AIAA JOURNAL
Vol. 35, No. 3, March 1997

Two-Dimensional Unsteady Navier–Stokes Solution Method with Moving Overset Grids

Ismail H. Tuncer*
U.S. Naval Postgraduate School, Monterey, California 93943-5106

A simple numerical algorithm to localize intergrid boundary points and to interpolate unsteady solution variables across two-dimensional, structured overset grids is presented. Overset grids are allowed to move in time relative to each other. Intergrid boundary points are localized in a triangular stencil on the donor grid by a directional search algorithm. The final parameters of the search algorithm give the interpolation weights at the intergrid boundary point. Numerical results are presented for steady and unsteady viscous flow solutions over an airfoil undergoing a sinusoidal flapping motion. Computed flowfields demonstrate the accuracy of the method, and excellent agreement is obtained against the single grid solutions. The method is independent of numerical solution algorithms, and it may easily be implemented on any two-dimensional, single-block flow solver to make it a multiblock, zonal solver with arbitrarily overset/overlapping computational grids.

Nomenclature

c	= chord length
$G_{i,j}$	= grid point at (i, j)
h	= plunge amplitude
k	= reduced frequency, $\omega c/2U_\infty$
M_∞	= freestream Mach number
Re_∞	= Reynolds number based on chord
t	= nondimensional time, $t'/(c/U_\infty)$
t'	= time
U_∞	= freestream velocity
X_P, Y_P	= x, y coordinates of point P
α	= angle of attack
α, β, γ	= coefficients; also interpolation weights
ω	= angular frequency

Introduction

AMONG structured, block-structured, and unstructured computational grids, structured grids still remain the most common in the computational fluid dynamics (CFD) community.[1] The basic advantage of the structured grids is that they are still the most suitable for the highly efficient solution algorithms developed and refined since the early establishment of the science of CFD. Because of the inherent connectivity information, numerical solution algorithms with structured grids are easy to implement on conventional, shared memory/multiprocessor computers.

In discretizing complex geometries and irregular boundaries, the block-structured grid techniques,[2,3] where a group of structured grid blocks are patched together with enforced continuity across the interfaces, are currently being utilized. Although the block-structured grids have all of the inherent advantages of structured grids, they are not well suited for unsteady flow problems with moving boundaries, in which all the grid blocks need to be regenerated at every time step to satisfy the continuity across interfaces.

The recent applications of unstructured grids[1,4] offer a promising alternative for treating irregular boundaries. However, for computation of unsteady flows with moving boundaries, the whole computational grid has to be regenerated at every time step similar to the block-structured grids. This process may become quite complex and costly.[5] Furthermore, the application of algebraic turbulence models in viscous flow problems, which utilizes the direction normal to solid surfaces, causes additional difficulties in unstructured grid topologies.[4,6]

An alternative for treating complex geometries with moving boundaries is the use of overset grids, in which individual structured grids are overset on each other and free to move with respect to each other. In the present work, a new method is presented for the solution of the unsteady Navier–Stokes equations on overset, moving grids. The present method is similar to the Chimera/Pegsus[7] approach in essence; however, the two approaches differ in the way the intergrid boundary points are localized and the flow variables are interpolated.

In Pegsus, the localization of intergrid boundary points and the interpolation of flow variables across subgrids are done in several stages. First, all of the grid points on the base grid are tested to determine if they lie within a certain distance from a temporary origin placed in the overlapping subgrid. All of the grid points within the specified distance are next tested to see if they lie inside or outside of the overlapping grid boundary. The sign of the vector $\boldsymbol{N} \cdot \boldsymbol{R}$, where $\boldsymbol{R}$ is the relative position vector of the grid points and $\boldsymbol{N}$ is the unit outward normal at the cross-section of $\boldsymbol{R}$ and the overset grid boundary, determines the right direction. The closest points outside of the overlapping boundary are then assigned as the boundary points on the base grid. Finally, interpolation cells on the base grid, into which a boundary point of the overlapping grid falls, are localized, and the corresponding bilinear (trilinear in three dimensions) interpolation data are stored in an external file to be utilized by the flow solver. For unsteady flows, the Pegsus code has to be integrated into the flow solver and to be executed at every time step.

