AIAA JOURNAL
Vol. 45, No. 7, July 2007

Technical Notes

TECHNICAL NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Overset Adaptive Cartesian/Prism Grid Method for Stationary and Moving-Boundary Flow Problems

Ravishekar Kannan* and Z. J. Wang†
Iowa State University, Ames, Iowa 50014

DOI: 10.2514/1.24200

I. Introduction

THE use of unstructured grids in computational fluid dynamics (CFD) has become widespread during the last two decades, due to their ability to discretize arbitrarily complex geometries and the flexibility in supporting solution-based grid adaptations to enhance the solution accuracy and efficiency [1–7]. In the early days of unstructured grid development, triangular/tetrahedral grids were employed the most in dealing with complex geometries. Recently, mixed or hybrid grids including many different cell types have gained popularity because of the improved efficiency and accuracy over pure tetrahedral grids. For example, hybrid prism/tetrahedral grids [8], mixed grids including tetrahedral/prism/pyramid/hexahedral cells [9], and adaptive Cartesian grid methods [10–15] have been used in many applications with complex configurations. The success demonstrated by unstructured grids for steady flow problems has prompted their applications to unsteady, moving-boundary, flow problems.

A powerful approach for moving-boundary flow problems is the overset Chimera grid method [16]. Originally, the Chimera grid method was used to simplify domain decomposition for complex geometries using structured grids. The method is particularly useful for moving-boundary flow simulations because grid remeshing can be avoided [17]. However, frequent hole-cutting and donor-cell searching may be necessary to facilitate communications between the moving Chimera grids. With continuous improvement over the last decade and a half, the Chimera grid method has achieved tremendous success in handling very complex moving-boundary flow problems. More recently, to further simplify the grid-generation process, unstructured grids are also used in a Chimera grid system for moving-boundary flow computations, making the approach even more flexible in handling complex geometries [18].

In this Technical Note, we advocate the use of an overset adaptive Cartesian/prism grid method for moving-boundary flow computations. The method combines the advantage of the adaptive Cartesian/prism grid in geometry flexibility with that of the Chimera approach in tackling moving-boundary flow without grid remeshing. There are several reasons why an adaptive Cartesian grid is used for moving-boundary problems:

1) Cartesian cells are more efficient at filling space given a certain length scale than triangular/tetrahedral cells.

2) Searching operations can be performed very efficiently with the octree data structure.

3) Solution-based and geometry-based grid adaptations are straightforward to carry out. The grid-generation process is as follows. Body-fitted prism grids are generated first near solid bodies to resolve viscous boundary layers using the advancing layer approach by marching the prism-layer grid in the approximate surface normal directions. The prism-grid-generation approaches (and their limitations) have been well researched in the literature in the past two decades, for example, in [6,8,18,19], and the current implementation follows similar ideas. Therefore, interested readers should consult these references for prism-grid-generation methods and their limitations. An adaptive Cartesian grid is then generated to cover the outer domain and serve as the background grid for bridging the "gaps" between the prism grids. The outer boundaries of the prism grids are used to generate holes in the adaptive Cartesian grid to facilitate data communication. If the bodies move, the prism grids move with the bodies, whereas the Cartesian grid remains stationary. After a few (tens of) time steps, new holes are cut out of the Cartesian grids, and new donor cells are also identified. Solution fields are interpolated from the old Cartesian grid to the new grid using a cellwise linear reconstruction technique.

This Technical Note is organized as follows. In the next section, the overset adaptive Cartesian/prism grid-generation and hole-cutting approaches will be presented, together with illustrative examples. In Sec. III, several steady and unsteady moving-boundary flow problems are computed. Grid-refinement studies are performed to ensure the computational solutions are grid-independent. Computational results are compared with experimental data and other simulations whenever possible. Finally, conclusions from this study are summarized in Sec. IV.

II. Issues on Grid Generation and Hole-Cutting

A. Adaptive Cartesian Grid Generation

After the prism-grid generation, an adaptive Cartesian grid was generated automatically matching the grid resolution near the outer boundaries of the prismatic grids. To support arbitrary local-grid adaptations, the octree data structure was used. The following steps were employed to generate the initial grid:

1) Generate a single-root Cartesian cell based on the size of the computational domain. For external flow problems, the outer boundary can be easily put 100 body lengths away.

2) Recursively subdivide the root cell until all cells are smaller than a specified maximum cell size.

