AIAA JOURNAL
Vol. 33, No. 10, October 1995

Advance in Overset Grid Schemes: From Chimera to DRAGON Grids

Kai-Hsiung Kao* and Meng-Sing Liou†
NASA Lewis Research Center, Cleveland, Ohio 44135

A new approach that uses nonstructured mesh to replace the arbitrarily overlapped structured regions of embedded grids is presented. The present methodology uses the Chimera overlapped structured grids that are independently generated and body fitted, yielding a high-quality grid readily accessible for efficient solution schemes. A hybrid grid scheme is then proposed that maximizes the advantages of the Chimera scheme and adapts the strengths of the unstructured grid while at the same time keeps its weaknesses minimal. This new adaptation to the Chimera thinking is the direct replacement of arbitrary grid overlapping by nonstructured (DRAGON) grid. The nonstructured grid region sandwiched between the structured grids is limited in size, resulting in only a small increase in memory and computational effort. The DRAGON grid method has three important advantages: 1) preserving strengths of the Chimera grid, 2) eliminating difficulties sometimes encountered in the Chimera scheme, and 3) making grid communication in a fully conservative and consistent manner insofar as the governing equations are concerned. Numerical results on representative two-dimensional steady and unsteady inviscid flows are demonstrated.

I. Introduction

DURING the last decade both structured and unstructured grid systems have been developed and applied for the computations in various computational fluid dynamics (CFD) problems. To deal with situations in which complex geometry imposes great constraints and difficulties in generating grids, composite structured grid schemes and unstructured grid schemes currently are the two mainstream approaches.

The Chimera grid scheme,[1] which uses overset grids to resolve complex geometry or flow features, is generally classed into the composite structured grid category. Overset grids allow structured grids to be used without excessive distortion or inefficient use of grid density. It has been used to compute inviscid and high-Reynolds flow about complex configurations,[2,3] and even been demonstrated for unsteady three-dimensional viscous flow problems[4,5] in which one body moves with respect to another. Overset grids have also been used as a solution adaptation procedure.[6,7]

The Chimera method is an outgrowth of the attempt to generalize a powerful solution approach to more complex situations. It is recognized, however, that there are weaknesses that must be removed if Chimera is to remain competitive with unstructured grids. There are two main criticisms leveled against the current implementations of the Chimera method: 1) the complexity of the interconnectivity is perhaps as difficult as dealing with an unstructured grid, resulting in orphan points and bad quality of interpolation stencils, and 2) nonconservative interpolations to update interface boundaries are often used in practical cases. The fact that interpolation is generally used to connect grids implies that conservation is not strictly enforced. Our experiences have indicated that the interpolation error can become globally significant if numerical fluxes are not fully conserved, in particular when a discontinuity runs through the interpolated region. Also, this error can be strongly affected by the underlying flux schemes, such as central vs upwind schemes. As has been noted, conservative interpolation schemes in two dimensions[8,9] have been shown to be relatively easy. Extension to three dimensions in determining the volume weights for irregular polyhedra, however, is not all that straightforward. Even if three-dimensional conservative interpolation proves to be feasible, a fundamental difficulty exists because of the nonunique distribution of the coarse-grid data to the fine grid.

On the other hand, the unstructured grid method is found to be very flexible to generate grid meshes around complex geometries.[10,11] In particular, solution adaptivity is perhaps its greatest strength.[12,13] The unstructured method, however, has been shown to be extremely memory and computation intensive.[14] Also choices of efficient flow solvers are limited, thus further affecting the efficiency of the unstructured method.

In general, unstructured grid methods are considered to be more versatile and easier to adapt to complex geometry whereas composite structured grid methods are considered to use more efficient numerical algorithms and require less computer memory. However, both pure-strain approaches have their strengths and weaknesses.

We will begin with the premise that the Chimera grid is a formidable choice for dealing with a multicomponent, complex geometry. This choice leaves us a clear task for our research: preserve/maximize desirable features of the method and eliminate shortcomings associated with the current practice. An alternative way to ensure interface conservation and avoid uncertainty on the interpolation scheme altogether would be to introduce an unstructured flow solver in the vicinity of the interface boundaries, as proposed in Refs. 15 and 16. In this paper a new approach is presented that uses a nonstructured mesh to replace the overset region in the Chimera embedded grids. We believe that this approach not only eliminates the shortcoming of the current Chimera method but also preserves its advantages. The extension of the present method to three dimensions is straightforward and only paced by the capability of three-dimensional unstructured grid generation techniques.

II. Chimera Grid

Since the Chimera method is a major steppingstone in our development, it would be worthwhile to briefly describe it for completeness. The Chimera scheme is a grid embedding technique which provides a conceptually simple method for domain decomposition. For instance, a major grid is generated about a main body element; minor grids are then overset on the major grid so as to resolve interesting features of the configuration. Usually, the minor grids are overset on top of the major grid without requiring the mesh boundaries to join in any special way. A common or overlap region, however, is always required to provide the means of matching the solutions across boundary interfaces. To increase the flexibility in the selection of subdomains, the Chimera scheme also allows an implementation to remove regions of a mesh containing

Received Oct. 31, 1994; revision received March 20, 1995; accepted for publication March 24, 1995.

*Senior Research Associate, Institute for Computational Mechanics in Propulsion. Member AIAA.

†Senior Research Scientist, Internal Fluid Mechanics Division. Member AIAA.

