AIAA JOURNAL
Vol. 37, No. 8, August 1999

Cartesian Cut-Cell Method for Axisymmetric Separating Body Flows

G. Yang,* D. M. Causon,† and D. M. Ingram‡
Manchester Metropolitan University, Manchester, England M1 5GD, United Kingdom

A method for the calculation of unsteady, axisymmetric compressible flows involving both fixed and separating bodies is presented. The method uses a Cartesian cut-cell mesh approach and high resolution upwind finite volume scheme. A stationary background Cartesian mesh is generated on the computational domain with complex solid geometries represented by different types of cut cell. Solid bodies are allowed to move across the mesh using a finite volume scheme modified to deal with moving boundary problems. The flow solver employed is a MUSCL–Hancock Godunov-type scheme in conjunction with an approximate Riemann solver of the Harten, Lax, and van Leer type (for flow interfaces) and an exact Riemann solver for a moving piston (for fixed or moving solid faces). A cell-merging technique is used to maintain numerical stability in the presence of arbitrarily small cut cells and to retain strict conservation at moving boundaries. The method is applied to a muzzle blast flow and muzzle break problems involving both fixed and separating bodies.

I. Introduction

NUMERICAL predictions of transient flows are important in many practical situations. Although such flows can be simulated numerically using current high resolution numerical schemes,[1–3] the simulation of unsteady flows involving one or two bodies in relative motion is still a challenging problem in computational fluid dynamics (CFD). Typical examples can be found in many applications including store separation from aircraft, a shell emerging from the muzzle of a gun, and sabot/projectile separation. Among these problems muzzle-blast flows and the related barrel-muzzle flows involving the motion of a projectile are interesting and complex flow phenomena to model. Highly time-dependent shock-on-shock interactions and relative body motion are a characteristic feature. Because such flows also present difficulty for the experimental fluid dynamicist, examining the possibility of modeling them numerically is reasonable.

In the past several years significant progress has been made in grid-generation techniques for unsteady compressible flows involving separating bodies, and unstructured grid methods in particular have become increasingly popular (see, e.g., Refs. 4–10). Full configurations and/or multiple bodies present particular gridding difficulties, and multiblock[11] or patched methods[12,13] are generally implemented in cases where structured meshes are to be used. However, if the geometry of an individual component is extremely complex, the problems of achieving a suitable body-fitted grid remain unresolved unless, for example, a set of overlapping meshes is employed based on the Chimera[12] or FAME[14] approaches. These require continuous identification of mesh intersection points from all overlapping meshes to transfer information between the various mesh patches as bodies move. Even unstructured mesh generation is not simple for complex configurations; such methods, although very powerful and flexible, require a significant amount of user intervention and may encounter difficulties preserving boundaries and reducing grid skewness as bodies move.[15,16] Furthermore, owing to the need for continual interpolation while remeshing, unstructured mesh methods may also suffer from the effects of numerical diffusion.

In this paper we adopt an alternative approach for the prediction of unsteady compressible flowfields involving both fixed and separating bodies. This approach involves the use of a Cartesian cut-cell technique and does not at any stage involve any movement of the mesh or remeshing process as the bodies move. Rather, a stationary background Cartesian mesh is employed, and solid bodies are simply cut out of it. When the bodies move, they are allowed to cross the stationary mesh by merging cells in the vicinity of solid body boundaries. In this way the Cartesian cut-cell approach deals with complex flows around complicated geometries, either stationary or in relative motion. Cartesian mesh methods have been used with significant success for computing flows about complicated geometries (see, e.g., Refs. 17–28). For example, Berger and LeVeque[25] and Quirk[26] have used adaptive mesh refinement (AMR) techniques to obtain highly resolved solutions of unsteady shock hydrodynamic problems, with recent extensions to detonation wave phenomena and multicomponent flows.[29] Schemes based on upwind methods on adaptively refined Cartesian meshes have been described previously by Berger and LeVeque[25] for unsteady flows. De Zeeuw and Powell,[20] Chaing et al.,[21] and Coirier and Powell[23] have also developed Cartesian mesh methods for steady and unsteady flows, employing time-step adaptation to increase computational efficiency. An assessment of the accuracy of Cartesian mesh approaches has been made by Coirier and Powell.[23] In the present paper the authors describe an extension of their previously published Cartesian cut-cell method developed initially for static bodies[24] to unsteady flows involving separating bodies. A similar approach has been reported recently by Falcovitz et al.[28] In the next section the basic principles of the Cartesian cut-cell technique are described. The numerical discretization and extension of the Cartesian cut-cell method to moving boundary problems are discussed in Sec. III. In Sec. IV, several example calculations for both fixed and separating body flows are described. Finally, in Sec. V, some conclusions are drawn.

II. Cartesian Cut-Cell Mesh

To cope with arbitrarily complex geometries, which may in general be stationary or moving relative to one and another, we use a Cartesian cut-cell approach. Solid bodies are simply cut out of a background Cartesian mesh, and their boundaries represented by different types of cut cell. Figure 1 shows the three main types of cell, all possibilities for the four basic subtypes and special subtypes of cut cell. Essentially, a two-dimensional Cartesian cut-cell mesh can be generated as follows:

1) Construction of a background Cartesian mesh: All mesh cells are initially flagged as flow cells or solid cells.

2) Locating the intersection points between the Cartesian mesh lines and the boundaries of solid bodies: The cells partially cut by

Received 22 May 1996; presented as Paper 96-1973 at the AIAA 27th Fluid Dynamics Conference, New Orleans, LA, 17–20 June 1996; revision received 30 November 1998; accepted for publication 5 February 1999.