3) Identify all Cartesian cells intersecting the outer boundaries of the prismatic grids.

4) Recursively refine the intersected cells until all the cells intersecting the interfaces match the grid resolution of the prismatic cells.

The final adaptive Cartesian grid is smoothed so that the length scales between two neighboring cells do not differ by a factor more

Presented as Paper 0322 at the 43rd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 10–13 January 2005; received 27 March 2006; revision received 19 January 2007; accepted for publication 6 February 2007.

*Graduate Research Assistant, Department of Aerospace Engineering, 0237 Howe Hall.

†Associate Professor of Aerospace Engineering, Department of Aerospace Engineering. Associate Fellow AIAA.

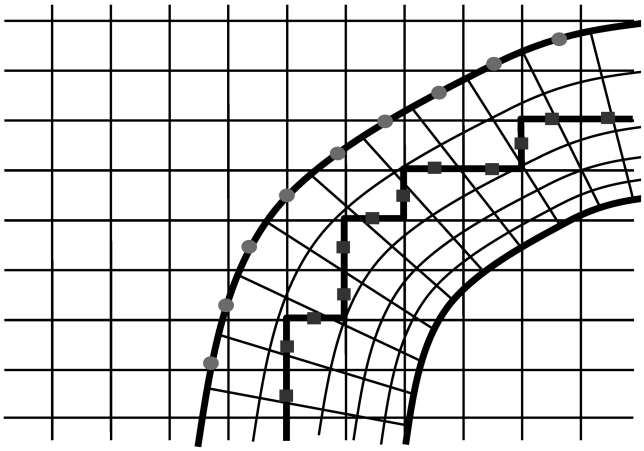


Fig. 1 Schematic of hole-cutting operation.

than two in any coordinate direction. In addition, several buffer layers with the same grid resolution near the outer boundaries of the prismatic grids were used to minimize local discretization error.

B. Automated Hole-Cutting and Donor-Cell Identification

The use of overset adaptive Cartesian and prismatic grids has the potential of handling moving-boundary problems without any user interferences. A critical element in achieving this level of automation is an automated hole-cutting algorithm, in which invalid Cartesian grid cells (cells inside the solid boundary) are excluded from the calculation and donor cells are identified for the hole boundary cells (inner Cartesian boundary) and the prism outer boundary cells. A schematic of the hole-cutting operation is shown in Fig. 1. In the present implementation, we decided to interpolate the flow variables at the faces of the outer boundary of the prism grid (solid circles in Fig. 1) and those of the internal Cartesian boundary (solid squares in Fig. 1). This choice has two advantages. First, it avoids cyclic data interpolation with the minimum overset region. Second, an accurate flux at these faces can be computed directly from the interpolated solutions without a separate data reconstruction, thus easily guaranteeing second-order solution accuracy. Another important decision is which surface to use to perform hole-cutting. In a previous implementation, the solid wall was used for this purpose. However, the problem with this approach is that the Cartesian internal boundary cell could be within the viscous boundary, thus incurring large errors in the solution on the Cartesian grid because of insufficient grid resolution for the viscous boundary layer. A more accurate approach is to locate the internal Cartesian boundary as far as possible from the viscous boundary layer while avoiding cyclic data interpolation. This is achieved by using the second outermost prism-layer grid as the hole-cutting surface, and the Cartesian-grid-hole internal boundary is located immediately inside this prism surface, as shown in Fig. 1.

The efficiency of the hole-cutting algorithm is critical because hole-cutting is performed many times in a moving-boundary flow simulation. To achieve the maximum efficiency, search trees were used extensively. The hole-cutting algorithm consists of the following steps:

1) Find and mark all of the cells in the Cartesian grid that intersect the hole-cutting surface. The octree data structure is used extensively in this step, for efficiency.

2) The Cartesian cells inside the marked intersected cells are blanked and thus removed from the computational domain. Next, the internal Cartesian hole boundary faces are identified.

3) Use the alternating digital tree (ADT) to find the prismatic donor cells. These are the prism cells that bound the centroids of the boundary faces of the inner Cartesian hole boundary faces [20].

4) Use the octree tree to identify the Cartesian donor cells. These Cartesian cells bound face centers of the outermost prism layers [20].