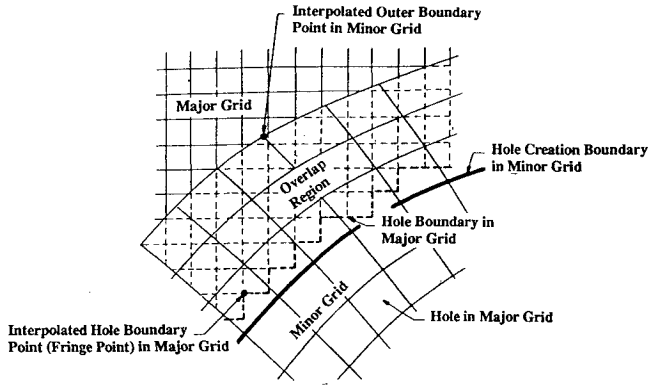

Fig. 1 Construct of the Chimera grid.

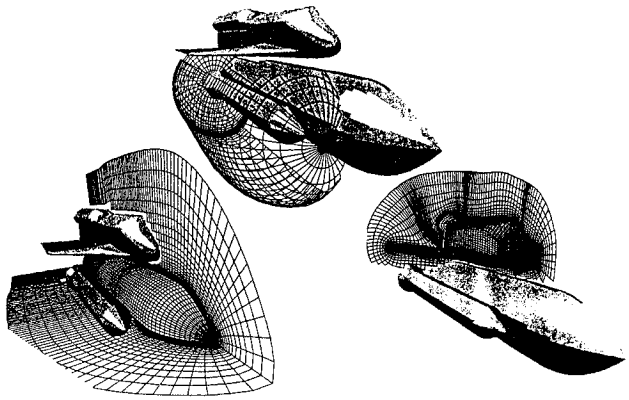

Fig. 2 Chimera grid used for integrated Space Shuttle geometry.[3]

an embedded grid from that mesh. That is, an embedded mesh introduces a "hole" into the mesh in which it is embedded. Typically, a hole is defined through a creation boundary that consists of a surface or a group of surfaces. The purpose of a hole creation boundary is to identify points that are within this boundary. A mesh point is considered to be inside a hole creation boundary if it is inside all surfaces that define the boundary. Figure 1 illustrates the connections between composite overlapping grids, with hole points being blanked by a prescribed creation boundary. A practical application of the Chimera grid system for the integrated Space Shuttle vehicle is shown in Fig. 2, in which three grids conforming the orbiter, external tank, and solid rocket booster together with the associated hole boundaries are shown, representing a triumphant case for the method. Details regarding the grid embedding technique can be found in Refs. 1 and 17.

The Chimera method has two principal elements: 1) decomposition of a chosen computation domain into subdomains and 2) communication of solution data among these subdomains. Software is needed to automatically interconnect grids of subdomains, define the hole region, and supply pointers to facilitate communication among grids during the solution process. We have used the PEGSUS[1,17] codes to perform these task. PEGSUS 4.0[17] is the latest release and executes four basic tasks: 1) precess the grids and user inputs for all subdomains, 2) identify the hole and interpolation boundary points, 3) determine the interpolation stencil and interpolation coefficients for each interpolation boundary point, and 4) supply diagnostic information on the execution and output the results for input to the flow solver.

It must be noted that the PEGSUS codes, even though very powerful and general, should not be taken as error free. Some problems may arise, for example, when the slope can not be uniquely defined either on a hole creation boundary or on the grid. Using the PEGSUS, as any other new techniques and softwares, will require some knowledge about the underlying techniques and experiences. Fortunately, PEGSUS has enough flexibility and some ingenuity on the user's part can usually resolve difficulties.

III. Data Communication

Since the separate grids are to be solved independently, boundary conditions must be made available. Boundary conditions on the interpolated hole boundary and interpolated outer boundary are supplied from the grid in which the boundaries are contained. There are currently several approaches (e.g., Refs. 1, 8, 9, and 18–20) to obtain data for these conditions, but all involve some form of interpolation of data in a grid. Generally, they can be grouped into two categories: nonconservative and conservative interpolations, which are discussed in the following.

A. Nonconservative Interpolation

Once the interpolation stencils are searched and identified, PEGSUS employs a nonconservative trilinear interpolation scheme. There is much uncertainty as regards measuring the local or global effects of the nonconservative interpolation on the solution, especially when a strong gradient region intersects the interpolation region. In Ref. 21, the order of interpolation has been studied in relation to the order of partial differential equations (PDE), the order of the discretization, and the width of interpolation stencils for a model boundary value problem. For a second-order formula, it was found necessary to use an interpolation of at least third order, as the overlap width is on the order of the grid size. An assessment of the Ref. 21 proposal's validity for a range of problems is warranted because interpolating has the advantage of being a relatively simple matter to perform. Another study[22] on incompressible viscous flow also has indicated that a certain amount of mass source is created along the interpolation boundaries, resulting in a pressure buildup in those regions. For some cases,[23] use of direct interpolation for supplying required boundary information results in incompatible boundary conditions which prevent the numerical solution from converging.

B. Conservative Interpolation

It has been asserted that the conservation property needs be enforced for cases in which shock waves or other high-gradient regions cross the region of interpolation. Several conservative interpolation schemes have been proposed for patched interfaces[7,20] and arbitrarily overlapped regions.[8,23] These schemes are relatively easy in a two-dimensional domain but are still substantially more complex than the nonconservative schemes. Their extensions to three dimensions are extremely difficult, if not impossible. Simplification is possible for patched grids in three dimensions,[24,25] but more restrictions are placed on grid generation, making the schemes less attractive.

A fundamental deficiency in all of these approaches is that a choice has to be made concerning the distribution of fluxes from one grid to another, even though the sum of fluxes can be made conservative. Since the overlapped region is necessarily arbitrary, there will be great disparity in grid density and orientation along the overlapped region (or hole boundary). The choice of weighting formulas is not clear and certainly not unique.

