A SOLUTION ADAPTIVE TECHNIQUE USING TETRAHEDRAL UNSTRUCTURED GRIDS

Shahyar Z. Pirzadeh
NASA Langley Research Center
Hampton, Virginia, U.S.A.

Keywords: *unstructured grids, grid adaptation, remeshing*

Abstract

An adaptive unstructured grid refinement technique has been developed and successfully applied to several three dimensional inviscid flow test cases. The method is based on a combination of surface mesh subdivision and local remeshing of the volume grid. Simple functions of flow quantities are employed to detect dominant features of the flowfield. The method is designed for modular coupling with various error/feature analyzers and flow solvers. Several steady-state, inviscid flow test cases are presented to demonstrate the applicability of the method for solving practical three-dimensional problems. In all cases, accurate solutions featuring complex, nonlinear flow phenomena such as shock waves and vortices have been generated automatically and efficiently.

1 Introduction

Generation of efficient grids for the computational fluid dynamics (CFD) applications usually requires some prior knowledge of the flow behavior in order to match the grid resolution to the essential features of the problem. While such information is usually unavailable in advance, a number of 'trial-and-error' iterations between the solution and grid generation are often required to tailor the grid to the specific nature of the problem at hand. Alternatively, an overly fine grid is often generated to guarantee the desired solution accuracy. In both cases, the amount of time, effort, and computational resources may become excessively large for solving complex problems.

The solution adaptive grid technology is a powerful tool in CFD, which provides three important benefits: automation, improved efficiency, and increased solution accuracy. Since the distribution of grid points are efficiently determined by an automatic process, an adapted grid contains far fewer number of points than an initial fine grid having similar local resolution at the crucial regions. This important feature of grid adaptation results in a substantial saving in computational time and memory requirement.

In general, most adaptive methods fall into three broad categories: grid movement (r-refinement), grid enrichment (h-refinement), and local solution enhancement (p-refinement). While the methods in the first two classes modify the grid density to improve the solution accuracy (grid adaptation), those under the third category enhance the order of numerical approximation at locations where the solution undergoes abrupt variations (solution adaptation). Most adaptive techniques used in the CFD applications fall into the first two classes.

In the grid movement approach, nodes are redistributed and moved towards regions where a higher degree of accuracy is needed. Since the grid topology remains unchanged throughout the grid adaptation, the process of grid movement

Copyright © 2000 by the National Aeronautics and Space Administration. Published by the International Council of the Aeronautical Sciences, with permission.

can be simply incorporated into the solver in a modular fashion. In addition, no data transfer (i.e., interpolation among different grids) is required since the grid structure remains intact during the process. Therefore, no solution accuracy is lost from one adaptation cycle to the next. The method is especially advantageous for transient problems involving moving surfaces and unsteady solutions. However, since the number of grid nodes remains constant, transferring nodes from one part of the grid to another may cause local 'depletion' of grid elements, and thus severe distortion of the grid may be introduced [1]. Adaptation by grid movement has primarily been applied to structured grids and 2D triangular meshes.

In the grid enrichment technique, more nodes are added to regions where higher accuracy of the solution is desired. Nodes can also be removed from locations where the solution is smooth and requires less grid resolution. Due to node addition or deletion, the topology (connectivity) of the grid changes from one adaptation cycle to another. Consequently, interpolation of data between consecutive grids is required which curtails the applicability of these methods for unsteady problems. Adaptive methods by grid enrichment are particularly attractive for their flexibility, especially when applied in conjunction with unstructured grids.

Among the adaptive grid methods by enrichment, two techniques are notable: grid subdivision and grid remeshing. In grid subdivision, 'parent' cells are divided into several smaller cells. The method is efficient and fast. Once a systematic data structure is established prior to the adaptation cycles, both grid refinement and coarsening can easily be implemented. The grid subdivision methods have been best demonstrated on Cartesian meshes [2] and can be implemented in triangular grids conveniently. However, their application to tetrahedral grids involves complex data structures and, in most cases, results in refinement complications and grid distortion [3].

Global and partial remeshing have also been employed for adaptive grid refinement successfully [4,5]. Two significant advantages of these methods are 1) flexibility for refinement and unconstrained coarsening (in subdivision methods, for example, grids cannot be coarsened beyond their initial resolutions) and 2) good quality grids generated in each refinement cycle. On the other hand, the grid generation time and the cost of solution interpolation are extensive in these methods, especially in the global remeshing.

As there is no single grid type or generation method (e.g., structured, unstructured, etc.) to fit all classes of CFD problems, neither is an individual adaptive methodology which can be universally applied to a variety of problems. Different methods offer certain advantages to different classes of grids and problems [2,5]. Therefore, it is beneficial to exploit the advantages of several techniques in a hybrid adaptive grid method for solving complex problems [1]. In the present work, an attempt has been made to combine the efficiency of h-refinement and the flexibility of remeshing for solution adaptive refinement. A grid movement technique has also been developed (not presented in this paper) for geometric adaptation of volume grids to moving or deforming surfaces (see Ref. 6). The focus of this paper is on the refinement mechanism aspect of the solution adaptive problem as applied to realistic 3D problems.

2 Approach

The proposed grid adaptation strategy is summarized in the flowchart shown in Fig. 1. In this chart, each block represents an independent module readily exchangeable in the system. Starting with a reasonably coarse mesh and a corresponding flow solution, the grid adaptation process proceeds with an assessment (analysis)

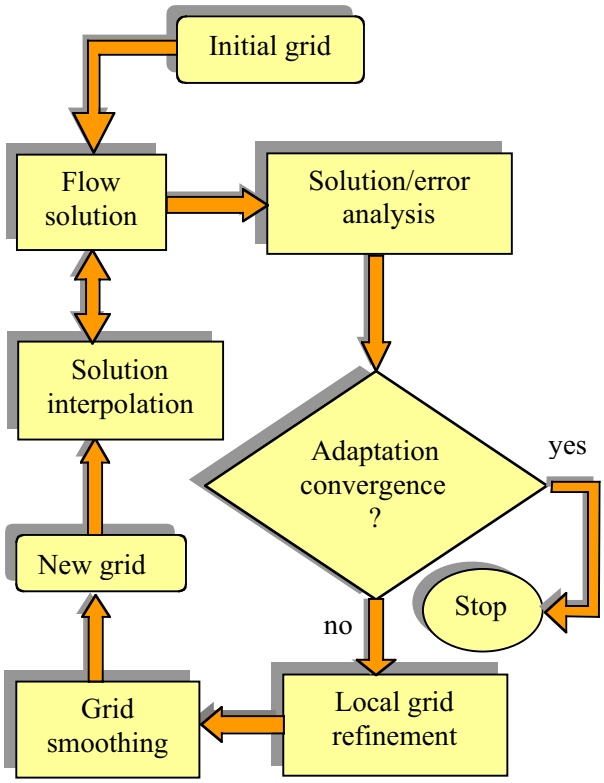

Figure 1. Flowchart of the proposed grid adaptation strategy.