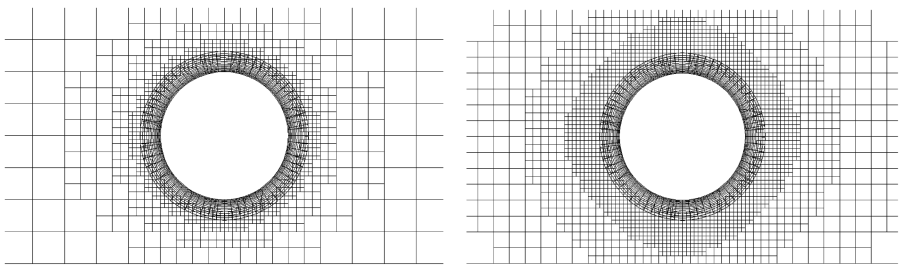
In addition, it was found essential to place at least three buffer layers in the Cartesian grid when grid resolution changes from fine to coarse, as shown in Fig. 2. This is done to ensure a smooth solution transition from the prism to the Cartesian grid and within the Cartesian grid. Figure 2 shows two possible scenarios: one with just one buffer layer and the other with four buffer layers. Clearly, the scenario with just one buffer layer does not allow a smooth switch from the prism to the Cartesian grid.

After each time step/iteration, the field variables at the outer boundary faces are interpolated from the Cartesian grid using a local linear data reconstruction on the Cartesian grid. Similarly the solutions at the hole boundary faces are interpolated from the prismatic grids. The presence of moving bodies changes the relative orientation of the overset grids continually during the flow simulation. In theory, the hole-cutting subroutine needs to be invoked after each and every time step. However, it is not necessary in most viscous simulations to rebuild the chimera hole after each time step. This is due to the fact that the time step computed based on the stability criteria at the highly clustered viscous grid is generally small enough so as to not affect the grid geometry radically near the prism outer boundaries. It could be figured intuitively that the maximum distance that can be traversed (before the chimera hole is rebuilt) should not be greater than the thickness of the outermost prism layer.

For a hole boundary face or an interpolation boundary face, the fluxes are required at the face center to update the conservative variables at cells adjacent to the faces. To compute the flux at the face center, the solution at the face center is computed from the following technique. The cells bounding the face centers of the outer boundary faces or interpolation boundary faces are found and called donor cells. The solution is assumed linear over the donor cell. The solutions at the face center are computed using a first-order Taylor expansion. Then the fluxes are computed based on the interpolated solutions.

III. Test Results

In this section, the overset adaptive Cartesian/prism grid method is tested for both stationary and moving-boundary flow problems. The flow solver employs a second-order finite volume method for arbitrary grids, and details are contained in [14,15]. The following two cases are presented.



a) **b)**

Fig. 2 Cartesian grid with different buffer layers: a) single buffer layer and b) four buffer layers.

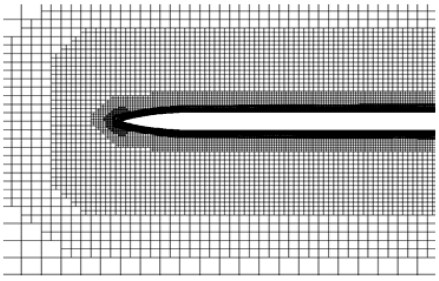


Fig. 3 Overset Cartesian/prism grid generated for the missile.

A. High-Angle-of-Attack Missile Aerodynamics

The first test case was performed for an ogive/cylinder configuration, for which extensive experimental data is available for comparison. The configuration used in the current study is a 3-caliber ogive with a 10-caliber cylindrical afterbody. The geometry of the missile and the overset Cartesian/prism grids generated for the fine mesh are shown in Fig. 3. The coarse grid had a total of 700,000 cells and the fine grid had a total of 1.3×10^6 cells. The flow conditions are: $M = 1.8$, $\alpha = 14$ deg, and $Re = 6.56 \times 10^6$. This particular configuration results in large separation regions and is a severe test for any turbulence model. The Spalart–Allmaras turbulence model was deployed for the current simulation.

The computed pressure coefficient profiles were compared with the existing computational data [14] and experimental data [21] at several cross-sectional stations in Fig. 4. The plots indicate that the obtained solutions are approaching grid independence and are in reasonable agreement with the experimental data. The results clearly indicate that the overset grids can be used with confidence for solving high Reynolds number and supersonic cases.