In the present approach, the boundary point localization and interpolation of flow variables are integrated into a directional search algorithm in the main flow solver. No external data are stored. The directional search algorithm localizes the boundary points sequentially, and the flow variables are interpolated as an integral part of the localization process. Thus, the application of intergrid boundary conditions is robust and efficient. The present method has already been applied to overlapping grid boundaries in an extensive study on the airfoil interference effects in Ref. 8. In this work, the method is extended to fully overset grids moving relative to each other.

Boundary Point Localization

In the present approach, the intergrid boundary points on overset subgrids are localized on the background grids with a directional search algorithm. The directional search algorithm is depicted in Fig. 1. Here, a single point P, denoted by the + sign, is localized on a Cartesian grid in terms of the three closest grid points, which constitute a triangular stencil. The search process starts from an arbitrary grid point $G_{i,j}$ as denoted in the figure. It is then tested to see whether the point P falls into the triangular stencil defined by

Received Nov. 7, 1995; presented as Paper 96-0822 at the AIAA 34th Aerospace Sciences Meeting, Reno, NV, Jan. 15–18, 1996; revision received Oct. 14, 1996; accepted for publication Nov. 22, 1996; also published in *AIAA Journal on Disc*, Volume 2, Number 2.

*Research Assistant Professor, Department of Aeronautics and Astronautics. Member AIAA.

grid points at $G_{i,j}$, $G_{i\pm1,j}$, and $G_{i,j\pm1}$, where the correct sign $\pm$ is determined by the search direction. The search direction is based on the geometry gradients on the donor grid. The test of whether the point P falls into the triangle is based on the representation of a convex surface defined by three points. Because any point in a convex triangular surface may uniquely be represented by three real numbers provided that they are all positive, the solution of the following three linear equations in terms of α, β, and γ determines whether point P falls inside the triangle:

$$X_P = \alpha X_{i,j} + \beta X_{i\pm1,j} + \gamma X_{i,j\pm1}$$

$$Y_P = \alpha Y_{i,j} + \beta Y_{i\pm1,j} + \gamma Y_{i,j\pm1}$$

$$1 = \alpha + \beta + \gamma$$

where X_P and Y_P are the coordinates of the point to be localized, P. If the coefficients α, β, and γ are less than or equal to one and are all positive, then the point P lies inside the triangular stencil. If not, the search location $G_{i,j}$ is advanced diagonally, as shown by arrows and circles in Fig. 1.

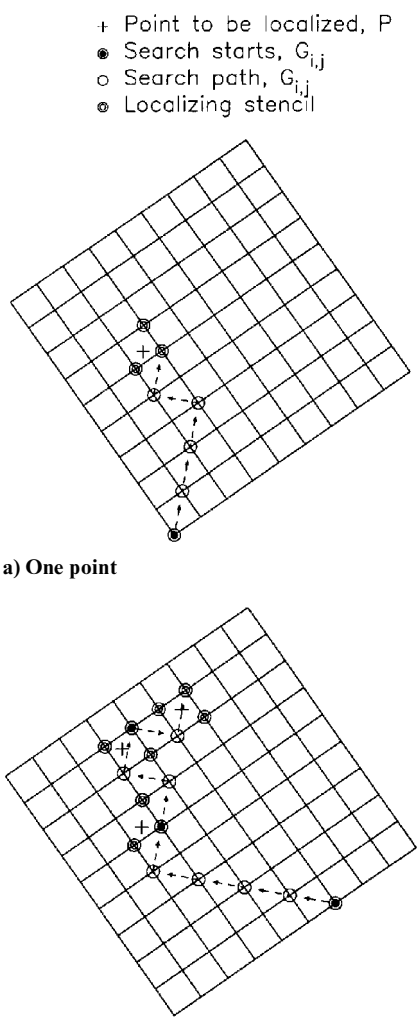
Several algorithms were developed to determine the search direction. The most robust one was found to be the quadrant walking algorithm, in which the search direction is based on the local quadrant the boundary point falls into. As shown in Fig. 1, the search point $G_{i,j}$ advances diagonally in one of the four quadrants into which point P falls. The quadrant point P falls into is determined by the dot products of the vector $(\overline{P - G_{i,j}})$ with the vectors $(\overline{G_{i\pm1,j} - G_{i,j}})$ and $(\overline{G_{i,j\pm1} - G_{i,j}})$.