of the current flow solution to determine where in the field the solution needs further improvement. For a successful adaptation, the initial coarse grid should be adequately resolved in order to prevent the adaptive solution from converging to a 'wrong' solution. Once the locations requiring solution improvement are identified, the corresponding grid elements are flagged for resolution enhancement. A local remeshing strategy is then employed to refine the grid at these locations. A new solution is next obtained on the modified mesh followed by another solution assessment. The process continues until a preset goal such as a desired level of solution accuracy, the optimization of an objective function, or simply a certain number of adaptation cycles is accomplished.

There are two main components in any grid adaptation technique. First, a strategy is employed to determine where in the field the grid (solution) needs modification, e.g., by means of error estimation, flow feature detection, or any other type of solution analysis. Secondly, a mechanism is utilized to either change the grid density or modify the solution method. The focus of this paper is mainly on the latter, i.e. a grid alteration procedure that is automatically controlled by the flow solution characteristics. Although it is not the objective of this paper to elaborate on the former topic, which is the subject of a separate paper, a brief discussion of error estimation and flow feature detection is presented for completeness.

2.1 Error and Flow Feature Detection

In most grid adaptation techniques, the question of where to modify the grid resolution or the solution accuracy is addressed through the concept of 'error equidistribution'. This principle states that grid nodes should be clustered in such a way that the computational errors are uniformly distributed throughout the grid. In other words, the grid should be proportionally denser where the solution incurs larger error, e.g., where the flow undergoes rapid changes, and vice versa. The principle of error equidistribution is strictly applied in the methods by r-refinement and the global remeshing techniques to redistribute grid points in the field optimally. The magnitude of the computed errors directly determines the grid spacing parameters in these methods.

In the methods based on h-refinement, however, error estimation practically serves as a means to locate the grid elements experiencing large computational errors [7]. A separate mechanism modifies the distribution of grid nodes at these locations without considering the magnitude of errors. Unlike the r-refinement which aims for an optimum grid with equal errors at the nodes, the h-refinement fulfills the objective of reducing the maximum error through several preset refinement steps. Theoretically, as the number of h-refinement cycles increases, the distribution of errors approaches an equilibrium state throughout the grid. In practice, however,

only a few cycles of refinement are usually performed to adapt the grid. Therefore, the role of error estimation for h-refinement reduces to indication of computational errors induced by the dominant flow features.

Many error (feature) indicators in use are based on some physical flow quantities such as density, pressure, entropy, etc. Functions of the first or second gradients of these quantities are usually employed to estimate errors or detect dominant flow characteristics. For h-refinement, even a crude indicator such as a simple increment of a flow quantity is sufficient as long as it correctly detects the desired flow feature. In this work, the following simple indicator is employed to detect expansion and compression waves.

$$\vartheta_i = (1 + \delta_i / \delta_a) \, |\Delta p_i| / p_i \qquad (1)$$

where p_i and Δp_i are the local static pressure and its increment associated with the i^{th} grid element, respectively, δ_i is the local grid spacing, and δ_a is an average spacing in the grid. The inclusion of the grid-spacing correction factor in Eq. 1 results in a better detection of weak flow discontinuities in larger grid cells that are away from the geometry.

Functions based on vorticity or entropy have been used as indicators for detecting vortices and adapting grids to vortical flows. In this work, the following simple measure of entropy is used to capture vortices.

$$\varepsilon_i = (\gamma \, p_i / \rho_i^{\gamma}) - 1 \qquad (2)$$

where ρ_i is the local density and γ is the specific heat ratio. It should be emphasized that the parameters defined by Equations (1) and (2) do not represent the magnitude of computational errors but only indicate the location of dominant flow features inducing errors. In practice, the flow feature indicators measured at each grid element are compared with some threshold constants prescribed by the user. If the value of an indicator is greater than the threshold, the corresponding grid element is flagged for refinement.

The challenge in the practical implementation of an adaptive method for solving complex problems is the choice of appropriate error or indicating functions. While a particular indicator may work well for certain class of flow features, it may not be as effective in recognizing other flow phenomena. Usually, a prior knowledge of the flow characteristics is needed in order to select relevant functions. Since the information about the flow is generally not available in advance, an automatic indicator based on a global objective (e.g., drag reduction) is desirable. Such a universal indicator should be able to capture all relevant flow features that influence the objective function (e.g., shock waves, vortices, etc.) and should even determine the flow characteristics that contribute to the formation of these flow features. In addition, the indicator must be 'smart' enough to distinguish between the actual flow variations and the numerical 'noise' present in the solution. Otherwise, the grid may be refined in the wrong locations. The development of an automatic universal indicator requires comprehensive research, which is beyond the scope of the present project and this paper. Further in-depth study of the subject is planned for future work. Once such a capability becomes available, it can be readily incorporated into the present modular adaptation system.

2.2 Adaptive Local Mesh Refinement

The inherent irregularity of unstructured grids offers two important benefits: 1) high degree of flexibility to handle complex shapes and 2) ease of mesh alteration. The lack of a regular structure in tetrahedral grids provides arbitrary cell groupings which, in effect, makes every part of a grid independent of the rest. Consequently,

any section of a tetrahedral grid can be removed and locally remeshed without disturbing the rest of the grid. Furthermore, the local resolution of a grid can be arbitrarily changed when the grid is partially remeshed. This important property makes unstructured grids particularly suitable for adaptive local remeshing.