B. Flow over a Moving Sphere

This case was selected to validate the moving-grid flow solver. A sphere moves from right to left in quiescent air with a Mach number of 0.2. If the reference frame is fixed on the moving sphere, the flowfield should reach a steady state after the initial transients propagate out of the solution domain.

1. Inviscid Flow over a Moving Sphere

The computational grids at two different times are shown in Fig. 5. The outer boundary of the computational grid is located 32 times the diameter away from the initial position of the sphere.

The pressure distributions at two different times are displayed in Fig. 6. Note that initially, a very high/low-pressure region was created on the left/right side of the sphere due to the sudden motion. As time goes, the flowfield becomes nearly "steady" for an observer stationed on the sphere. In fact, the pressure field created by the moving sphere after a long time is compared with that created by a freestream of Mach 0.2 over a stationary sphere in Fig. 7. It is observed that the pressure fields are very similar. The coefficient of pressure was compared with the results of the stationary case. The match was excellent. This is shown in Fig. 8.

2. Laminar Flow over a Moving Sphere

For this simulation, the sphere was moved from the right to the left in quiescent air. The fluid was assumed to be laminar. The Reynolds number was 118. The c_p and c_f profiles were computed and were compared with the results obtained from the flow over the stationary sphere. These are plotted in Figs. 9 and 10. In general, the match between the moving case and the stationary case was very good.

A contour plot of the Mach number was obtained for the current case. This plot is shown in Fig. 11. Note that the contour lines

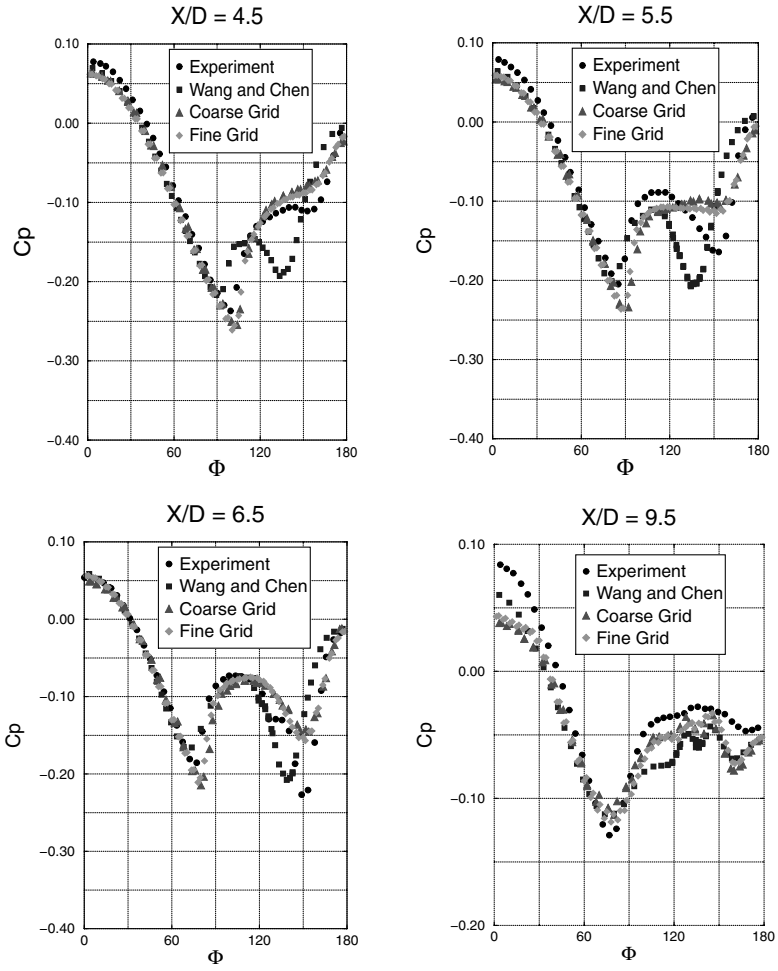


Fig. 4 Comparison between the computed and the existing c_p profiles at various cross-sectional stations.