IV. Direct Replacement of Arbitrary Grid Overlapping by Nonstructured Grid

We conclude from the preceding section that 1) maintaining grid flexibility and the quality of the Chimera method are definitely to be preserved and even maximized (improved), but 2) focusing on improving (choosing) interpolation schemes perhaps only leads to more complication and does not seem to be a fruitful way to follow.

An alternative method which avoids interpolation and strictly enforces flux conservation altogether for both steady and unsteady problems is to solve the region in question on the same basis as the rest of the domain.

Since the overlapped region is necessarily irregular in shape, the unstructured grid method is most suitable for gridding up this region. Furthermore, since this region is, in general, away from the body where the viscous effect is less important compared to the inviscid effect, the unstructured grid would avoid penalty from accuracy consideration. In this approach we in effect directly replace arbitrary grid overlapping by a nonstructured (DRAGON) grid.[26] Major differences of the current approach from other hybrid methods are as follows. 1) We heavily utilize the proven Chimera method and

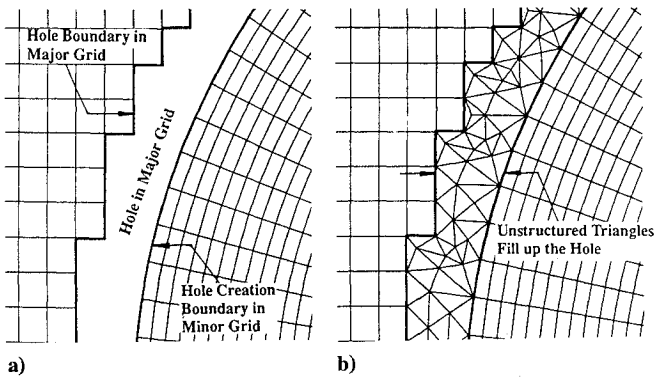

Fig. 3 Direct replacement of arbitrary grid overlapping by nonstructured grid: a) embedded grids with hole region blanked and b) DRAGON grid.

the powerful and versatile automatic code PEGSUS, thus retaining the attractive features already mentioned in the Introduction. 2) We use unstructured grids only in a limited region and mostly located away from viscous-dominant regions, thus minimizing the disadvantages of unstructured grids.

In what follows we will describe the structured and unstructured grid features of the DRAGON grid method separately.

A. Structured Grid Region

The PEGSUS code now is used and modified to provide information necessary for the DRAGON grid. The algorithmic steps are enumerated as follows.

1) As in the Chimera grid, the entire computation domain is divided into subdomains.

2) Hole regions are created, see Fig. 3a, and the IBLANK array is generated in which the elements associated with grid points inside the hole boundary are set to 0 (default value is 1). For example, the outer boundary of a minor grid is used as the hole creation boundary.

3) Both fringe points (hole boundary points) and interpolation boundary points are no longer treated as blanked points. Instead, they are now represented as interior points, and their IBLANK values are set to 1 (rather than 0) in the PEGSUS code.

4) The fringe points as well as the interpolation boundary points are now forming the new boundaries for the unstructured grid region. Details for the unstructured grid will be addressed shortly.

5) Since the interpolation process is no longer performed between structured grid blocks, output files providing interpolation information are deleted from the PEGSUS code.

It is noted that we now achieve a grid system in which grids are not overlapped. When no overlapped region is found, then the empty space will be filled with an unstructured grid, thus resulting in a much more robust and flexible procedure.

B. Nonstructured Grid Region

The gap region created by arbitrarily overset grids inevitably is of irregular shape and it would be very difficult to represent it with structured grids. Unstructured grids are most suitable to fill in this irregularly shaped space. Triangular cells, especially, can provide a good deal of flexibility to adapt to the odd shape. Recall that one important feature in the DRAGON grid is to eliminate any cumbersome interpolation. Unstructured grids alone are not sufficient to do the task. An additional constraint to the grid generation is imposed to require that the boundary nodes of the structured grid, which are output from the PEGSUS code, coincide with vertices of boundary triangular cells. Fortunately, this constraint fits well in unstructured grid generation. The Delaunay triangulation scheme is applied to generate an unstructured grid in the present paper. Figure 3b depicts the unstructured grid filling up the hole created in the structured grid by the Chimera method. The steps adopting the unstructured cells in the framework of the Chimera grid scheme are summarized next.

1) Boundary nodes provided by the PEGSUS code are reordered according to their geometric coordinates.

2) The Delaunay triangulation method is then performed to connect these boundary nodes based on the Bowyer algorithm.[27]

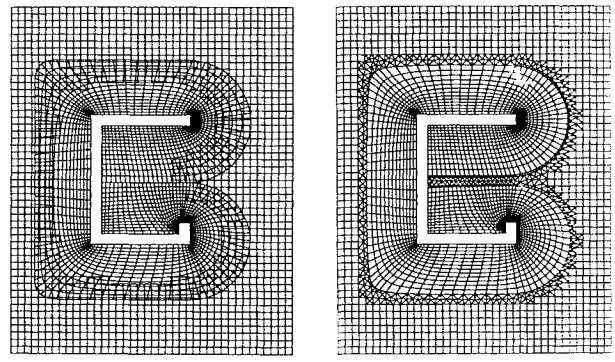

Fig. 4 Comparison of the letter C in a channel obtained by a) Chimera grid and b) DRAGON grid.