Finally, the triangular stencil into which point P falls is localized as depicted by the circles at its vertices. Once the point is localized, the coefficients α, β, and γ give the geometric interpolation weights at point P in terms of the function values at the three interpolation points. Thus, the interpolation weights are readily available to interpolate flow variables at point P from the donor grid:

$$Q_P = \alpha Q_{i,j} + \beta Q_{i\pm1,j} + \gamma Q_{i,j\pm1}$$

The sequential localization of several points with a different initial search point is depicted in Fig. 1b. As shown, the search path similarly follows the quadrant walking algorithm, and the search process for localizing the next consecutive point starts from the $G_{i,j}$ grid point, which localizes the previous point.

Note that no interpolation data need to be stored for the application of intergrid boundary conditions. Because intergrid boundary points are localized sequentially with the quadrant walking search algorithm and the search algorithm is unified with the interpolation scheme, the application of the intergrid boundary conditions with the present method becomes quite robust and efficient.



a) One point

b) Sequential

Fig. 1 Localization of points.

Numerical Method

In the overset grid approach, the computational domain consists of structured subgrids, which may overlay on each other. The structured subgrids created around each component in the flowfield or over subdomains of complex geometries are put together to discretize the whole computational domain. Figure 2 shows the overset grids used.

The Navier–Stokes equations are solved in each subgrid with the proper boundary conditions. In addition to the boundaries of the computational domain, subgrids have intergrid boundaries with the neighboring donor subgrids and may also contain holes, as shown in Fig. 2. The proper boundary conditions, therefore, are needed at the intergrid boundaries of a subgrid, and the hole points should be excluded from the integration of the Navier–Stokes equations.

Navier–Stokes Solver

An implicit, thin–layer, unsteady Navier–Stokes solver[8–10] with the third-order accurate Osher's upwind biased flux difference splitting scheme is employed. The strong conservation-law form of the two-dimensional, thin-layer Navier–Stokes equations in a curvilinear coordinate system, (ξ, ζ), along the axial and circumferential direction, respectively, is given as follows:

$$\partial_t \hat{\boldsymbol{Q}} + \partial_\xi \hat{\boldsymbol{F}} + \partial_\zeta \hat{\boldsymbol{G}} = Re^{-1} \partial_\zeta \hat{\boldsymbol{S}} \tag{1}$$

where $\hat{\boldsymbol{Q}}$ is the vector of conservative variables, $1/J(\rho, \rho u, \rho w, e)$, $\hat{\boldsymbol{F}}$ and $\boldsymbol{G}$ are the inviscid flux vectors, and $\hat{\boldsymbol{S}}$ is the thin-layer approximation of the viscous fluxes in the ζ direction, normal to the airfoil surface:

$$\hat{\boldsymbol{F}} = \frac{1}{J}\begin{pmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho w U + \xi_z p \\ (e+p)U - \xi_t p \end{pmatrix}, \qquad \hat{\boldsymbol{G}} = \frac{1}{J}\begin{pmatrix} \rho W \\ \rho u W + \zeta_x p \\ \rho w W + \zeta_z p \\ (e+p)W - \zeta_t p \end{pmatrix}$$

$$\hat{\boldsymbol{S}} = \frac{1}{J}\begin{pmatrix} 0 \\ \mu m_1 u_\zeta + (\mu/3) m_2 \zeta_x \\ \mu m_1 w_\zeta + (\mu/3) m_2 \zeta_z \\ \mu m_1 m_3 + (\mu/3) m_2 + (\zeta_x u + \zeta_z w) \end{pmatrix}$$

where

$$m_1 = \zeta_x^2 + \zeta_z^2 \qquad m_2 = \zeta_x u_\zeta + \zeta_z w_\zeta$$

$$m_3 = (u^2 + w^2)/2 + \kappa Pr^{-1}\left(\frac{\partial a^2}{\partial \zeta}\right)$$

and U and W are the contravariant velocity components. In the equations, all dimensions are normalized with the airfoil chord length c. Here, ρ is the density normalized with the freestream density ρ_∞; u and w are the Cartesian velocity components in the physical domain, which are normalized with the freestream speed of sound a_∞; e is the total energy per unit volume normalized with

