied Numerical Mathematics*, Vol. 1, No. 1, 1985, pp. 3–27.

[2]Kim, H. J., and Thompson, J. F., "Three-Dimensional Adaptive Grid Generation on a Composite Block Grid," AIAA Paper 88-0311, Jan. 1988.

[3]Tu, Y., and Thompson, J. F., "Three Dimensional Solution-

Adaptive Grid Generation on Composite Configurations," AIAA Paper 90-0329, Jan. 1990.

[4]Thompson, J. F., "A Composite Grid Generation Code for General 3-D Regions—The EAGLE Code," *AIAA Journal*, Vol. 26, No. 3, 1988, p. 271.

[5]Rogers, S. E., Kwak, D., and Chang, J. L., "INS3D—An Incompressible Navier-Stokes Code in Generalized 3-D Coordinates," NASA TM 100012, Nov. 1987.

[6]Whitfield, D. L., "Implicit Upwind Finite Volume Scheme for the 3-D Euler Equations," MSSU-EIRS-ASE-85-1, Mississippi State Univ., Mississippi State, MS, Sept. 1985.

[7]Warsi, Z. U. A., "Basic Differential Models for Coordinate Generation," *Numerical Grid Generation*, edited by J. F. Thompson, North-Holland, New York, 1982, pp. 41–77.

[8]Saltzman, J. S., and Brackbill, J. U., "Applications and Generalizations of Variational Methods for Generating Adaptive Mesh," *Numerical Grid Generation*, edited by J. F. Thompson, North-Holland, New York, 1982, pp. 865–884.

[9]Roache, P. J., and Steinberg, S., "A New Approach to Grid Generation Using a Variational Formulation," AIAA Paper 85-1527, June 1985.

[10]Warsi, Z. U. A., and Thompson, J. F., "Application of Variational Methods in Fixed and Adaptive Grid Generation," *Computers and Mathematical Applications*, Vol. 19, Nos. 8 and 9, 1990, pp. 31–41.

[11]Kennon, S. R., and Dulikravich, G. S., "A Posteriori Optimization of Computational Grids," AIAA Paper 85-0483, Jan. 1985.

[12]Carcaillet, R., "Optimization of 3-D Computational Grids and Generation of Flow Adaptive Computational Grids," AIAA Paper 86-0156, Jan. 1986.

[13]Soni, B. K., and Mastin, C. W., "Variational Methods for Grid Optimization," *Journal of Applied Mathematics and Computers* (to be published).

[14]Anderson, D. A., "Equidistribution Schemes, Poisson Generators, and Adaptive Grids," *Applied Mathematics and Computation*, Vol. 24, No. 3, 1987, pp. 211–227.

[15]Anderson, D. A., "Generating Adaptive Grids with a Conventional Grid Scheme," AIAA Paper 86-0427, Jan. 1986.

[16]Johnson, B. F., and Thompson, J. F., "Discussion of a Depth-Dependent Adaptive Grid Generator for Use in Computational Hydraulics," *Numerical Grid Generation in Computational Fluid Mechanics*, edited by J. Hauser and C. Taylor, Pineridge Press International, Swansea, Wales, UK, 1986, pp. 629–640.

[17]Thompson, J. F., Unpublished Research, Coastal Engineering Research Center, U.S. Army Engineering Waterways Experiments Station, Vicksburg, MS, 1986.

[18]Eiseman, P. R., "Adaptive Grid Generation," *Computer Methods in Applied Mechanics and Engineering*, Vol. 64, Nos. 1–3, 1987, pp. 321–376.

[19]Kerlike, G. D., and Klopfer, G. H., "Assessing the Quality of Curvilinear Coordinate Meshes by Decomposing the Jacobian Matrix," *Numerical Grid Generation*, edited by J. F. Thompson, North-Holland, New York, 1982, pp. 787–807.

[20]Gatlin, B., Thompson, G. F., Yeon, Y. F., Luong, P., Ganapathiraju, D., and Walverton, M. K., "Extensions to the EAGLE Grid Code for Quality Control and Efficiency," AIAA Paper 90-0148, Jan. 1991.

[21]Luong, P. V., "Analysis and Control of Grid Quality in Computational Simulation," Ph.D. Dissertation, Mississippi State Univ., Mississippi State, MS, Dec. 1991.