In this work, an unstructured tetrahedral grid generation system, VGRIDns [8], is used to generate initial grids. The grid generation method is based on the Advancing-Front [9] and the Advancing-Layers [8] techniques. Both techniques are based on marching processes in which tetrahedral cells grow on an initial front (triangular boundary mesh) and gradually accumulate in the field around the subject geometry. The front, made of the exposed triangular faces of the tetrahedrons, continuously evolves and marches outward as new cells are created and added in the computational domain. The process continues until the entire domain is filled with tetrahedral cells when no triangular face remains in the grid. The grid characteristics, used during the marching process, are prescribed through a set of source elements included in a 'transparent' Cartesian background-grid [10]. The information is first distributed smoothly from the sources onto the background grid nodes by solving a Poisson equation and then interpolated in the field to distribute unstructured grid points during the marching process.

An important feature of the advancing front technique, like any other marching method, is that the solution process can be restarted at any time. Since a grid segment, once constructed, does not influence the rest of the mesh yet to be generated, the process can be stopped and restarted without "carrying" the grid portion already generated in the previous run(s). The only data required to restart the generation process are those defining the current front. The flexibility of unstructured grids for local remeshing, along with the restart feature of the advancing front method, offers an excellent opportunity for grid adaptation. An efficient grid restart capability and a local remeshing technique have previously been developed and incorporated into the VGRIDns system for post-processing of the generated grids [11]. In this work, the existing capabilities are extended for adaptive grid refinement.

The process of local grid refinement is demonstrated on a simple triangular grid in Fig. 2. In this example, a transonic flow field around a simple airfoil is assumed. An initial coarse grid (Fig. 2a), along with a corresponding flow solution, is supplied to the adaptive refinement scheme. An appropriate flow analyzer, such as that given by Eq. (1), is used to detect the dominant flow features. For example, a diffused shock wave and a large pressure gradient at the leading edge of the airfoil are assumed in this case. The grid cells experiencing large variations in the flow properties, along with additional layers of cells, are identified for removal (shaded triangles in Fig. 2b). In the next step, the flagged elements are deleted to create voids (empty pockets) in the mesh (Fig. 2c). The remaining grid points and cells are then renumbered, and the faces exposed in the pockets are grouped to form a new front in the grid. If any portion of the geometry is exposed in the voids, the corresponding faces on the surface are h-refined, and the newly inserted nodes are projected onto the geometry model. The rest of the front faces in the field remain unrefined to maintain a contiguous connectivity between the elements of the new grid segment (being generated in the pockets) and the original grid. Finally, the grid density is readjusted in the pockets, and the voids are remeshed by the Advancing-Front method as in a regular restart run (Fig. 2d).

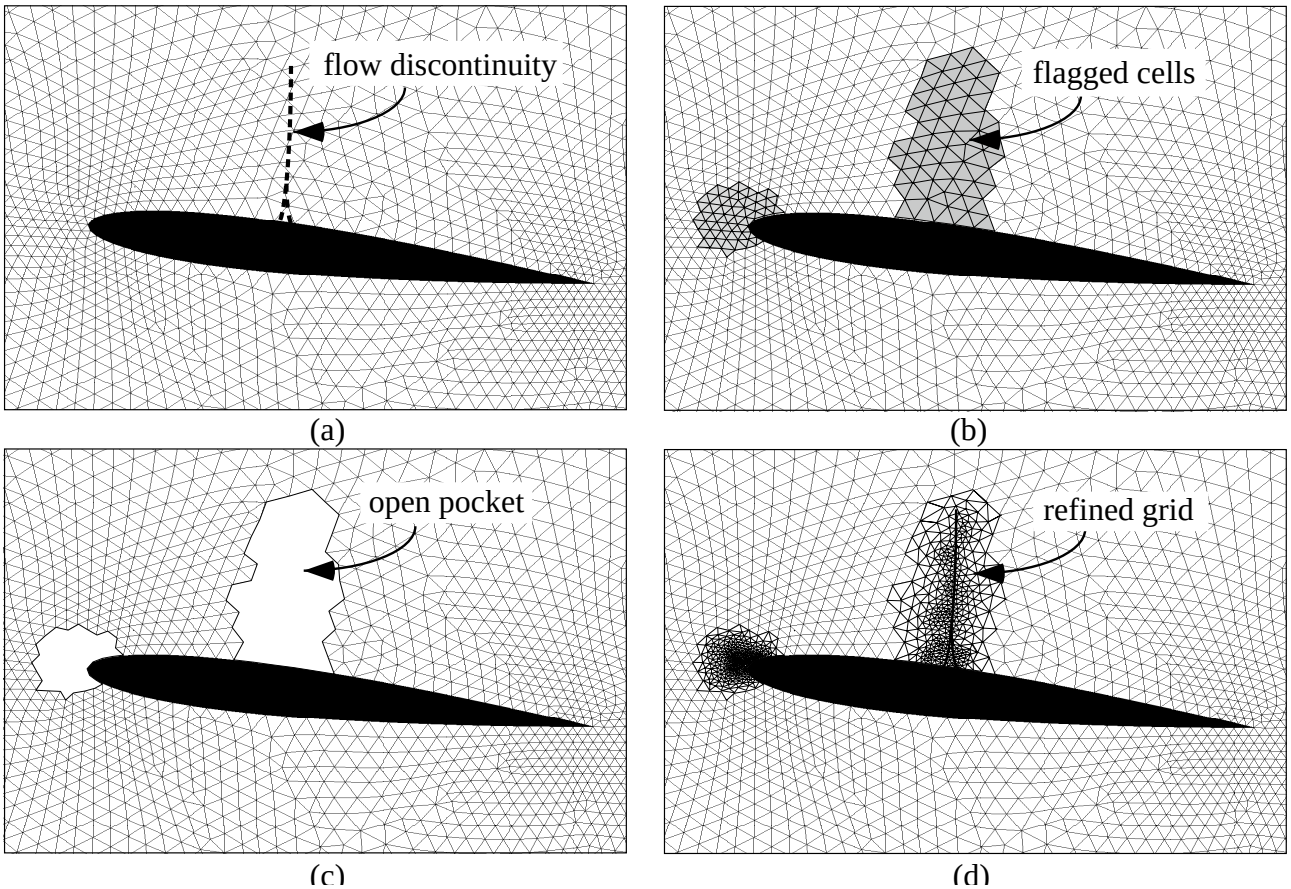

Figure 2. Adaptive refinement steps by local remeshing: (a) initial grid, (b) flagged cells in regions of rapid flow gradients, (c) removal of the flagged cells, and (d) locally refined grid.