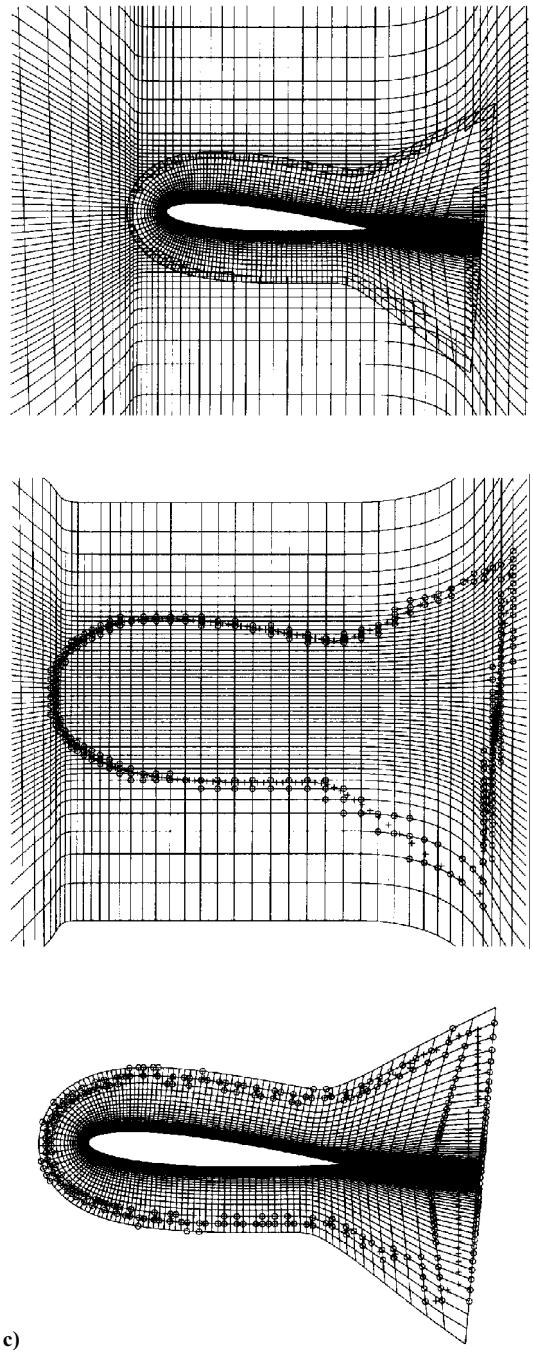


Fig. 2 Overset grid system at $\alpha = 5$ deg and localization of the intergrid boundary points.

$\rho_\infty a_\infty^2$; and Pr is the Prandtl number. The pressure is related to density and total energy through the equation of state for an ideal gas, $p = (\gamma - 1)[e - \rho(u^2 + w^2)/2]$.

The flowfield is assumed to be fully turbulent. The Baldwin–Lomax algebraic turbulence model and the Baldwin–Barth one-equation turbulence model[11] are currently implemented. The Baldwin–Barth model gives a continuous turbulent eddy viscosity distribution across the intergrid boundaries and is well suited for solutions with overset grids.

Boundary Conditions

Boundary conditions are applied on the airfoil surfaces and at the farfield and intergrid boundaries. On the airfoil surfaces, the no-slip boundary condition is applied. For the flapping airfoil, the surface fluid velocity is set equal to the prescribed airfoil flapping velocity so that the no-slip condition is satisfied. Because the formulation of the Navier–Stokes solver is based on an inertial frame of reference, the flapping motion of the airfoil is implemented by moving the airfoil and the computational grid around it in the transverse direction as described by the frequency and the amplitude of the flapping motion. The density and the pressure values on the airfoil surfaces are obtained from the interior by a simple extrapolation.

At the far-field inflow and outflow boundaries, the flow variables are evaluated using the zero-order Riemann invariant extrapolation.

At the overset intergrid boundaries, the conservative flow variables are interpolated at every time step using the present boundary point localization and interpolation method.

In the Baldwin–Barth turbulence model, the freestream turbulent Reynolds number is set to 0.1. At the intergrid boundaries, turbulent Reynolds number is similarly interpolated from the donor grid.