*Research Assistant, Centre for Mathematical Modelling and Flow Analysis.
†Professor, Centre for Mathematical Modelling and Flow Analysis.
‡Senior Lecturer, Centre for Mathematical Modelling and Flow Analysis.

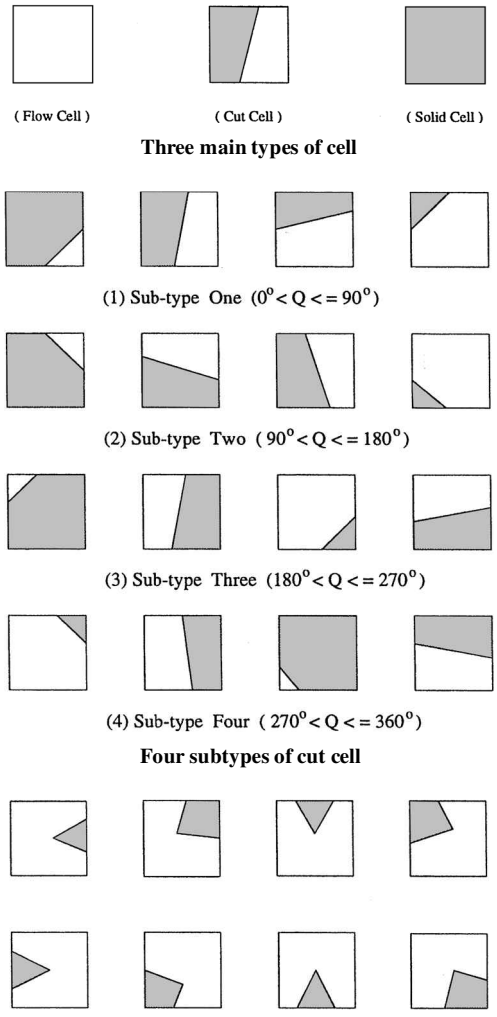

Fig. 1 Cartesian cut cell.

the boundaries are registered as cut cells. Accordingly, the subtype, volume, and other geometric information for each cut cell are determined.

3) Locating solid cells: Sweeps across the background mesh are then performed to identify which cells or rows of cells are bounded by solid or cut cells; these are registered as solid cells.

In practice, problems with the numerical stability of the flow solver may arise at arbitrarily small cut cells if a time step based on the flow cells is used. To avoid this problem a cell-merging technique[3,30] is implemented. A minimum acceptable cell volume $V_{\min}$ is specified, and when the volume of a cut cell becomes smaller than $V_{\min}$ a suitable neighboring cell is found to merge with it.

The choice for the minimum volume criterion $V_{\min}$ is based on a tradeoff between the time step and resolution accuracy. In our calculations $V_{\min}$ was allowed to be as large as one-half the flow cell size.

III. Numerical Scheme

A. Governing Equations

The Euler equations for two-dimensional/axisymmetric, compressible flow in a general moving reference frame may be written in integral form as

$$\frac{\partial}{\partial t}\int_{V_t} \boldsymbol{U}\,\mathrm{d}V + \oint_{S_t} \boldsymbol{F}\cdot\boldsymbol{n}\,\mathrm{d}S = \int_{V_t} \phi\boldsymbol{g}\,\mathrm{d}V \tag{1}$$

where $\boldsymbol{U}$ is the vector of conserved variables, $\boldsymbol{F}$ is the flux vector function, $\boldsymbol{g}$ is an axisymmetric flow source term vector function, and $\boldsymbol{n}$ is the outward unit vector normal to the boundary S_t, which encloses the time-dependent volume V_t. In axisymmetric flow $\phi = 1$, whereas in plane flow $\phi = 0$. $\boldsymbol{U}$, $\boldsymbol{F}$, and $\boldsymbol{g}$ are given by

$$\boldsymbol{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \qquad \boldsymbol{F} = \begin{bmatrix} \rho(\boldsymbol{v}-\boldsymbol{v}_s) \\ \rho u(\boldsymbol{v}-\boldsymbol{v}_s) + p\boldsymbol{i} \\ \rho v(\boldsymbol{v}-\boldsymbol{v}_s) + p\boldsymbol{j} \\ (e+p)(\boldsymbol{v}-\boldsymbol{v}_s) + p\boldsymbol{v}_s \end{bmatrix}$$

$$\boldsymbol{g} = \frac{1}{y}\begin{bmatrix} \rho v \\ \rho v u \\ \rho v v \\ (e+p)v \end{bmatrix} \tag{2}$$

where ρ, u, v, p, and e are density, x and y components of fluid velocity $\boldsymbol{v}$, pressure and total energy per unit volume, $\boldsymbol{i}$ and $\boldsymbol{j}$ are the Cartesian unit base vectors, and $\boldsymbol{v}_s$ is the velocity of the boundary of the control volume V_t. In this paper a stationary background Cartesian mesh is used to deal with moving boundary problems; therefore, $\boldsymbol{v}_s = 0$ at the interfaces of a flow cell, but at the moving solid face of a cell $\boldsymbol{v}_s$ is the velocity of the moving boundary. Finally, the governing equations are closed by the ideal-gas equation of state:

$$p = (\gamma - 1)[e - (\rho/2)(u^2 + v^2)] \tag{3}$$

where γ is the ratio of specific heats.