The process of local h-refinement is depicted in Fig. 3 for a portion of a 3D triangular surface mesh. In this figure, the shaded surface triangles are assumed covered with tetrahedral volume grid (Figs. 3a and b). The unshaded area represents a portion of the surface mesh exposed in a void after a segment of the volume grid is removed at the location of a shock wave (Fig. 3c). To refine the exposed triangles, new grid points are first added to the edges of the triangles (Fig. 3d). Each interior triangle is then divided into four smaller triangles by connecting the new nodes (Fig. 3e). The "buffer" triangles (those adjacent to the unrefined region) are divided into two or three triangles, depending on their number of edges on the pocket boundary. Finally, the void is remeshed and filled with smaller tetrahedrons (Fig. 3f).

Since the length of each surface mesh edge is cut in half by the h-refinement, the spacing parameters defined by the background grid are also reduced by 50% for regenerating the volume grid in the voids. Therefore, every time a pocket is opened for remeshing, the newly generated grid portion becomes twice as fine as the surrounding parent grid. The modified grid spacing provides the required compatibility between the h-refined surface and locally remeshed volume grids. To ensure a smooth variation of grid resolution between the refined cells and the surrounding parent grid, an average grid spacing is employed to generate the first layer of tetrahedral cells on the pocket walls.

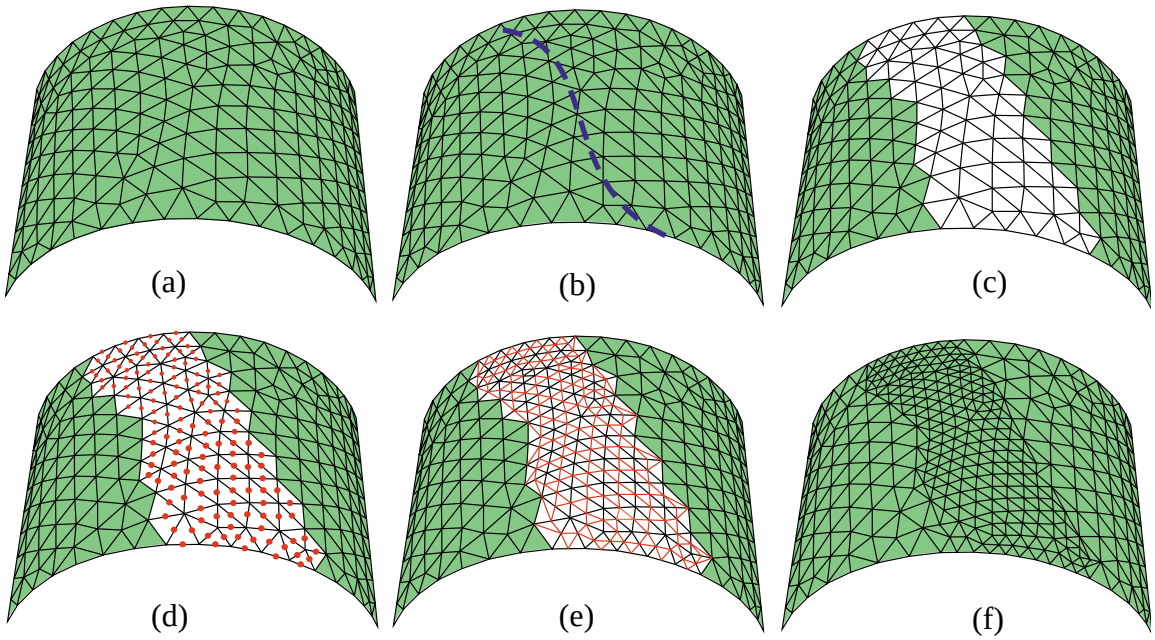

Figure 3. Process of local surface mesh subdivision in three-dimension: (a) initial coarse grid, (b) footprint of flow discontinuity on the surface, (c) exposed triangles after a portion of the volume grid is removed, (d) insertion of new points on the edges of exposed triangles, (e) subdivision of the exposed triangles, and (f) final adapted grid.

The average spacing is based on the actual size of the interior front faces (those which are not h-refined) and the modified background grid spacing. These "buffer" cells provide a gradual transition from the coarse to the fine cells generated in the pockets as shown in Fig. 2d.

3. Results

Adaptive solutions are presented in this paper for three steady-state, 3D test cases: 1) an ONERA M6 wing at transonic speed, 2) an experimental high performance fighter aircraft at subsonic speed, and 3) an experimental aerospace vehicle at supersonic speed. Each of these flow cases represents a distinct aerodynamic feature suitable for the adaptive solution. The examples clearly demonstrate the three benefits that grid adaptation provides, i.e. accuracy, automation, and efficiency. All inviscid flow computations, presented in this paper, were performed using the upwind, cell-centered, finite-volume, unstructured grid solver USM3D [12].

3.1 ONERA M6 Wing

An ONERA M6 wing configuration has been employed to demonstrate the transonic shock capturing capability of the present solution adaptive grid method. The flow condition is at a free stream Mach number of 0.84 and an incidence of 3.06 degree.

A reasonably coarse grid with a nearly uniform point distribution chordwise was generated to serve as the initial grid for adaptation. The grid, shown in Fig. 4a, contains 2,615 boundary nodes, 15,432 total nodes, and 83,356 tetrahedral cells. An inviscid flow computation on this grid reveals the presence of a weak "λ" shock wave on the upper surface of the wing. The surface pressure contours are also illustrated in Fig. 4a. As expected, the flow is under-expanded on the upper surface at the leading edge, and the shock wave is diffused due to coarseness of the grid.