Numerical Implementation

The numerical integration in each computational grid is performed using an upwind biased, factorized, iterative, implicit numerical scheme[12] given by

$$\begin{aligned}&\left[I + h_\xi\left(\nabla_\xi^b \tilde{A}_{i,k}^+ + \nabla_\xi^f \tilde{A}_{i,k}^-\right)\right]^p \\ &\times\left[I + h_\zeta\left(\nabla_\zeta^b \tilde{B}_{i,k}^+ + \Delta_\zeta^f \tilde{B}_{i,k}^- - Re^{-1}\delta_\zeta \tilde{M}_{i,k}\right)\right]^p \left(Q_{i,k}^{p+1} - Q_{i,k}^p\right) \\ &= -\left(Q_{i,k}^p - Q_{i,k}^n\right) - h_\xi\left(\hat{F}_{i+\frac{1}{2},k}^p - \hat{F}_{i-\frac{1}{2},k}^p\right) \\ &\quad - h_\zeta\left(\hat{G}_{i,k+\frac{1}{2}}^p - \hat{G}_{i,k-\frac{1}{2}}^p\right) + Re^{-1} h_\zeta\left(\hat{S}_{i,k+\frac{1}{2}}^p - \hat{S}_{i,k-\frac{1}{2}}^p\right)\end{aligned} \tag{2}$$

In Eq. (2), $h_\xi = \Delta\tau/\Delta\xi$, etc., and $\tilde{A}^\pm = (\partial\tilde{F}/\partial\tilde{Q})$, etc., are the flux Jacobian matrices and Δ, ∇, and δ are the forward, the backward, and the central difference operators, respectively. The superscript n denotes the time step and p refers to Newton subiterations within each time step. $\hat{F}_{i+1/2,k}$ and $\hat{G}_{i,k+1/2}$ are numerical inviscid fluxes, which are evaluated using Osher's third-order-accurate upwinding scheme. The inviscid flux Jacobian matrices $\tilde{A}$ and $\tilde{B}$ on the left-hand side are evaluated by the Steger–Warming flux-vector splitting. The viscous fluxes $\hat{S}_{i,k+1/2}$ are computed with second-order-accurate central differences.

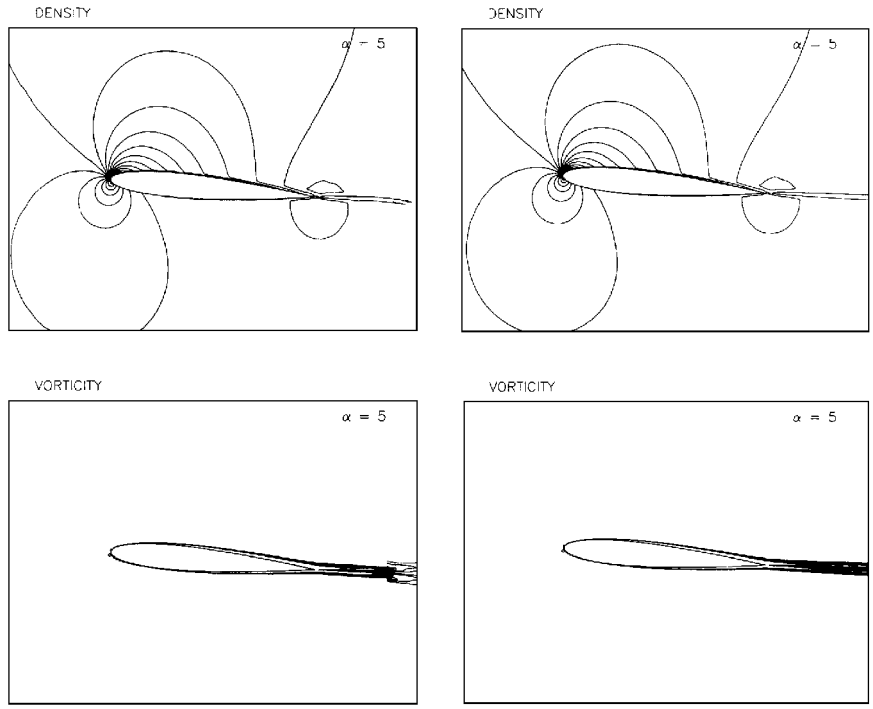
The numerical integration at the hole grid points is accommodated by i blanking, which replaces the corresponding coefficient matrices by an identity matrix and sets the right-hand side to zero. This process essentially imposes no change in the flow variables on the hole grid points. One-sided differencing is also used at the grid points next to the hole boundary points.

In the implementation of the Baldwin–Barth turbulence model, the governing equation for the turbulent Reynolds number is solved with first-order accuracy. The i blanking at the hole grid points and the intergrid boundary conditions are similarly applied.