B. Numerical Discretization

The numerical scheme is a Godunov-type method. Higher-order spatial accuracy is achieved by using the MUSCL reconstruction approach of van Leer. Time integration is performed using an explicit predictor-corrector scheme by Hancock.[31] The resulting MUSCL-Hancock scheme has second-order accuracy in both time and space in smooth regions. The predictor step uses a nonconservative approach that defines an intermediate value over a half time interval $\Delta t/2$:

$$(V\boldsymbol{U})_{ij}^{n+\frac{1}{2}} = (V\boldsymbol{U})_{ij}^{n} - \frac{\Delta t}{2}\left[\sum_{k=1}^{m} \boldsymbol{F}(\boldsymbol{U}_k)\cdot\boldsymbol{S}_k^n - (V\boldsymbol{g})_{ij}^n\right] \tag{4}$$

where V is the cell volume, $\boldsymbol{S}$ is the cell face area vector, and m is the maximum number of cell faces.

The flux vector function $\boldsymbol{F}(\boldsymbol{U}_k)$ is evaluated at the midpoints of cell faces following a linear reconstruction of the flow solution within each cell, via

$$\boldsymbol{U}_k = \boldsymbol{U}_{ij}^n + \boldsymbol{r}_k\cdot\nabla\boldsymbol{U}_{ij}^n \tag{5}$$

where $\boldsymbol{r}_k$ is the normal distance vector from the cell centroid to face k and $\nabla\boldsymbol{U}_{ij}^n$ is a limited gradient vector in space (to be defined).

The corrector step of the scheme is fully conservative. The intermediate solution from the predictor step is used to define a set of left- and right-hand states for a series of Riemann problems. The solution of these Riemann problems provides a set of upwind interface fluxes that are used to update the flow solution over the time interval Δt, that is,

$$(V\boldsymbol{U})_{ij}^{n+1} = (V\boldsymbol{U})_{ij}^{n} - \Delta t\left[\sum_{k=1}^{m} \boldsymbol{F}\left(\boldsymbol{U}_k^{L,R}\right)\cdot\boldsymbol{S}_k^{n+\frac{1}{2}} - (V\boldsymbol{g})_{ij}^{n+\frac{1}{2}}\right] \tag{6}$$

where the upwind flux $\boldsymbol{F}(\boldsymbol{U}_k^{L,R})$ is obtained by solving a local Riemann problem normal to the cell interface. The left- and right-hand states at the interface k may be calculated by

$$\boldsymbol{U}_k^L = \boldsymbol{U}_{ij}^{n+\frac{1}{2}} + \boldsymbol{r}_k^L\cdot\nabla\boldsymbol{U}_{ij}^n, \qquad \boldsymbol{U}_k^R = \boldsymbol{U}_{lm}^{n+\frac{1}{2}} + \boldsymbol{r}_k^R\cdot\nabla\boldsymbol{U}_{lm}^n \tag{7}$$

where l and m relate to the right neighboring cell and $\boldsymbol{r}_k^L$ and $\boldsymbol{r}_k^R$ are the normal distance vectors from the appropriate left/right cell centroids to the face k.

To solve the Riemann problem, an improved version of the Harten, Lax, and van Leer (HLL) approximate Riemann solver, called HLLC,[32] that more accurately resolves contact surfaces is used for the fluxes at fluid interfaces of a cell, and an exact Riemann solution for a moving piston is used at a static or moving solid boundary (or face) of a cut cell.

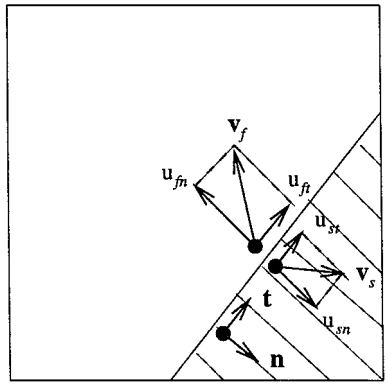

Fig. 2 Cut cell at a moving boundry.

C. Riemann Problem on a Moving Boundary

Consider a cut cell with a moving boundary as shown in Fig. 2, where $\boldsymbol{v}_f$ is the reconstructed flow velocity at the solid boundary and $\boldsymbol{v}_s$ is the solid boundary velocity. Now, projecting $\boldsymbol{v}_f$ and $\boldsymbol{v}_s$ into the normal direction $\boldsymbol{n}$ and tangential direction $\boldsymbol{t}$ of the solid boundary, we have

$$u_{\mathrm{fn}} = \boldsymbol{v}_f \cdot \boldsymbol{n}, \qquad u_{\mathrm{ft}} = \boldsymbol{v}_f \cdot \boldsymbol{t}, \qquad u_{\mathrm{sn}} = \boldsymbol{v}_s \cdot \boldsymbol{n}, \qquad u_{\mathrm{st}} = \boldsymbol{v}_s \cdot \boldsymbol{t} \tag{8}$$

In the tangential direction any difference between u_{ft} and u_{st} can be regarded as a shear wave superimposed on a contact discontinuity. In the normal direction, however, the Riemann solution for a moving piston is incorporated where the solution for u_n^* can be found immediately, that is,

$$u_n^* = u_{\mathrm{sn}} \tag{9}$$

and p_n^* can be obtained by comparing the two velocities u_{fn} and u_{sn}.

If $u_{\mathrm{sn}} > u_{\mathrm{fn}}$, both left and right moving waves are rarefactions. Then, p_n^* can be obtained from the rarefaction relations

$$\frac{p_n^*}{p_f} = \left(1 - \frac{\gamma - 1}{2}\frac{u_n^* - u_{\mathrm{fn}}}{a_f}\right)^{2\gamma/(\gamma-1)} \tag{10}$$

where p_f and a_f are the reconstructed pressure and sound speed, respectively.