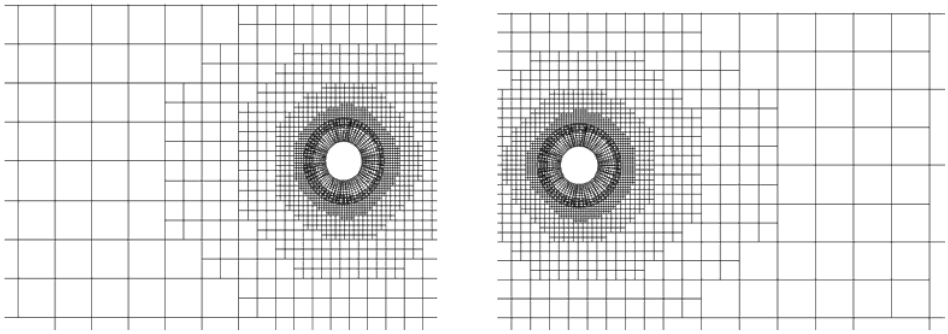


Fig. 5 Computational grids at two different times for the moving sphere problem.

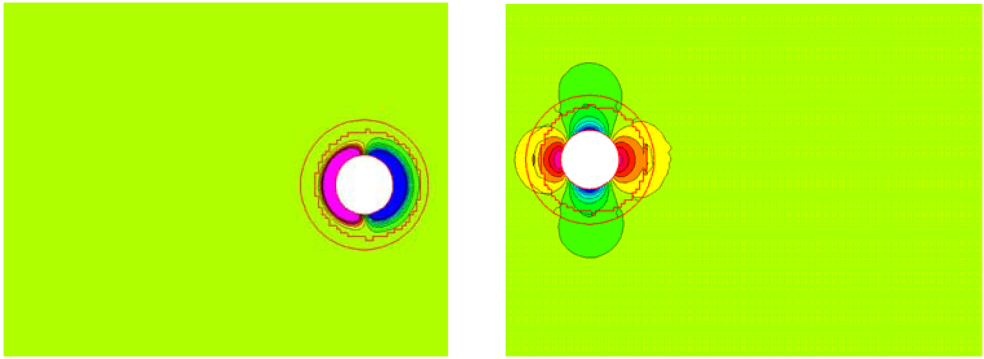
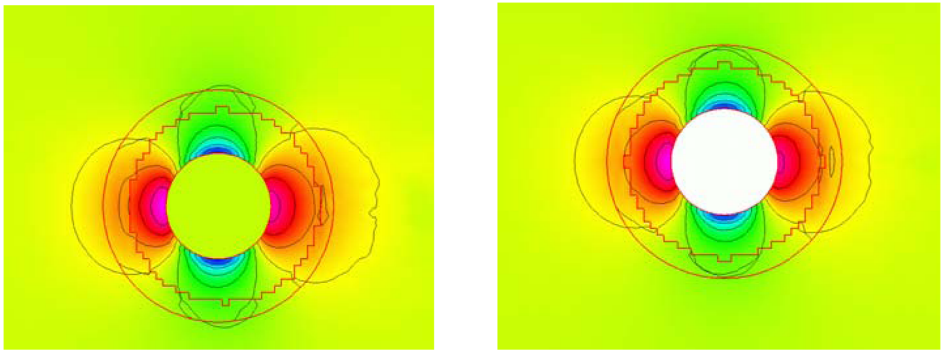


Fig. 6 Pressure distributions at two different times for the moving sphere problem.



a) b)

Fig. 7 Comparison of pressure distributions for a) a moving sphere in quiescent air and b) flow around a stationary sphere.

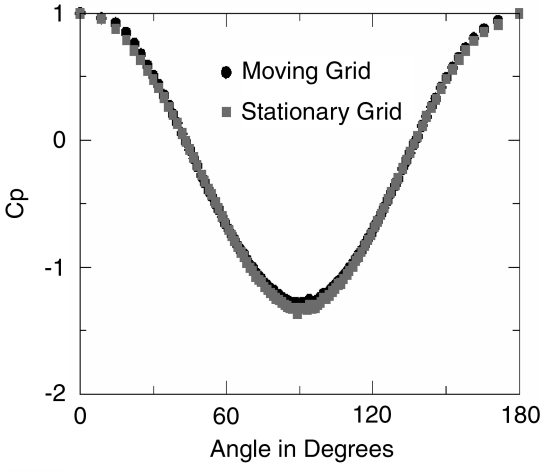


Fig. 8 Static pressure coefficient obtained for inviscid flow over a sphere.