Results and Discussion

The present intergrid boundary point localization and interpolation method has been applied to steady and unsteady flows over a NACA0012 airfoil. The numerical solutions obtained with overset grids were compared against single-grid solutions in terms of the distribution of flow variables and the aerodynamic loads.

First, the steady-state flowfield is computed at $\alpha = 5$-deg incidence, $Re = 3 \times 10^6$, and $M_\infty = 0.3$. The overset grids and the localization of the intergrid boundary points are shown in Fig. 2. A 137×51 size viscous C grid around the airfoil is overset onto a clustered 71×71 size rectangular background grid at $\alpha = 5$-deg incidence. There are 121 grid points around the airfoil, and the grid spacing normal to the airfoil surface varies from 2×10^{-5} at the leading edge to 5.5×10^{-5} at the trailing edge. As shown in Fig. 2b, the outer boundary points of the overset C grid, which are denoted by the + sign, are first localized on the background grid. The initial search starts arbitrarily from the middle of the background grid, and the outer boundary point at the lower wake cut is first localized by quadrant walking. The rest of the boundary points are then localized sequentially in the clockwise direction. This process defines the boundary points of the hole cut in the background grid. The hole in the background grid is then reduced by extending the boundaries



Overset grid **Single grid**

Fig. 3 Steady-state flowfield at $Re = 3 \times 10^6$ and $M_\infty = 0.3$.

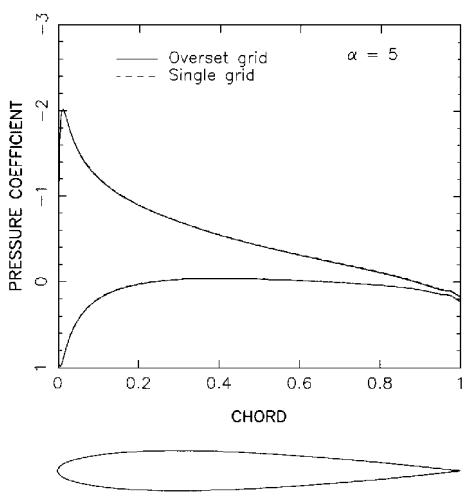


Fig. 4 Steady-state surface pressure distribution at $Re = 3 \times 10^6$ and $M_\infty = 0.3$.

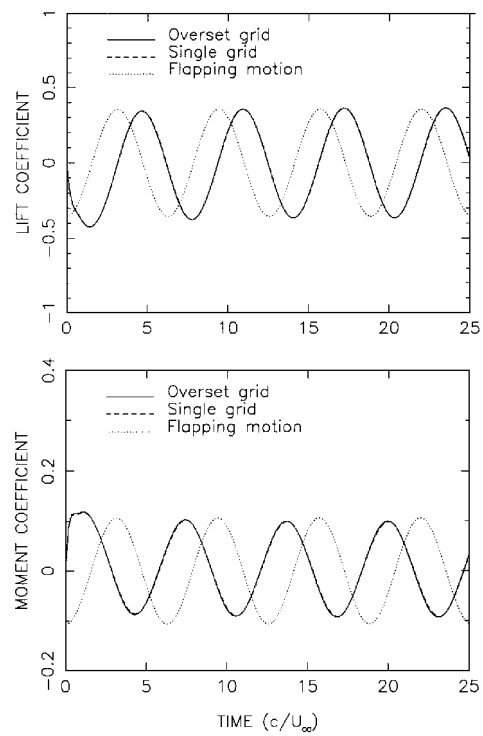


Fig. 5 Time history of the unsteady aerodynamic loads, flapping NACA0012 airfoil with $h = -0.10c\cos(\omega t)$, $k = 0.5$ at $Re = 3 \times 10^6$, and $M_\infty = 0.3$.

of the hole by one grid point inward in both the i and j directions. This process ensures that all of the boundary points defining the hole fall into the C grid. An indexing array is utilized to denote the final hole boundary points on the background grid. Next, as shown in Fig. 2c, the hole boundary points are similarly localized on the overset C grid.

The flowfield is assumed to be fully turbulent, and the eddy viscosity field is computed by the Baldwin–Barth one-equation turbulence model in each grid with the proper intergrid boundary conditions. In Fig. 3, the computed flowfield on the overset grid is compared with a single C-grid solution. The single C grid is of 181×81 size and is the extension of the C grid used in the overset grids. In the figures, the same contour levels are plotted in both cases. The overset grid solution yields a smooth and continuous distribution of the flow variables across the intergrid boundaries and agrees remarkably well with the single-grid solution. However, it is also noted that the interpolation of the flow variables from the highly clustered overset C grid at the downstream end of the wake cut onto the coarser background grid smears the vorticity in the wake. Figure 4 shows the computed surface pressure distribution. The excellent agreement between overset and single-grid solutions indicates that the flow variables are accurately interpolated across the intergrid boundaries.