If $u_{\mathrm{sn}} \le u_{\mathrm{fn}}$, both left and right moving waves are shocks. Thus, p_n^* can be obtained from the normal shock relations

$$\frac{p_n^*}{p_f} = 1 + \frac{\left(u_{\mathrm{fn}} - u_n^*\right)^2}{2C_f^2} + \frac{u_{\mathrm{fn}} - u_n^*}{2C_f^2}\sqrt{\frac{8\gamma}{\gamma+1}C_f^2 + \left(u_{\mathrm{fn}} - u_n^*\right)^2} \tag{11}$$

where $C_f = 2p_f/(\gamma+1)\rho_f$ and ρ_f is the reconstructed density.

The flux side vector function on the moving boundary is then

$$\boldsymbol{F}_n^* \cdot \boldsymbol{S} = \begin{pmatrix} 0 \\ p_n^* S_y \\ p_n^* S_x \\ p_n^* u_{\mathrm{sn}} |\boldsymbol{S}| \end{pmatrix} \tag{12}$$

where S_x and S_y are the x and y components of the solid face side vector $\boldsymbol{S}$.

D. Gradient Calculation

Calculation of a limited gradient vector $\nabla \boldsymbol{U}_{ij}^n$ is straightforward on flow cells away from a solid boundary. For cells near a solid boundary, a different approach is needed. Because a solid boundary can be either static or moving, appropriate boundary conditions must be implemented in the gradient calculation. Here, we use reflection boundary conditions at the solid boundary where the variables in a fictional cell R (Fig. 3) are

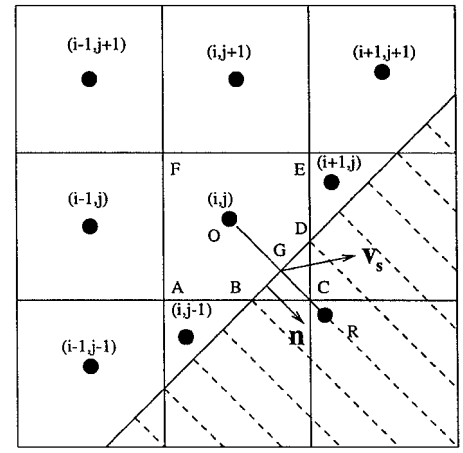

Fig. 3 Calculation of gradients on a cut cell.

$$\rho_R = \rho_{ij}, \qquad \boldsymbol{v}_R = \boldsymbol{v}_{ij} - 2(\boldsymbol{v}_{ij}\cdot\boldsymbol{n})\boldsymbol{n} + 2(\boldsymbol{v}_s\cdot\boldsymbol{n})\boldsymbol{n}$$
$$p_R = p_{ij}, \qquad e_R = p_R/(\gamma-1) + \tfrac{1}{2}\rho_R|\boldsymbol{v}_R|^2 \tag{13}$$

The gradients on cut cell (i, j) in Fig. 3 can be of two types: fluid and solid. The fluid gradients can be calculated as for flow cells, i.e.,

$$\boldsymbol{U}_x^f = G(\boldsymbol{U}_{i+1,j} - \boldsymbol{U}_{i,j}, \boldsymbol{U}_{i,j} - \boldsymbol{U}_{i-1,j})$$
$$\boldsymbol{U}_y^f = G(\boldsymbol{U}_{i,j+1} - \boldsymbol{U}_{i,j}, \boldsymbol{U}_{i,j} - \boldsymbol{U}_{i,j-1}) \tag{14}$$

where G is a slope limiter function that is used to prevent unphysical over- or under-shoots. The limiter function may take various forms, e.g., the Superbee limiter:

$$G(a,b) = s\cdot\max[0, \min(2|b|, s\cdot a), \min(|b|, 2s\cdot a)]$$
$$s = \mathrm{sign}(b) \tag{15}$$

or the van Leer limiter:

$$G(a,b) = \frac{a|b| + |a|b}{|a| + |b|} \tag{16}$$

Then, the solid gradients are estimated similarly, that is,

$$\boldsymbol{U}_x^s = G(\boldsymbol{U}_R - \boldsymbol{U}_{i,j}, \boldsymbol{U}_{i,j} - \boldsymbol{U}_{i-1,j})$$
$$\boldsymbol{U}_y^s = G(\boldsymbol{U}_{i,j+1} - \boldsymbol{U}_{i,j}, \boldsymbol{U}_{i,j} - \boldsymbol{U}_R) \tag{17}$$

Once the two types of gradients have been calculated, a length-average technique is used to obtain unique components of the limited gradient in the cut cell:

$$\boldsymbol{U}_x = \frac{\Delta y_s \boldsymbol{U}_x^s + \Delta y_f \boldsymbol{U}_x^f}{\Delta y}, \qquad \boldsymbol{U}_y = \frac{\Delta x_s \boldsymbol{U}_y^s + \Delta x_f \boldsymbol{U}_y^f}{\Delta x} \tag{18}$$

where Δx and Δy are the uncut cell side lengths in x and y directions, respectively. A limited gradient vector in each cut cell can then be defined as

$$\nabla \boldsymbol{U}_{ij} = \begin{bmatrix} \boldsymbol{U}_x \\ \boldsymbol{U}_y \end{bmatrix} \tag{19}$$

Given the limited gradient vector $\nabla \boldsymbol{U}_{ij}$, a reconstructed solution vector $\boldsymbol{U}(x, y)$ can be found anywhere within a cut cell from

$$\boldsymbol{U}(x, y) = \boldsymbol{U}_{ij} + \boldsymbol{r}\cdot\nabla\boldsymbol{U}_{ij} \tag{20}$$

where $\boldsymbol{r}$ is the normal distance vector from the cell centroid to any specific interface or solid boundary.