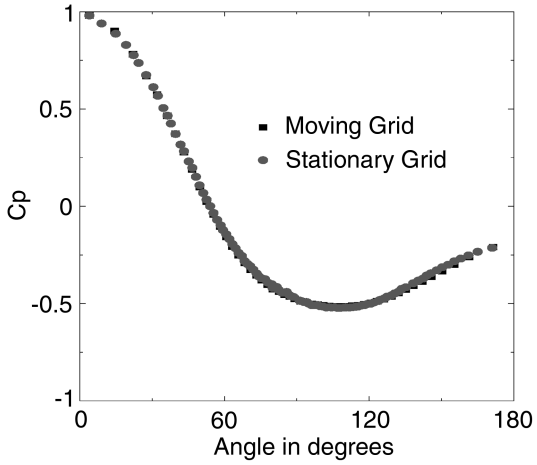


Fig. 9 Static pressure coefficient obtained for laminar flow over a sphere.

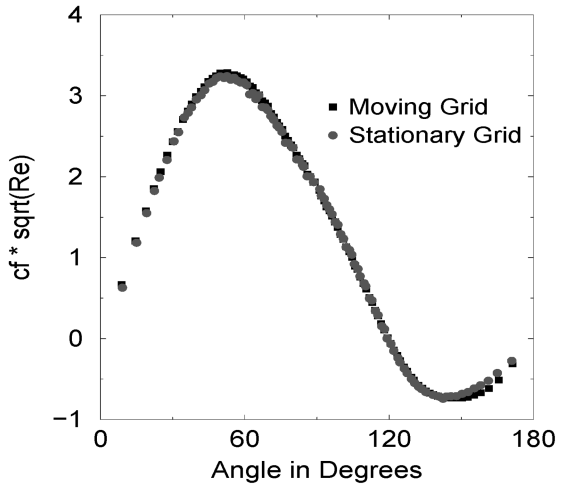


Fig. 10 Skin friction coefficient obtained for laminar flow over a sphere.

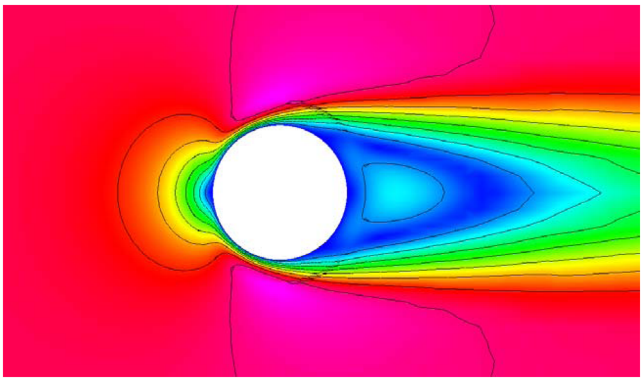


Fig. 11 Mach contours obtained for the $Re = 118$ case.

transition reasonably smoothly from the prism to the adaptive Cartesian grids.

IV. Conclusions

In the present study, an overset adaptive Cartesian/prism grid method was developed to simulate moving-boundary flow problems. The method combines the advantage of an adaptive Cartesian/prism grid in geometry flexibility with that of the Chimera approach in tackling moving-boundary flow without grid remeshing. Advantages of the method include:

1) Cartesian cells are more efficient in filling space given a certain length scale than triangular/tetrahedral cells.

2) Searching operations can be performed very efficiently with the octree data structure

3) Solution-based and geometry-based grid adaptations are straightforward to carry out.

Human interference for moving-boundary flow simulation was reduced to a minimal level. This is mainly due to the automation of the hole-cutting and the donor-cell-identification procedures. The preceding procedures used the ADT and the octree data structured for carrying out the search operations in an efficient manner. An implicit LU-SGS approach is employed for the time integration. This allows the user to specify the time step based on geometric bounds rather than stability limits. A significant reduction in the number of hole-cutting operations is attained. Hence the overhead created by the grid-generation procedures is reduced to less than 10%.

The grid generator and flow solver were coupled successfully to tackle a moving-boundary flow problem with reasonable computational results. Future research includes the flowing areas: 1) enabling solution-based grid adaptation and 2) enabling nonrigid body deformations.

Acknowledgments

The authors gratefully acknowledge the support from the U.S. Air Force Office of Scientific Research (grant number FA9550-04-1-0053), with Fariba Fahroo as the Technical Monitor.