3) In the unstructured grid, since there is no logical ordering of the cells and their neighbors, connectivity matrices containing the cell-based as well as edge-based information are introduced. Also the present approach requires additional matrices to connect the structured and unstructured grids. The connectivity matrices used in the present two-dimensional version are summarized as follows:

IEDGENODE(1:2, edges)	2 nodes for each edge
ICELLEDGE(1:3, cells)	3 edges for each triangular cell
IEDGECELL(1:2, edges)	2 neighboring cells for each edge
IEDGETYPE(edges)	edge type (type of boundary condition)
IEDGEFLUX(grids, edges)	edge number that connects structured and unstructured grids
IFLUXINDX(grids, edges)	i index of structured cell that shares the edge fluxes with unstructured cell
IFLUXINDY(grids, edges)	j index of structured cell that shares the edge fluxes with unstructured cell

Questions concerning triangular grid quality and improvements are not pursued here because they are beyond the scope of the present paper.

This step completes the specification of the initial grid. Figure 4 now compares the DRAGON grid and the Chimera grid for the letter C in a channel. The Chimera method wraps around the C with a body-conforming (minor) grid and oversets it on the background (major) H type grid. Note that we intentionally make an unsymmetric C to emphasize grid flexibility.

V. Governing Equations and Solver Algorithm

A computer code based on a time-accurate, finite volume, high-resolution scheme for solving the compressible Navier–Stokes equations has been developed to include both the Chimera overset grid and the nonstructured mesh schemes.

A. Governing Equations

Considering both structured and unstructured grid systems, the time-dependent compressible Euler equations can be expressed in an integral form over an arbitrary control volume Ω:

$$\int_{\Omega} \frac{\partial \boldsymbol{U}}{\partial t} \, \mathrm{d}v + \int_{\partial\Omega} \boldsymbol{F} \cdot \mathrm{d}\boldsymbol{S} = 0 \tag{1}$$

where the conservative-variable vector $\boldsymbol{U}$ is

$$\boldsymbol{U} = [\rho, \rho \boldsymbol{u}, \rho E]^T \tag{2}$$

the inviscid flux $\boldsymbol{F}$

$$\boldsymbol{F} = [\rho \boldsymbol{u}, \rho \boldsymbol{u}\boldsymbol{u} + p\tilde{I}, \rho H \boldsymbol{u}]^T \tag{3}$$

and the equation of state for ideal gas

$$p = (\gamma - 1)\rho e \tag{4}$$

Here e represents the internal energy. The vector quantities, expressed in terms of Cartesian coordinates, are bold face, and the tensor quantities are indicated with a tilde or a dyadic notation such as $\boldsymbol{uu}$.

B. Flux Splitting

Based on the cell-centered finite volume method, the governing equations are semidiscretized. We use the new flux scheme[28] AUSM$^+$ to express the numerical flux at the cell faces. The AUSM$^+$ scheme allows an exact capture of a normal shock by using a suitably chosen interface speed of sound, yields smoother solution by way of including higher order polynomials, and leads to faster convergence rate.

The scheme first splits the full flux into convective and pressure fluxes, $\boldsymbol{F}^c_{1/2}$ and $\boldsymbol{F}^p_{1/2}$ respectively,

$$\boldsymbol{F}_{\frac{1}{2}} = \boldsymbol{F}^c_{\frac{1}{2}} + \boldsymbol{F}^p_{\frac{1}{2}} = c_{\frac{1}{2}} \Phi_{\frac{L}{R}} + P_{\frac{1}{2}} \tag{5}$$

at the cell interface $L < 1/2 < R$. The interface pressure $P_{1/2}$ simply comprises the positive and negative data from appropriate domains of dependence via characteristic speed decomposition. The passive scalar variables $\Phi \equiv (\rho, \rho \boldsymbol{u}, \rho H)^T$ are transported by a common convective velocity $c_{1/2}$ that is constructed in a fashion similar to $P_{1/2}$. Then the upwind idea is used to select the state, i.e., L or R, of the variables to be convected. As such, the interface flux can be recast in the following form:

$$F_{\frac{1}{2}} = c_{\frac{1}{2}} \tfrac{1}{2}[\Phi_L + \Phi_R] - \tfrac{1}{2}\left|c_{\frac{1}{2}}\right| \Delta_{\frac{1}{2}} \Phi + P_{\frac{1}{2}} \tag{6}$$

where $\Delta_{1/2}() = ()_R - ()_L$. Here the first term on the right-hand side is clearly not a simple average of the L and R fluxes, but rather a weighted average via the convective velocity. Readers are encouraged to find the details about the AUSM$^+$ scheme in Ref. 28.

C. Time Integration

The time integration scheme updates the conservative variables at the cell center. The present method originates from the Taylor series expansion in time, as was done in the Lax–Wendroff scheme. Then a two-step scheme, called predictor and corrector steps, with second-order time accuracy can be obtained.

Predictor:

$$\boldsymbol{U}^* = \boldsymbol{U}^n + \Delta t \frac{\partial \boldsymbol{U}^n}{\partial t} \tag{7}$$

Corrector:

$$\begin{aligned} \boldsymbol{U}^{**} &= \boldsymbol{U}^* + \Delta t \frac{\partial \boldsymbol{U}^*}{\partial t} \\ \boldsymbol{U}^{n+1} &= \tfrac{1}{2}(\boldsymbol{U}^n + \boldsymbol{U}^{**}) \end{aligned} \tag{8}$$

It is noted that 1) the predictor step allows a full time step, 2) like other two-step integration schemes, only two levels of storage are needed as $\boldsymbol{U}^n$ is absorbed in forming the residual, and 3) both predictor and corrector steps are identical, with no need of defining a midpoint for the corrector step, leading to simplification of coding and the complexity of evaluating the transport terms.