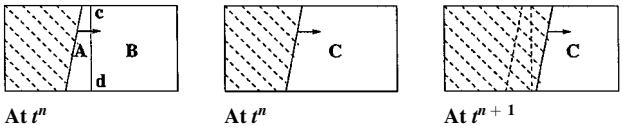

Fig. 4 Cell-merging technique for a moving boundry.

E. Extension to Moving Body Problems

Body motion can be prescribed or determined by the aerodynamic forces acting on the body surface. During each time step, the new position of the moving body is determined twice: at the half time step and at the full time step. Once the new position of the moving body has been found, a new list of cut cells is determined. By comparing the new cut-cell list with the old one, changes in cell type near the boundary of the moving body can be found. In general, the cell change information falls into four categories: 1) cut cell becomes a solid cell; 2) cut cell becomes an uncut flow cell; 3) cut cell remains unchanged; and 4) uncut flow cell becomes a cut cell.

Categories three and four do not cause any problems. However, in cases where a cut cell becomes solid (category one), the volume of the cell at the end of time step is zero; obviously, this will lead to problems within the finite volume scheme. In the case of category two, failure to consider the new-born cell will result in strict conservation being lost.

Appropriate strategies are therefore needed to ensure that both conservation and numerical stability are maintained when using a time-dependent, explicit, finite volume integration scheme for moving boundary problems. In general, these problems can be solved by using a cell-merging technique.[2,3] The basic idea is to merge a small cut cell with one or several neighboring cells, so that any interface between the merged cells is ignored and the waves are allowed to travel in the larger merged cell without reducing the global value of Δt.

For example, a time step Δt based on flow cell B will be too large for cut cell A, which will also become solid after Δt (see Fig. 4). To illustrate the implementation of the cell-merging technique, we consider the corrector step of the MUSCL-Hancock scheme. First, the updates for cells A and B are computed as usual:

$$\Delta(V\boldsymbol{U})_{A,B} = -\Delta t\left[\sum_{k=1}^{m}\boldsymbol{F}\left(\boldsymbol{U}_k^{L,R}\right)\cdot\boldsymbol{S}_k^{n+\frac{1}{2}} - (V\boldsymbol{g})_{ij}^{n+\frac{1}{2}}\right]_{A,B} \tag{21}$$

Then, we ignore the interface between cells A and B and update the merged cell C simply by combining the volume updates of cells A and B:

$$\Delta(V\boldsymbol{U})_C = \Delta(V\boldsymbol{U})_A + \Delta(V\boldsymbol{U})_B \tag{22}$$

The fluxes on the interface $|cd|$ between cells A and B cancel out automatically because the flux calculation is conservative. The conserved variable $\boldsymbol{U}$ for cell C at time t^{n+1} is

$$V_C^{n+1}\boldsymbol{U}_C^{n+1} = V_A^n\boldsymbol{U}_A^n + V_B^n\boldsymbol{U}_B^n - \Delta t\left[\sum_{k=1}^{m_A}\boldsymbol{F}\left(\boldsymbol{U}_k^{L,R}\right)\cdot\boldsymbol{S}_k^{n+\frac{1}{2}} + \sum_{k=1}^{m_B}\boldsymbol{F}\left(\boldsymbol{U}_k^{L,R}\right)\cdot\boldsymbol{S}_k^{n+\frac{1}{2}} - (V\boldsymbol{g})_A^{n+\frac{1}{2}} - (V\boldsymbol{g})_B^{n+\frac{1}{2}}\right] \tag{23}$$

Although cut cell A finally disappears, its contribution to the mass, momentum, and energy will be transferred into neighboring cells so that conservation is automatically maintained.

To prevent a single cut cell becoming solid without merging with neighboring cells, an appropriate estimate of time step is introduced:

$$\Delta t_{x,y} = \frac{\sqrt{V_{\min}}}{\max(|\boldsymbol{v}_s|_{x,y}, |\boldsymbol{v}|_{x,y}) + a} \tag{24}$$

$$\Delta t = \mathrm{CFL}\ \min(\Delta t_x, \Delta t_y) \qquad (\mathrm{CFL} < 1.0) \tag{25}$$

where a is sound speed.

IV. Numerical Results

A. Open-Ended Shock Tube

A Mach 1.76 shock wave emerging from an open-ended shock tube has been studied extensively in the past, both experimentally[33] and numerically.[1,3] In the numerical studies a Mach 1.76 shock wave was specified at the exit of the shock tube. In our simulation a moving piston was used to create a shock of the same strength moving down the tube. The piston was placed initially at a distance $5.5D$ (D is the diameter of the shock tube) upstream of the tube exit. The gas in front of the piston was assumed to be quiescent while a postshock state corresponding to Mach 1.76 conditions was imposed behind the piston. The piston was set in motion impulsively at the speed of the postshock gas. A Cartesian mesh with 450×150 cells was used on a computational domain of $9D \times 3D$. Figure 5a shows a shock wave positioned at the exit of the shock tube at $t_0 = 1.4128$ ms. A sequence of shock diffractions was obtained at approximately the same times as the corresponding experimental ones in Ref. 10. Figure 5b shows computed density contours at two time stages, which compare favorably with the experimental interferogrammes (not shown). Figure 6 shows a comparison of calculated and measured overpressure variation with time on the