Using the surface grid subdivision and local remeshing procedures described earlier, three levels of adaptive refinement were performed for this case. The indicator given by Eq. (1) was employed to detect regions of large pressure variation. In each refinement cycle, cells indicating 20% or larger increment in the pressure parameters were deleted and remeshed. The final grid contains 9,739 boundary nodes, 54,385 total nodes, and 288,739 tetrahedrons. Figure 4b shows the adapted surface grid and the corresponding pressure contours. As evident, the grid is efficiently refined at the shock location and the leading edge of the wing at the so-called "suction peak" region. The effect of the automatic grid refinement is clearly reflected on the surface pressure contours, which show a sharp shock definition including minute details of the pressure gradient at the wing tip.

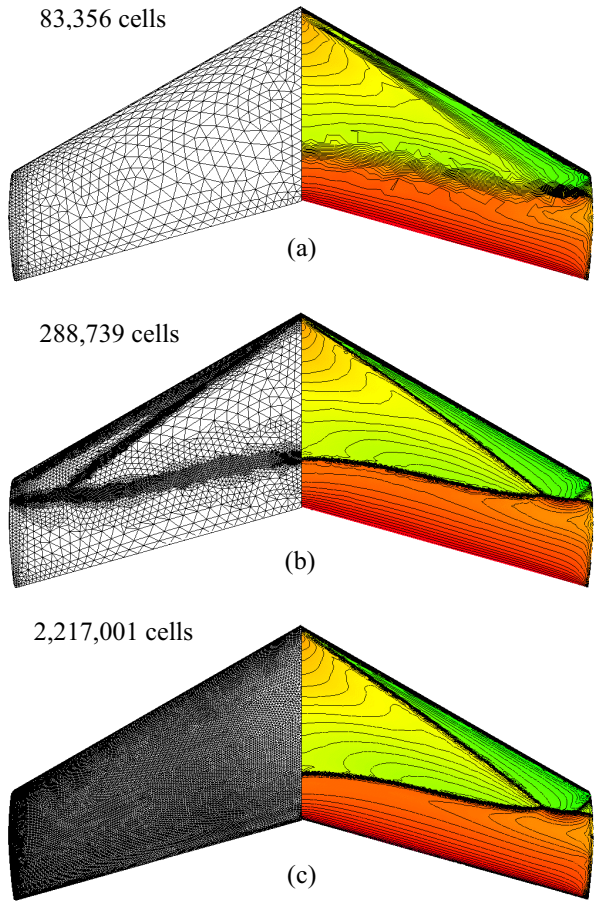

Figure 4. ONERA M6 wing surface grids and pressure contours at M_∞=0.84 and α=3.06°: (a) initial, (b) adapted, and (c) fine unadapted.

To investigate the effect of partial grid refinement on accuracy of the adapted solutions, a globally fine grid with a resolution similar to that of the adaptively refined grid was generated. This large grid contains 40,424 boundary nodes, 394,155 total nodes, and 2,217,001 tetrahedrons. The surface grid and the corresponding pressure contours are shown in Fig. 4c. A comparison of this solution with that of the adapted grid indicates that the differences between the two are negligible, and that the grid adaptation has produced an identical result with about an order of magnitude smaller grid size. Figure 5 illustrates several chordwise distributions of the surface pressure coefficient (Cp) for the initial coarse, adapted, and unadapted fine grids as compared with the experimental data at six different spanwise stations. As expected, there are insignificant differences between the adapted and the fine grid results. From the Cp distributions, it appears that the result of the coarse grid is in satisfactory agreement with the experimental data at the shock locations. However, it is well known that inviscid solutions predict stronger shocks further downstream as indicated by both the fine and adapted grid curves in Fig. 5. Addition of viscous effects usually weakens and moves shock waves upstream to the correct locations. As illustrated in Fig. 5, the automatic refinement of the grid at the leading edge has corrected the solution at the suction peak area. Also, note that both segments of the λ shock wave are captured by the adaptive and the fine grid solutions at the span station y/b=0.8. Accurate computation of the flow at this particular station is difficult due to its proximity to the coalescing shock waves.

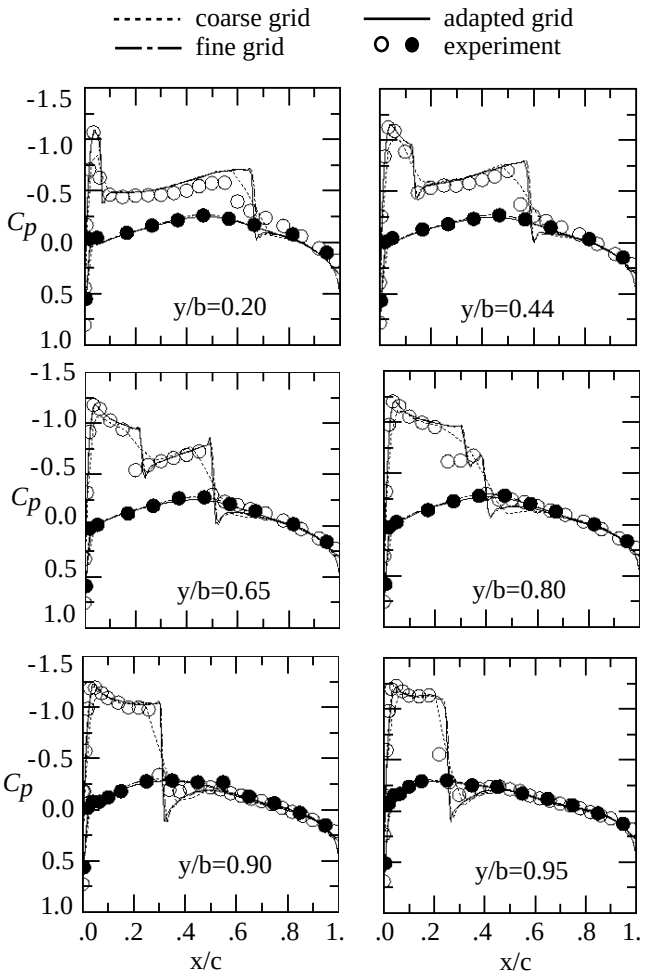

Figure 5. Chordwise distributions of surface pressure coefficient for ONERA M6 wing at M_∞=0.84 and α=3.06°.