The stability of the scheme is restricted by the Courant–Friedrichs–Lewy (CFL) number not exceeding unity. For steady-state cases, local time stepping is used to accelerate convergence.

As in the Chimera method, each subdomain grid, including the unstructured grids, is solved independently, the intergrid communication, which is vital for properly propagating flow variations throughout grids, must be established and is the subject of next section.

D. Data Communication Through Grid Interface

In the Chimera method, this communication is made through the hole boundary or the outer boundary. Since the interface treatment methods are not necessarily satisfying any form of conservative constraints, the solutions on overlaid grids are often mismatched with each other. More seriously, this may ultimately lead to spurious or incorrect solutions, especially when a shock wave or high-gradient region passes through boundaries of overlaid grids, as will be seen later.

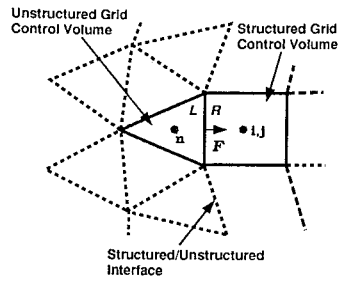

Fig. 5 Fluxes at the cell face connecting the structured and unstructured grids.

In the current work, both the structured and unstructured flow solvers were based on the cell center scheme in which the quadrilateral and triangular cells are used, respectively. Figure 5 shows the interfaces connecting both structured and unstructured grids. As described earlier, the numerical fluxes, evaluated at the cell interfaces, are based on the conditions of neighboring cells (L and R cells). For the unstructured grid, the interface flux F will be evaluated using the structured-cell value as the right state and the unstructured-cell value as the left state. Consequently, the interface fluxes which have been evaluated in the unstructured process can now be directly applied in computing the cell volume residuals for the structured grid.

Thus, the communication in the DRAGON grid is considered seamless in the sense that no manipulation of data, which introduces uncertainties, is required, and the solution is obtained on the same basis whether it be in the structured or unstructured grid region. Consequently, this strictly enforces conservation property locally and globally.

VI. Test Cases and Result Discussions

Computational results are presented in this section for two-dimensional, inviscid, steady and unsteady problems. The test cases include supersonic flow past a cylinder and the shock tube problem. In addition, a supersonic shock past a CFD body (letters) in a channel is simulated to demonstrate the strength of DRAGON grid in dealing with complex geometries. Since our unstructured flow solver is only first-order accurate, the same order of accuracy was also chosen for the structured solver. Implementation of a higher order accurate procedure to the unstructured code is currently underway. For the DRAGON grid approach, both structured and unstructured results can be displayed on the same plot using the flow analysis software toolkit[29] (FAST) visualization package on an IRIS workstation.

A. Case 1: Supersonic Flow past a Cylinder

The first problem solved was that of a circular cylinder in a supersonic freestream ($M = 4$) with the associated bow shock. The flowfield is resolved using the Chimera overset grid scheme where the H type grid (31×31) covers entire domain and the C type grid (41×21) wraps around the cylinder body, as seen in Fig. 6a. Note that the points inside a prescribed hole creation boundary are blanked and a sufficient overlaid region is placed. Thus, a nonconservative bilinear interpolation method is performed at the grid interfaces which allows fluid properties to communicate between the background and cylinder grids. With the DRAGON grid approach, the blanked hole is now extended beyond the outer boundary of the cylinder grid and a new gap region is formed between the embedded grids. The Delaunay triangulation method is then employed which generates triangular meshes to fill in this gap region (Fig. 6b). The new generated unstructured mesh includes 124 nodes and 166 triangular cells for the present test case.

The values of the dependent variables at all of the grid points are set equal to their freestream values initially. The equations of motion with the different grid systems are integrated until the solution converged to its steady-state value. Figures 7a and 7b illustrate the pressure contours obtained at convergence using the Chimera and the DRAGON grids, respectively. The solid-square symbols in these figures represent the shock position predicted by another numerical approach in Ref. 30. It is noted that excellent shock positions are

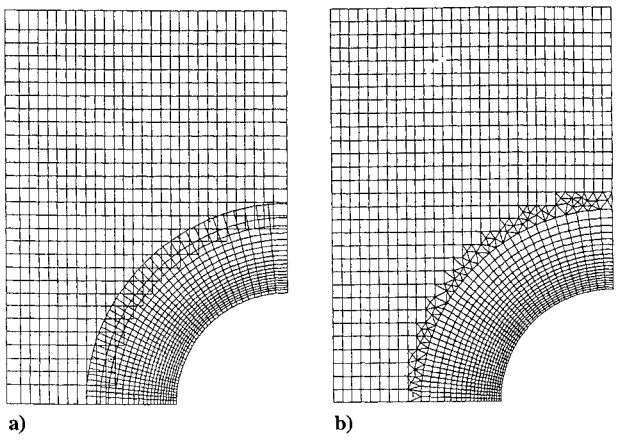

Fig. 6 Grid systems for supersonic flow past a cylinder: a) Chimera grid and b) DRAGON grid.