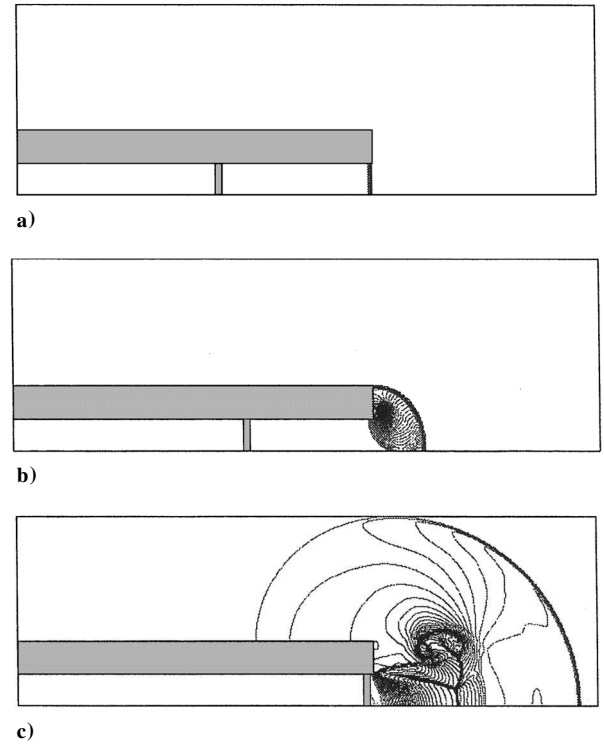

Fig. 5 Shock-tube problem: density contours at a) $t_0 = 1.4128$ ms, b) $t_0 + 0.2$ ms, and c) $t_0 + 1.0$ ms.

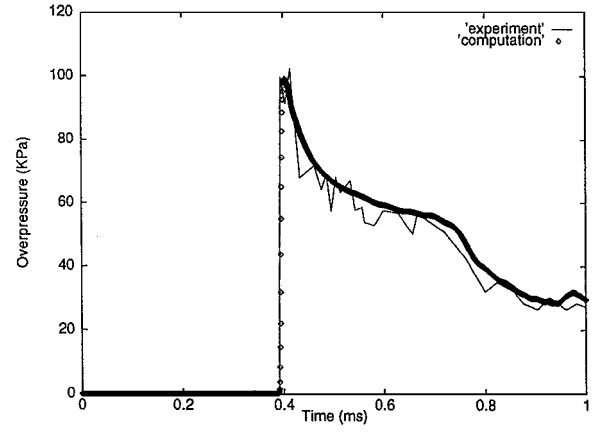

Fig. 6 Shock-tube problem: computed and measured overpressure on the centerline, 1.5 diameters downstream of the exit.

centerline of the tube, 1.5 diameters downstream of the exit. From the computed results one can see that the shock wave is accurately produced, and all features of the shock wave diffraction process, such as the formation of a Mach disk, contact surface, and vortex, are well-resolved.

B. Fifteen-Degree Wedge Flow at Mach 2

This problem has been used to validate the Cartesian cut-cell approach for moving body problems. The geometry is a two-dimensional channel with a 15-deg wedge on its lower wall. The channel is usually taken to be 1 unit high and 3 units long. However, for our simulation the length was increased to 7.5 units. Two computational meshes were employed with 300×40 cells and 600×80 cells. First, the case of a fixed wedge placed in an impulsively started Mach 2 flow was calculated. The wedge was located at $x = 7.25$ units with height 0.2588 units so that a 15-deg compression corner and a 15-deg expansion corner were formed by the wedge geometry. If a freestream flow at Mach number $M_\infty = 2$ passes through the channel from right to left, an attached oblique shock is formed at the compression corner. The oblique shock is then reflected at the upper wall of the channel and weakened by an expansion fan originating from the expansion corner. Downstream, multiple-shock reflections occur at the upper and lower walls of the channel until the shock exits from the domain (Fig. 7).

We now assume instead that the wedge moves suddenly at the same speed into quiescent gas, and hence expect an identical flowfield. The 15-deg wedge is initially located at $x = 1.25$ and suddenly starts to move with Mach number $M_w = 2$. Like the fixed wedge in a Mach 2 freestream, an attached oblique shock is gradually produced and reflected at the upper and lower walls. At time $t = 3$ the moving wedge is in exactly the same position as the fixed wedge. Line contours of pressure from the fine grid calculation for both the fixed and moving cases are shown in Fig. 7. Figure 8 shows a comparison of pressure and density distributions along the surface of the wedge at $t = 3$ obtained on both the coarse and fine meshes. We can see there is close agreement between the results for the wedge moving at $M_w = 2$ into quiescent gas and the fixed wedge placed in an impulsively started freestream at $M_\infty = 2$ and that the results are largely grid independent. These results demonstrate the robustness and accuracy of the Cartesian cut-cell method for both fixed and moving body problems and confirm that changing the frame of reference does not materially affect the solution.

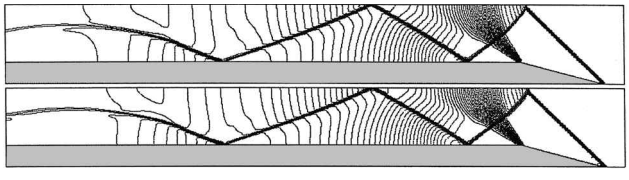

Fig. 7 Fifteen-degree wedge flow problem: pressure contours at $t = 3$; fixed wedge in a Mach 2.0 freestream (top); wedge moving at Mach 2.0 into quiescent gas (bottom).