As mentioned earlier, a salient feature of the adaptive grid methods by remeshing is the quality of the refined grids they produce. Every time a portion of the grid is removed for local remeshing, new cells are regenerated in the pockets without affecting the quality of grid elsewhere (unlike adaptation by grid movement.) The accuracy of the presented adaptive flow solutions and their ease of generation substantiate the viability and quality of the adapted grids generated with the present method. Furthermore, this example underscores the advantage of grid adaptation for providing more accurate flow solutions economically.

The initial grid/solution as well as the adapted results were all generated using a Silicon Graphics Octane workstation with a R10000 processor. While the mesh for the fine grid was also generated on the same workstation, the corresponding flow computation was performed on a CRAY C90 supercomputer due to its large memory requirement. A converged solution on the fine grid took about 36,548 CPU seconds on the CRAY C90. A sum of 40,335 CPU seconds of the SGI workstation was spent to obtain the initial as well as three levels of adaptive solutions. For the cases presented in this paper, all adapted solutions were started from the freestream condition at each adaptation cycle. Interpolation of solutions from one grid onto the next adapted grid (planned for future work) would expedite the overall solution convergence, resulting in a substantial saving in the adaptive solution time.

3.2 Modular Transonic Vortex Interaction Configuration

To demonstrate the effectiveness of the present solution adaptive method for predicting vortical flows, a generic fighter model referred to as the Modular Transonic Vortex Interaction (MTVI) has been employed. The configuration features a chine forebody with an included angle of 30 degrees, sixty-degree cropped delta wings, partially deflected wing leading-edge flaps, and twin vertical tails. All edges of the geometry are sharp inducing flow separations and vortices, which are independent of viscous effects. Inviscid solutions were obtained at a free stream Mach number of 0.4 and an incidence of 20 degrees.

An initial grid generated for this geometry contains 31,565 nodes and 163,619 tetrahedrons. As in the previous case, the grid density is marginally adequate to resolve the main features of the flow. No attempt has been made to cluster grid points at locations where vortices are

expected. Three levels of adaptation refine the grid at the critical locations, producing a final size of 108,014 nodes and 564,727 cells. Figure 6a illustrates the surface triangulation for the initial coarse grid (port) and after adaptation (starboard). Cross-sections of the initial and adapted volume grids are shown at a streamwise station ahead of the vertical tails in Fig. 6b. The automatic refinement of the surface and volume grids, as adapted to the chine and wing vortices, is clearly indicated in these figures.

The accurate flow solution obtained in this example highlights the benefit of increased automation provided by grid adaptation. The solution has not only predicted all the details of the vortical flow structure accurately, it has even revealed the onset of a vortex breakdown phenomenon at the proper location automatically. Figure 7 shows local refinement of the volume grid (open pockets) at the vortex locations at two different stages of adaptation. A refinement of the initial grid, triggered by the first solution, indicates a chine vortex extending beyond the aircraft tail (Fig. 7a). The final refined grid correctly predicts a chine vortex burst ahead of the vertical tail as indicated in Fig. 7b. The vortex breakdown phenomenon has also been observed on this geometry experimentally. Figure 8 depicts a wind tunnel visualization of the flow at a slightly different condition ($\alpha=30^{o}$, undeflected flaps). Similar formation of the chine vortex and its burst in front of the vertical tail is clearly visible in this picture. The importance of the automation

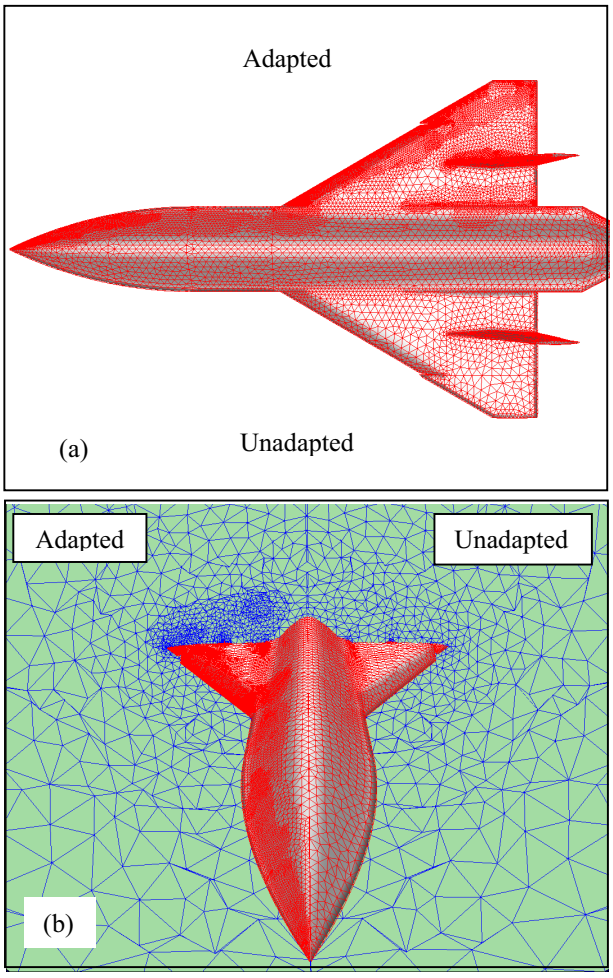

Figure 6. Initial and adapted unstructured grids on the MTVI configuration: (a) surface mesh and (b) surface/volume grid.

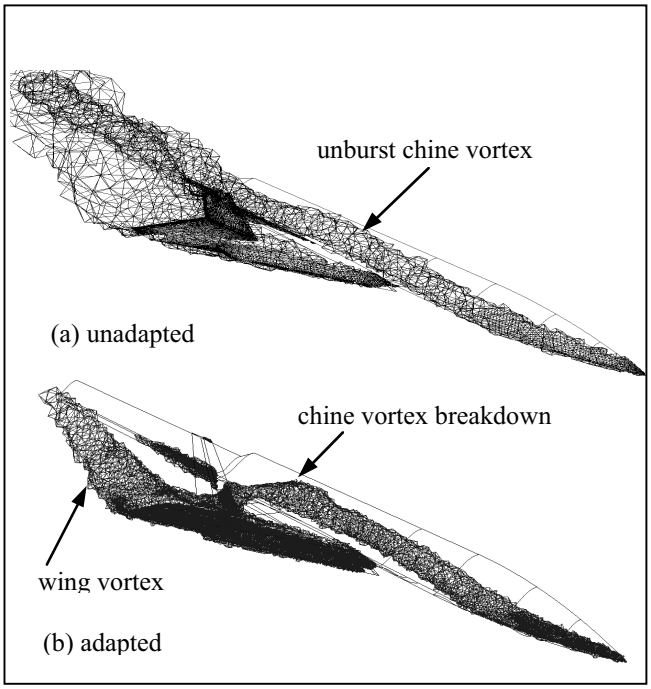

Figure 7. Local refinement of the MTVI volume grid: (a) initial grid and (b) adapted grid indicating the chine vortex burst.