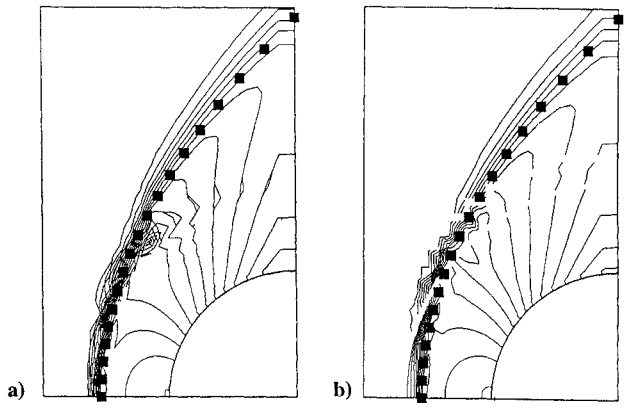

Fig. 7 Comparisons of pressure contours for supersonic flow past a cylinder: a) Chimera grid and b) DRAGON grid.

captured in both grid systems. As displayed in Fig. 7a, however, the Chimera grid produces pressure oscillations and spurious waves in the vicinity of the shock wave/grid interface intersection. On the other hand, with the DRAGON grid shown in Fig. 7b, the continuous contour lines are illustrated when they cross the grid interfaces, implying smooth shock transition. Note also in Fig. 7b that small gaps interrupting the contour lines are displayed along the interface boundaries because cell center values are displayed in the structured grid region whereas in the unstructured grid region variables are plotted at cell vertices.

We further perform comparisons of the primitive variables along the cylinder body, as illustrated in Figs. 8a–8d. The computed pressure, density, and the velocity components are plotted against the results given in Ref. 30. It is obvious that the DRAGON grid method predicts more accurate solutions than the Chimera scheme. Note that as a result of the first-order boundary condition applied at the exit plane ($\theta = 90$ deg), the solutions show a slight discrepancy and further improvement can be achieved using a higher order method.

B. Case 2: Shock Tube Problem

It has been of concern in the CFD community that inaccurate shock speed and strength may result if the numerical procedure is not fully conserved. This case serves to show the effect of interpolation in the Chimera grid and the validity of the DRAGON grid method for a transient problem as a plane shock moves across the embedded-grid region. The shock wave is moving into a quiescent region in a constant-area channel with a designed shock speed $M_s = 4$. Solutions were obtained using three grid systems, namely, the single grid, the Chimera grid, and the DRAGON grid, as displayed in Fig. 9. The single grid has dimensions of 801×21 (Fig. 9a). The solution on this grid will be used for benchmark comparison. Figure 9b illustrates the Chimera grid which includes a major grid (upstream region) of

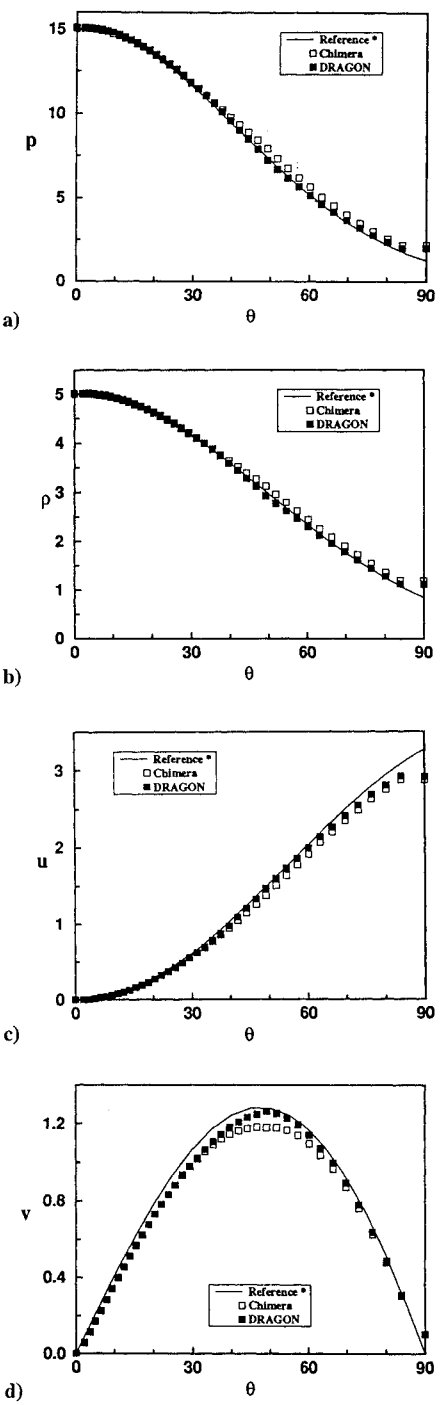

Fig. 8 Distributions of pressure, density, and velocity components along the cylinder body, ■, Ref. 30: a) pressure, b) density, c) u component, and d) v component.

size 41×21, and a minor grid (downstream region) of dimension 781×21. The grid lines in the overlaid grids are generated in any arbitrary fashion. Upon the replacement of grid overlapping by the nonstructured grid, the DRAGON grid is shown in Fig. 9c. Note that all three grid systems apply equivalent grid densities in the major portion of the computation domain, except for that near the interface boundaries.

It is designed so that the initial shock position is marked in the upstream region (Fig. 9). The initial conditions for the flowfield are assumed to have the left and right values as

$$\begin{pmatrix} \rho \\ p \\ u \\ v \end{pmatrix}_L = \begin{pmatrix} 4.57 \\ 13.21 \\ 3.125 \\ 0 \end{pmatrix}, \qquad \begin{pmatrix} \rho \\ p \\ u \\ v \end{pmatrix}_R = \begin{pmatrix} 1 \\ 0.71 \\ 0 \\ 0 \end{pmatrix}$$

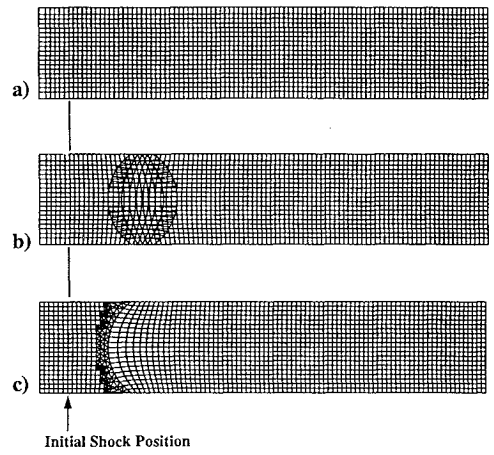

Fig. 9 Grids for moving shock problem, $M_s = 4$: a) single grid, b) Chimera grid, and c) DRAGON grid.