C. Muzzle Brake Flowfields

Widhopf et al.[34] presented experimental data and numerical results for a muzzle-blast wave interaction with a moving projectile and multielement brakes. To demonstrate that the present method can cope with complex compressible flows around multielement axisymmetric bodies in relative motion, this problem was also considered. In the earlier studies of Widhopf, a laboratory model of a 30-mm cannon with an attached axisymmetric double brake was used to provide data for practical design purposes. A projectile, initially located inside a barrel, is propelled by high-pressure propellant gas. As the projectile emerges from the barrel, the escaping propellant gas impinges on the baffle brake. In our numerical study sonic conditions were set at the muzzle exit, and so the time-dependent overpressure on the brake is characterized by a large amplitude wave relaxing to a steady-state distribution. The details of the geometry and initial conditions can be found in Ref. 34. In our calculations a uniform Cartesian mesh with 240×96 cells was used on a $12D \times 4.8D$ domain ($D =$ 30 mm). Figure 9 shows a partial view of the Cartesian mesh, muzzle brakes, and the moving projectile at time 0.18 ms. Figure 10 shows three sets of Mach number contours at times: 0.18, 0.28, and 0.37 ms. The first set of contours show the emergence of the propellant gases behind the projectile whereas the second and third sets show the gas emergence from the first and second brake, respectively. Figure 11 shows the calculated overpressure histories and comparison of computed and measured peak and steady-state overpressures on the first brake. Except at locations close to the brake gap, both the peak and steady-state overpressures are in very good agreement with the measured data.

D. Muzzle Sabot/Projectile Flowfields

Finally, a sabot/projectile emerging from the muzzle of a gun was computed. The geometry of the sabot/projectile model was a 120-mm diam circular body 650 mm in length. Initially, the circular body was set inside the muzzle. Behind the circular body a Mach 5.6 propellant gas was specified with the gas in front of it assumed to be in a quiescent state. The computational domain was

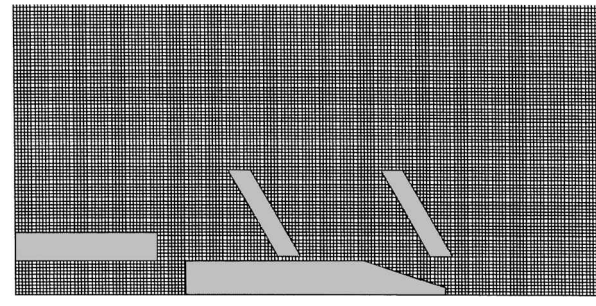

Fig. 9 Muzzle brake problem: static mesh, muzzle brakes, and position of projectile at time 0.18 ms.

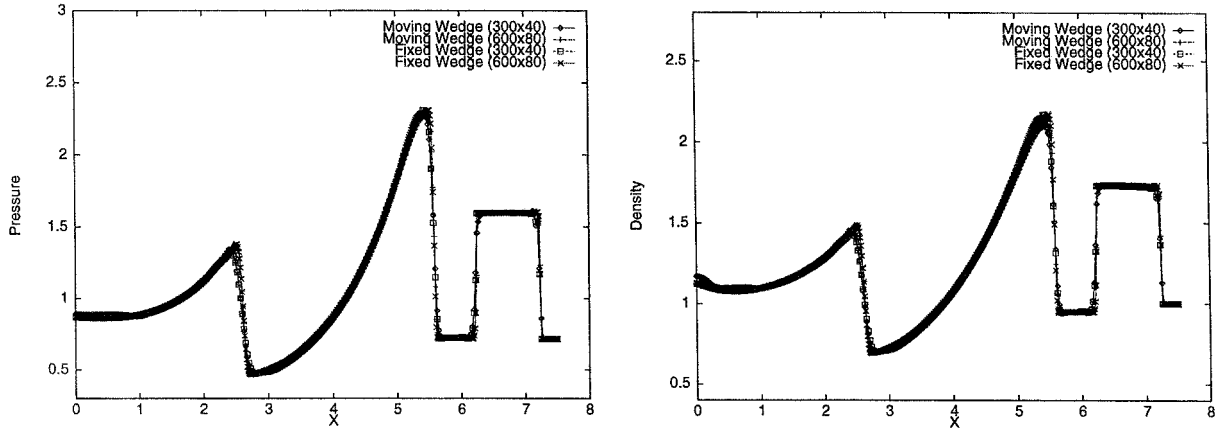

Fig. 8 Fifteen-degree wedge flow problem: pressure (left) and density (right) distributions on fixed and moving wedge surface at $t = 3$ and effect of mesh refinement.