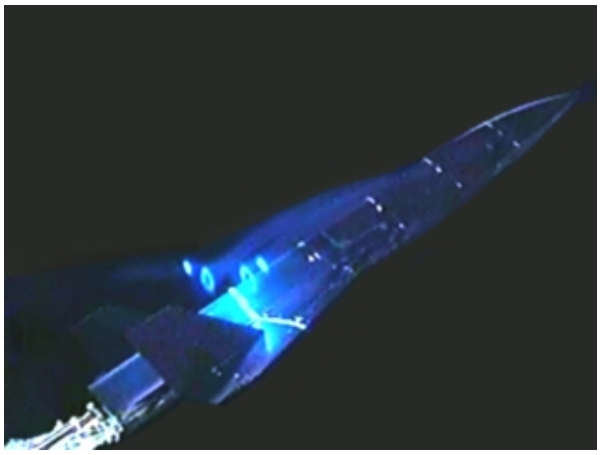

Figure 8. Wind tunnel visualization of flow around the MTVI configuration showing chine vortex burst.

aspect of grid adaptation for producing accurate solutions is well demonstrated in this example, as the initial unadapted grid actually yields a misleading solution to the problem.

Figure 9a compares surface pressure distributions before and after grid adaptation on the port and starboard sides of the aircraft, respectively. The adapted solution indicates crisper footprints of the wing vortices and a chine vortex, which does not extend as far downstream as that of the unadapted solution (indicative of the vortex burst phenomenon.) The pressure distributions in the field, showing the vortices in a cross-sectional plane, and on the surface are portrayed in Fig. 9b. As evident, a well-defined vortex generated by the sharp leading-edge of the deflected flap and even a smaller vortex emanating from the wing snag have been captured with grid adaptation (Fig. 9b, starboard). Figure 9b also shows a chine vortex in the field as predicted by the unadapted solution incorrectly (port). The absence of the chine vortex in the starboard side of the image indicates the breakdown phenomenon captured by the adapted solution at this location.

To detect vortices, the feature indicator given by Eq. (2) was employed in this

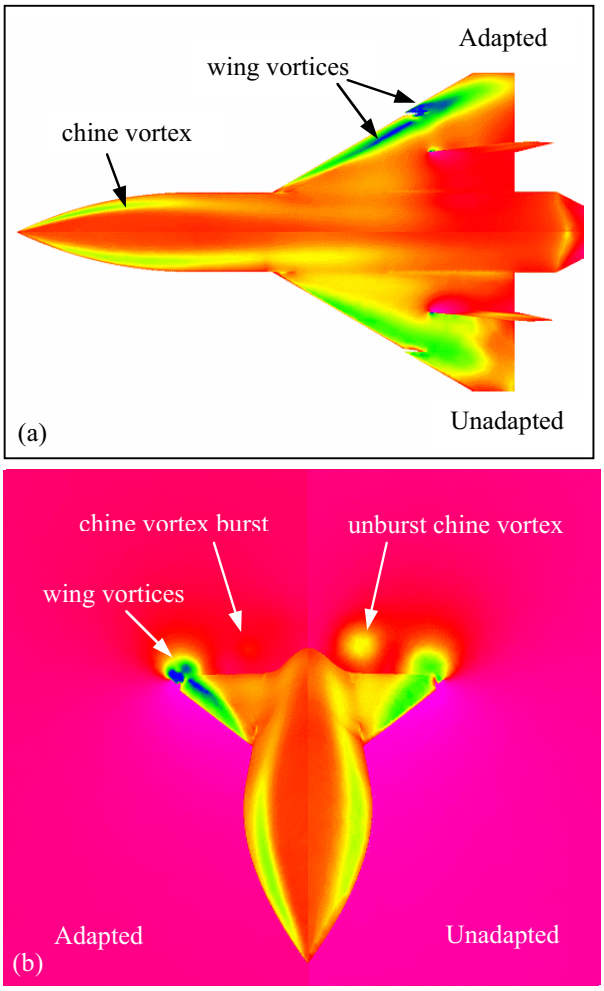

Figure 9. Initial and adapted static pressure distributions on the MTVI configuration: (a) surface and (b) surface/volume.

example. Grid cells experiencing entropy production levels of higher than a threshold (a small fraction of the maximum entropy produced in the field) were flagged for removal at each adaptation cycle. In the present example, a threshold value of 0.01 has been used. All the initial and adaptive computations (grids/solutions) were performed using the SGI workstation described earlier.

3.3 X-38 Forebody Configuration

The last test case is presented to emphasize the efficiency of the adaptive method as demonstrated on a supersonic flow featuring a

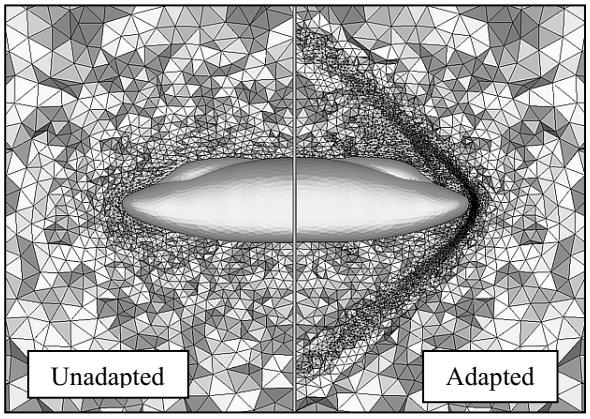

Figure 10. Initial and adapted tetrahedral grids around the X-38 forebody configuration.