Fig. 10 Pressure contours of the moving shock problem.

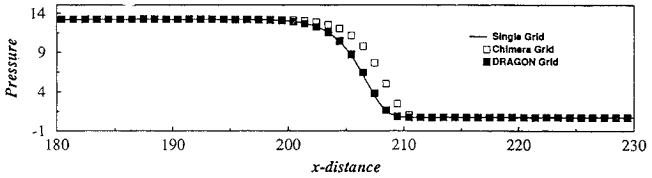

Fig. 11 Comparison of centerline pressures from the three grids.

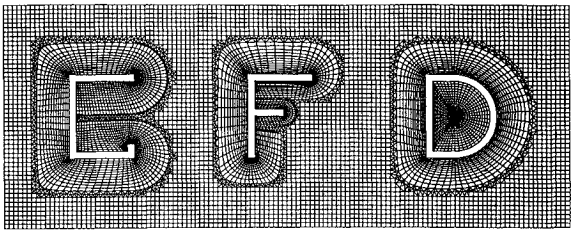

Fig. 12 DRAGON grid for letters CFD in a channel.

Figure 10 displays the pressure contours after 500 time steps; all three grids give similar clean contours after the shock wave has passed through the interface boundaries. A critical comparison of the pressure distributions along the centerline of the channel, however, as plotted in Fig. 11, shows that the Chimera scheme predicts a faster moving shock in the tube, whereas the present DRAGON grid and the single-grid results coincide, indicating that the shock is accurately captured and conservation property well preserved when going through the region of the embedded DRAGON grid.

C. Case 3: Supersonic Shock past Letters CFD Body in a Channel

This case is intended to show 1) the effectiveness of the DRAGON method for handling complex geometries involving odd shapes and composites of different grid topologies and 2) the performance of the solution near the hybrid grid region. Figure 12 shows the DRAGON grid generated for the letters CFD according to the procedure outlined in this paper. In this example, we have hybridizations of C–H grids for the letters C and F and O–H grids for the letter D. In letter F, there is also a hybridization of C–C grids. The strength of the Chimera and, thus, the DRAGON methods is exemplified here since 1) each letter is wrapped (easily) with a body-conforming grid which also well-resolves sharp corners and 2) each letter grid is generated independently and can be moved about at will without affecting other grids. Furthermore, it can be imagined readily (although not demonstrated here) that grid enrichment for resolving geometry or flow details can be added on in a specified region, or vice versa for the reverse process.

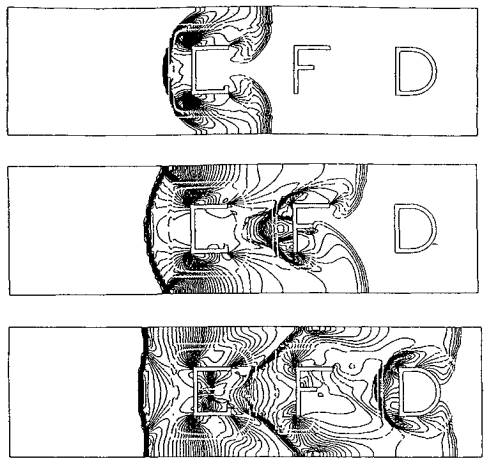

Fig. 13 Evolution of flow, as depicted by pressure contours, about the letters CFD at three representative times.

Next we show the time evolution of flow as a supersonic shock starts moving toward the letters CFD body. Figure 13 displays pressure-contour photographs of the flow at three representative times. It should be noted that the contour lines smoothly cross the intersection boundaries of the structured and unstructured cells.

VII. Conclusions

In this paper we presented a new approach, termed the DRAGON grid, that uses nonstructured meshes to replace the arbitrarily overlapped structured regions in the framework of the Chimera grid. It is designed to further enhance the flexibility of the Chimera embedded-grid technique and to enforce conservative grid communication between embedded grids. Numerical results obtained for steady and unsteady problems indicated that the fluid properties have been accurately carried through the interface boundaries using the DRAGON grid method.

Acknowledgments

We would like to thank P. Jorgenson and J. Loellbach for their help in generating the unstructured grids and providing the basic code of an unstructured flow solver. The computational results for the test problems were obtained on the Cray Y-MP at NASA Lewis Research Center and were partly calculated on the C90 at NASA Ames Research Center.

References

[1]Benek, J. A., Buning, P. G., and Steger, J. L., "A 3-D Chimera Grid Embedding Technique," AIAA Paper 85-1523, July 1985.

[2]Steger, J. L., and Benek, J. A., "On the Use of Composite Grid Schemes in Computational Aerodynamics," *Computer Methods in Applied Mechanics and Engineering*, Vol. 64, Nos. 1–3, 1987, pp. 301–320.

[3]Buning, P. G., Chiu, I. T., Martin, F. W., Jr., Meakin, R. L., Obayashi, S., Rizk, Y. M., Steger, J. L., and Yarrow, M., "Flowfield Simulation of the Space Shuttle Vehicle in Ascent," *Proceedings of the Fourth International Conference on Supercomputing* (Santa Clara, CA), April 1989, pp. 20–28.