References

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

[2] Peraire, J., Vahdati, M., Morgan, K., and Zienkiewicz, O. C., "Adaptive Remeshing for Compressible Flow Computations," *Journal of Computational Physics*, Vol. 72, No. 2, 1987, pp. 449–466.

[3] Lohner, R., and Parikh, P., "Generation of Three-Dimensional Unstructured Grids by the Advancing Front Method," *International Journal for Numerical Methods in Fluids*, Vol. 8, No. 10, 1988, pp. 1135–1149.

[4] Anderson, W. K., "A Grid Generation and Flow Solution Method for the Euler Equations on Unstructured Grids," *Journal of Computational Physics* Vol. 110, No. 1, 1994, pp. 23–38.

[5] Venkatakrishnan, V., "A Perspective on Unstructured Grid Flow Solvers," AIAA Paper 95-0667, Jan. 1995.

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

[7] Weatherill, N. P., "Unstructured Grids: Procedures and Applications," *Handbook of Grid Generation*, edited by J. F. Thompson, B. K. Soni, and N. P. Weatherill, CRC Press, Boca Raton, FL, 1998, Chap. 26.

[8] Kallinderis, Y., Khawaja, A., and McMorris, H., "Hybrid Prismatic/Tetrahedral Grid Generation for Complex Geometries," *AIAA Journal*, Vol. 34, No. 2, 1996, pp. 291–298.

[9] Coirier, W. J., and Jorgenson, P. C. E., "A Mixed Volume Grid Approach for the Euler and Navier-Stokes Equations," AIAA Paper 96-0762, Jan. 1996.

[10] Bayyuk, S. A., Powell, K. G., and van Leer, B., "A Simulation Technique for 2D Unsteady Inviscid Flows Around Arbitrarily Moving and Deforming Bodies of Arbitrarily Geometry," AIAA Paper 93-3391-CP, 1993.

[11] Aftosmis, M. J., Berger, M. J., and Melton, J. E., "Robust and Efficient Cartesian Mesh Generation for Component-Based Geometry," AIAA Paper No. 97-0196, 1997.

[12] Karman, S. L., "Splitflow: A 3D Unstructured Cartesian/Prismatic Grid CFD Code for Complete Geometries," AIAA Paper 95-0343, 1995.

[13] Murman, S., Aftosmis, M., and Berger, M., "Implicit Approaches for Moving Boundaries in a 3-D Cartesian Method," AIAA Paper 2003-1119, 2003.

[14] Wang, Z. J., and Chen, R. F., "Anisotropic Solution-Adaptive Viscous Cartesian Grid Method for Turbulent Flow Simulation," *AIAA Journal*, Vol. 40, No. 210 2002, pp. 1969–1978.

[15] Zhang, L. P., and Wang, Z. J., "A Block LU-SGS Implicit Dual Time-Stepping Algorithm for Hybrid Dynamic Meshes," *Computers and Fluids*, Vol. 33, No. 7, 2004, pp. 891–916.

[16] Benek, J. A., Steger, J. L., and Dougherty, F. C., "A Flexible Grid Embedding Technique with Application to the Euler Equations," AIAA Paper 83-1944, 1983.

[17] Meakin, R. L., "On the Spatial and Temporal Accuracy of Overset Grid Methods for Moving Body Problems," AIAA Paper 94-1925, 1994.

[18] Togashi, F., Nakahashi, K., Ito, Y., Iwamiya, T., and Shimbo, Y., "Flow Simulation of NAL Experimental Supersonic Airplane/Booster Separation Using Overset Unstructured Grids," *Computers and Fluids*, Vol. 30, No. 6, July 2001, pp. 673–688.

[19] Sharov, D., Hong, L., Baum, J. D., and Lohner, R., "Unstructured Navier-Stokes Grid Generation at Corners and Ridges," *International Journal for Numerical Methods in Fluids*, Vol. 43, Nos. 6–7, 2003, pp. 717–728.

[20] Wang, Z. J., and Parthasarathy, V., "A Fully Automated Chimera Methodology for Multiple Moving Body," *International Journal for Numerical Methods in Fluids*, Vol. 33, No. 7, 2000, pp. 919–938.

[21] Sturek, W. B., Birch, T., Lauzon, M., Housh, C., Manter, J., Josyula, E., and Soni, B., "The Application of CFD to the Prediction of Missile Body Vortices," AIAA Paper 97-0637, Jan. 1997.

K. Fujii
Associate Editor