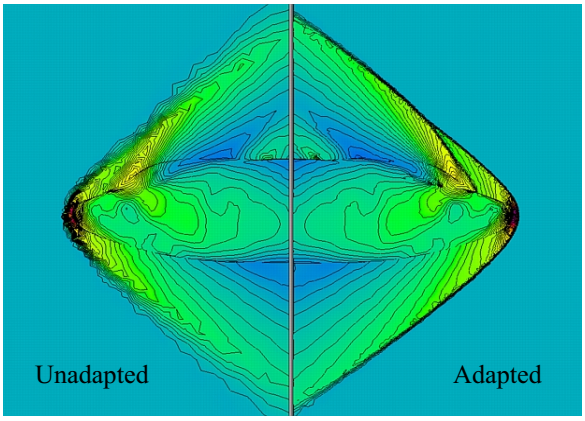

Figure 11. Comparison of the initial and adapted static pressure contours on the X-38 forebody configuration.

detached shock wave. The configuration selected for this purpose is the front portion of an experimental aerospace vehicle referred to as X-38. The flow condition is at Mach 2 and a zero incidence angle. This case represents a classic example for which the generation of an efficient unadapted grid is challenging due to the presence of a conical detached shock wave extending far into the field. Even with a prior knowledge of the shock location, it is difficult to control the distribution of grid points at a curved surface in the 3D space. Usually, the generated grids are either too coarse away from the geometry, which fail to capture flow discontinuities accurately, or globally too fine, which make the computational cost excessive. The economic advantage of solution adaptive gridding becomes more tangible for such applications.

In Fig. 10, two separate cross-sections of the field grids are illustrated around the geometry. The initial grid, shown on the left-hand side of the figure, contains 87,806 cells. This grid represents a typical unadapted grid, which is adequately clustered around the geometry but is too coarse in the field to resolve the flow accurately. The grid after three levels of adaptive refinement (shown on the right-hand side of the figure) contains 840,135 cells. The adapted grid is efficiently refined in the field at the 3D conical shock wave structure as clearly indicated in Fig. 10. Even a weaker shock wave in front of the canopy is automatically detected, and the grid is refined accordingly. Also, note the gradual transition of the grid spacing from the original coarse grid to the core of the refined sections where the shock waves are formed.

An unadapted globally fine mesh, with a resolution similar to that of the adapted grid at the shock locations, was also generated for comparison. The fine grid (not shown) contains 11,786,137 cells, which is a typical grid size for hypersonic flow computations. The large difference (more than an order of magnitude) between the size of the two grids illustrates the high degree of efficiency offered by grid adaptation.

The indicator given by Eq. (1) was also used to detect the detached shock waves in this example. The effect of the grid-spacing correction factor in Eq. (1) has resulted in a better detection of small pressure differences away from the geometry. Consequently, the larger cells, which have hardly experienced flow discontinuities in the initial solution, are also flagged and refined. Figure 11 shows static pressure contours on the surface and in the field.

The adapted solution, on the right, exhibits a sharp bow shock extending farther out to the outflow boundary. A secondary shock wave in front of the canopy is also well predicted by grid adaptation. The unadapted solution, exhibiting weak compression waves, is shown on the left-hand side of Fig. 11.

5 Concluding Remarks

A practical adaptive unstructured grid approach has been developed and tested on several three-dimensional cases. The method is based on the proven existing techniques extended for grid adaptation. The "pilot" technology has demonstrated potential for solving complex aerodynamic problems in an efficient and practical fashion. The presented sample results have clearly shown that accurate solutions can be generated automatically with substantially less amount of computational time and cost. Additional work is required to mature the pilot technology and to extend its capabilities. Further developments planned for future work include the implementation of better error/feature indicators for accurate adaptation of solutions involving multiple dominant flow features, solution interpolation between adaptation cycles, and extension of the method for the Navier-Stokes solution adaptive gridding.

6 Acknowledgements

The support of the High Performance Aircraft Office, the Configuration Aerodynamics Branch, and the Subsonic Aerodynamic Branch at the NASA Langley Research Center during the course of this study is gratefully acknowledged.

7 References

[1] Soni BK, Weatherill NP, and Thompson JF. Grid Adaptive Strategies in CFD. Invited Paper, International Conference on Hydro Sciences & Engineering, Washington, D.C., June 7-11, 1993.

[2] Coirier WJ and Powell KG. A Cartesian, Cell-based Approach for Adaptively-refined Solutions of the the Euler and Navier-Stokes Equations. Proceeding of the Surface Modeling, Grid Generation and Related Issues in Computational Fluid Dynamics Workshop, NASA Conference Publication 3291, May 1995.

[3] Mavriplis DJ. Adaptive Meshing Techniques for Viscous Flow Calculations on Mixed Element Unstructured Meshes. AIAA Paper 97-0857, January 1997.

[4] Peraire J, Peiro J, and Morgan K. Adaptive Remeshing for Three-Dimensional Compressible Flow Computations. *Journal of Computational Physics*, 103, pp 269-285, 1992.

[5] Baum JD, Luo H, and Lohner R. A New ALE Adaptive Unstructured Methodology for the Simulation of Moving Bodies. AIAA Paper 94-0414, January 1994.

[6] Pirzadeh SZ. An Adaptive Unstructured Grid Method by Grid Subdivision, Local Remeshing, and Grid Movement. AIAA Paper 99-3255, June 1999.

[7] Baker TJ. private communications.

[8] Pirzadeh S. Three-Dimensional Unstructured Viscous Grids by the Advancing-Layers Method. *AIAA Journal*, Vol. 34, No. 1, pp 43-49, 1996.

[9] Lohner R and Parikh P. Three-Dimensional Grid Generation by the Advancing-Front Method. *International Journal of Numerical Methods in Fluids*, 8, pp 1135-1149, 1988.

[10] Pirzadeh S. Structured Background Grids for Generation of Unstructured Grids by Advancing Front Method. *AIAA Journal*, Vol. 31, No. 2, pp 257-265, 1993.

[11] Pirzadeh S. Recent Progress in Unstructured Grid Generation. AIAA Paper 92-0445, January 1992.

[12] Frink NT. Tetrahedral Unstructured Navier-Stokes Method for Turbulent Flows. *AIAA Journal*, Vol. 36, No. 11, pp 1975-1982, 1998.