In overset grid computations, the overhead due to the application of the intergrid boundary conditions is about 0.2% of the total CPU per time step. In the present computations, the total number of grid points in the overset grids is about 18% less than that of the single grid, and the overall CPU usage for the overset grid computations

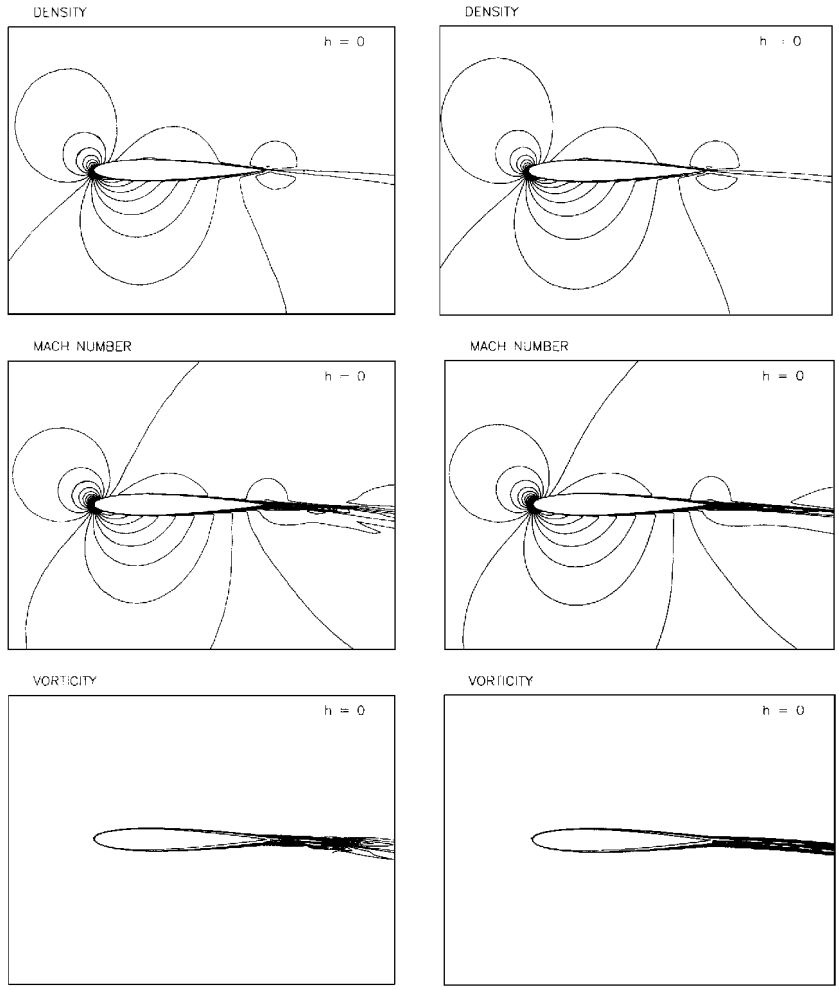


Fig. 6 Instantaneous flowfield at $t = 14.10$, flapping NACA0012 airfoil with $h = -0.10c\cos(\omega t)$, $k = 0.5$ at $Re = 3 \times 10^6$, and $M_\infty = 0.3$.

is about 22% less than that of the single-grid computations. Note that because the equations are solved implicitly with a block-tridiagonal solver, single-grid solutions are more CPU intensive per time step than overset grid solutions for the same total number of grid points.

Next, the same overset C grid is employed for an unsteady flow computation as the airfoil, which is now set at $\alpha = 0$-deg, undergoes a periodic flapping motion. The flapping motion is defined by $h = -0.10c\cos(\omega t)$ at a reduced frequency of $k = \omega c/2U_\infty = 0.5$, $Re = 3 \times 10^6$, and $M_\infty = 0.3$. Because the formulation of the Navier–Stokes solver is based on an inertial frame of reference, the flapping motion is implemented by moving the computational grid around it in the crossflow direction as prescribed. In the overset grid solution, the overset C grid is traversed over the stationary background grid. The intergrid boundary conditions are applied at every time step as the overset C grid is traversed. In the present unsteady computations, the Baldwin–Barth turbulence model had convergence problems and corrupted the solution. This behavior is attributed to the quasisteady solution algorithm in the model, which may require very small time steps at high-frequency motions. The Baldwin–Lomax algebraic eddy viscosity model, therefore, was employed only on the viscous C grid.