[4]Meakin, R. L., and Suhs, N., "Unsteady Aerodynamic Simulation of Multiple Bodies in Relative Motion," AIAA Paper 89-1996, June 1989.

[5]Dougherty, F. C., Benek, J. A., and Steger, J. L., "On Application of Chimera Grid Schemes to Store Separation," NASA TM 88193, Oct. 1985.

[6]Kao, K. H., Liou, M. S., and Chow, C. Y., "Grid Adaptation Using Chimera Composite Overlapping Meshes," *AIAA Journal*, Vol. 32, No. 5, 1994, pp. 942–949.

[7]Berger, M. J., and Oliger, J., "Adaptive Mesh Refinements for Hyperbolic Partial Differential Equations," *Journal of Computational Physics*, Vol. 53, 1984, pp. 484–512.

[8]Moon, Y. J., and Liou, M. S., "Conservative Treatment of Boundary Interfaces for Overlaid Grids and Multi-Level Grid Adaptations," AIAA Paper 89-1980, June 1989.

[9]Wang, Z. J., and Yang, H. Q., "A Unified Conservative Zonal Interface Treatment for Arbitrarily Patched and Overlapped Grids," AIAA Paper 94-0320, Jan. 1994.

[10]Jameson, A., Baker, T. J., and Weatherill, N. P., "Calculation of Inviscid Transonic Flow Over a Complete Aircraft," AIAA Paper 86-0103, Jan. 1986.

[11]Blake, K. R., and Spragle, G. S., "Unstructured 3D Delaunay Mesh Generation Applied to Planes, Trains and Automobiles," AIAA Paper 93-0673, Jan. 1993.

[12]Lohner, R., "An Adaptive Finite Element Scheme for Transient Problems in CFD," *Computer Methods in Applied Mechanics and Engineering*, Vol. 61, No. 3, 1987, pp. 323–338.

[13]Peraire, J., Vahdati, M., Morgan, K., and Zienkiewicz, O. C., "Adaptive Remeshing for Compressible Flow Computations," *Journal of Computational Physics*, Vol. 72, 1987, pp. 449–466.

[14]Ghaffari, F., "On the Vortical-Flow Prediction Capability of an Unstructured-Grid Euler Solver," AIAA Paper 94-0163, Jan. 1994.

[15]Nakahashi, K., and Obayashi, S., "FDM-FEM Zonal Approach for Viscous Flow Computations over Multiple-Bodies," AIAA Paper 87-0604, Jan. 1987.

[16]Weatherill, N. P., "On the Combination of Structured-Unstructured Meshes," *Numerical Grid Generation in Computational Fluid Mechanics*, edited by S. Sengupta, J. Hauser, P. R. Eiseman, and J. F. Thompson, Pineridge, Swansea, Wales, UK, 1988, pp. 729–739.

[17]Suhs, N. E., and Tramel, R. W., "PEGSUS 4.0 User's Manual," Arnold Engineering and Development Center, AEDC-TR-91-8, Arnold AFB, TN, Nov. 1991.

[18]Berger, M. J., "On Conservation at Grid Interfaces," Inst. for Computer Applications in Science and Engineering, NASA ICASE Rept. 84-43, Sept. 1984.

[19]Brown, D. L., Chesshire, G., and Henshaw, W. D., "Composite Grid Data," Los Alamos National Lab., LA-UR-89-2475, Los Alamos, NM, July 1989.

[20]Rai, M. M., "A Conservative Treatment of Zonal Boundaries for Euler Equation Calculations," *Journal of Computational Physics*, Vol. 62, 1986, pp. 472–503.

[21]Chesshire, G., and Henshaw, W. D., "Composite Overlapping Meshes for the Solution of Partial Differential Equations," *Journal of Computational Physics*, Vol. 90, 1990, pp. 1–64.

[22]Hubbard, B., and Chen, H. C., "A Chimera Scheme for Incompressible Viscous Flows with Application to Submarine Hydrodynamics," AIAA Paper 94-2210, June 1994.

[23]Wright, J. A., and Shyy, W., "A Pressure-Based Composite Grid Method for the Navier–Stokes Equations," *Journal of Computational Physics*, Vol. 107, 1993, pp. 225–238.

[24]Klopfer, G. H., and Molvik, G. A., "Conservative Multi-Zonal Interface Algorithm for the 3D Navier–Stokes Equations," AIAA Paper 91-1601, June 1991.

[25]Thomas, J. L., Walters, R. W., Reu, T., Ghaffari, F., Weston, R., and Luckring, J. M., "Patched-Grid Algorithm for Complex Configurations Directed Towards the F/A-18 Aircraft," AIAA Paper 89-0121, Jan. 1989.

[26]Kao, K. H., and Liou, M. S., "Direct Replacement of Arbitrary Grid-Overlapping by Nonstructured Grid," NASA TM 106601, May 1994.

[27]Bowyer, A., "Computing Dirichlet Tessellations," *Computer Journal*, Vol. 24, No. 2, 1981, pp. 162–166.

[28]Liou, M. S., "A Continuing Search for a Near-Perfect Numerical Flux Scheme, Part I: AUSM$^+$," NASA TM 106524, March 1994.

[29]Walatka, P. P., Clucas, J., McCabe, R. K., Plessel, T., and Potter, R., "FAST User Guide," NASA RND-93-010, July 1993.

[30]Lyubimov, A. N., and Rusanov, V. V., "Gas Flows Past Blunt Bodies, Pt. II: Tables of the Gasdynamic Functions," NASA TT F-715, 1970.