The unsteady computations were carried out for more than four cycles of the flapping motion. The time history of the lift and moment coefficients is given in Fig. 5. The unsteady flow prediction with the overset grids again agrees remarkably well with the single-grid solution at this relatively high-frequency flapping motion. In Fig. 6, the instantaneous flowfield as the airfoil passes the $h = 0$ position at $t = 14.10$, where the flapping velocity is maximum, is compared against the single-grid solution. The computed flowfields are again in excellent agreement.

Concluding Remarks

A method for the solution of unsteady Navier–Stokes equations on overset grids is presented. Intergrid boundary points are localized by the quadrant walking search algorithm and flow variables across the grids are interpolated as an integral part of the boundary point localization process in the main solver. No external interpolation data is stored. The steady and unsteady flowfield solutions are found to be in excellent agreement with the single-grid solutions. The overhead due to the application of the intergrid boundary conditions is about 0.2% of the total CPU per time step. In addition, overset grids may discretize the computational domains with less number of total grid points than single or patched grids. Therefore, the numerical solutions with overset grids using the present boundary point localization and interpolation method are highly accurate and efficient.

The present method is independent of numerical solution algorithms. It may easily be implemented on any two-dimensional, single-block flow solver to make it a multiblock, zonal solver regardless of the solution algorithm and the flow variables employed. The method may also be extended to three dimensional overset grids.

Acknowledgment

The author thanks Timothy J. Barth at the NASA Ames Research Center for providing the code for the Baldwin–Barth turbulence model.

References

[1]Thompson, J. F., and Weatherill, N. P, "Aspects of Numerical Grid Generation: Current Science and Art," AIAA Paper 93-3539, Aug. 1993.

[2]Steinbrenner, J. P., Chawner, J. R., and Fouts, C. L., "Multiple Block Grid Generation in the Interactive Environment," AIAA Paper 90-1602, June 1990.

[3]Thompson, J. F., "Composite Grid Generation Code for General Three-Dimensional Regions: the EAGLE Code," *AIAA Journal*, Vol. 26, No. 3, 1988, pp. 271, 272.

[4]Barth, J. T., "Aspects of Unstructured Grids and Finite-Volume Solvers for Euler and Navier–Stokes Equations," AGARD Rept. 787, Special Course Notes, May 1992.

[5]Morgan, K., Perarie, J., and Perio, J., "Unstructured Grid Methods for Compressible Flows," AGARD Rept. 787, Special Course Notes, May 1992.

[6]Mavripilis, D., "Adaptive Mesh Generation for Viscous Flows Using Delaunay Triangulation," Inst. for Computer Applications in Science and Engineering, ICASE Rept. No. 88-47, NASA Langley Research Center, Hampton, VA, June 1988.

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

[8]Tuncer, I. H., and Platzer, M. F., "Thrust Generation due to Airfoil Flapping," *AIAA Journal*, Vol. 34, No. 2, 1996, pp. 324–331.

[9]Tuncer, I. H., Ekaterinaris, J. A., and Platzer, M. F., "A Viscous–Inviscid Interaction Method for Unsteady Low Speed Airfoil Flows," *AIAA Journal*, Vol. 33, No. 1, 1995, pp. 151–154.

[10]Ekaterinaris, J. A., Cricelli, A., and Platzer, M. F., "A Zonal Method for Unsteady Viscous, Compressible Airfoil Flows," *Journal of Fluids and Structures*, Vol. 8, Jan. 1994, pp. 107–123.

[11]Baldwin, B. S., and Barth, T. J., "A One-Equation Turbulence Transport Model for High Reynolds Number Wall-Bounded Flows," NASA TM 102847, Aug. 1990.

[12]Rai, M. M., and Chakravarthy, S. R., "An Implicit Form of the Osher Upwind Scheme," *AIAA Journal*, Vol. 24, No. 5, 1986, pp. 735–743.

D. S. McRae
Associate Editor