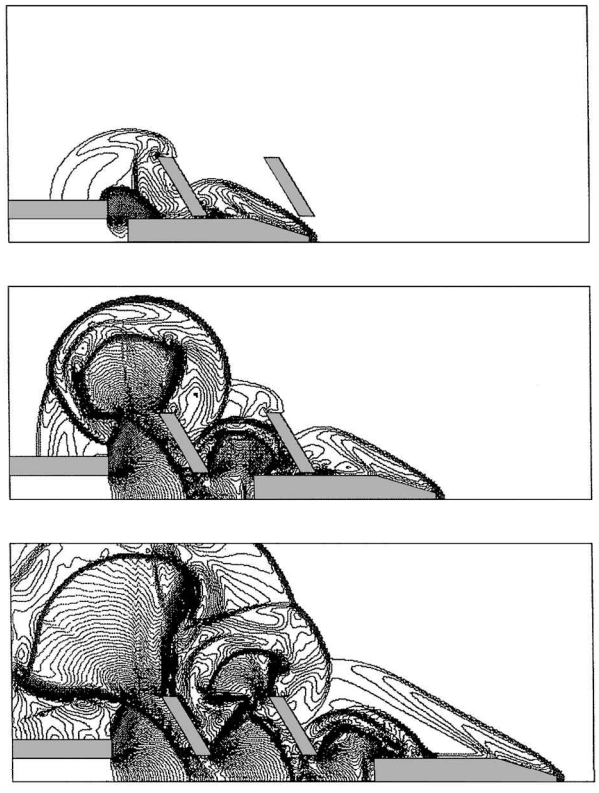

Fig. 10 Muzzle brake problem: Mach number contours at 0.18, 0.28, and 0.37 ms.

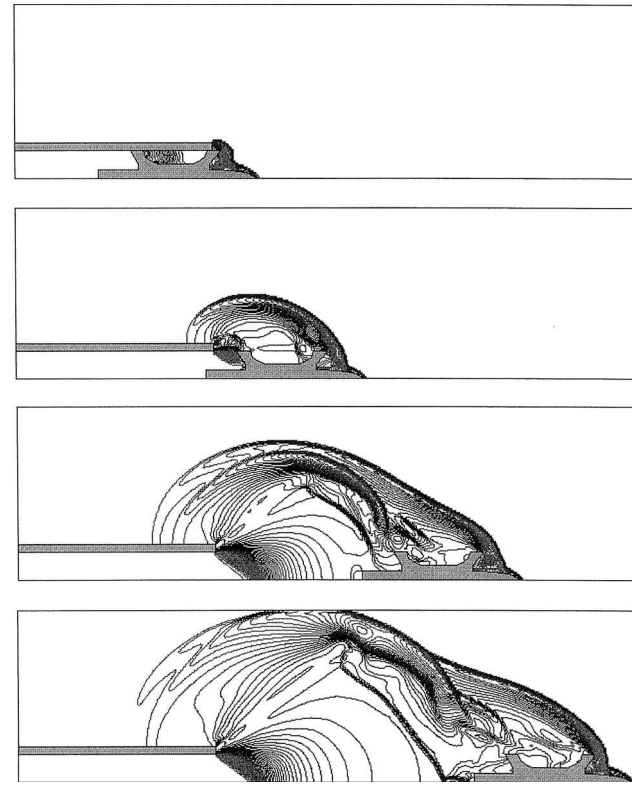

Fig. 12 Sabot/projectile emerging from the muzzle at different times: 0.38, 5.65, 10.59, and 14.12 ms.

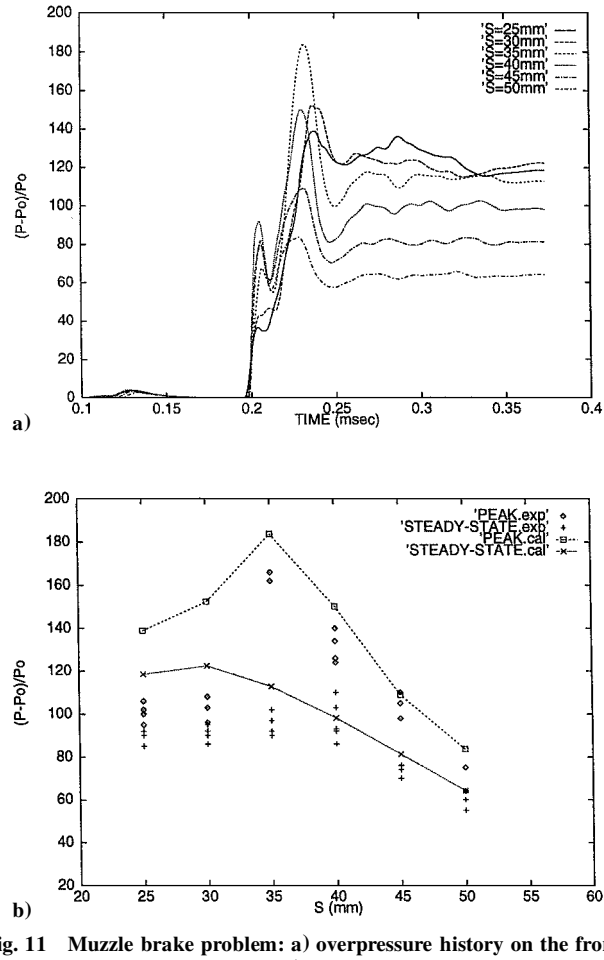

Fig. 11 Muzzle brake problem: a) overpressure history on the front face of the first baffle brake and b) computed and measured peak and steady-state overpressure on the brake.

extended to a 256×72 mm^2 area using a 320×90 cell Cartesian mesh. The circular body was set in motion impulsively at the speed of the propellant gas, giving rise to a precursor shock. A series of density contour plots in Fig. 12 show the development of the emerging precursor shock and uncorking propellant gas. At the leading surfaces of the sabot and penetrator, the precursor shock is partially attached and detached. At a time of 0.38 ms, the projectile can be seen just emerging from the muzzle with the precursor flow expanding around the tube; by 5.65 ms the sabot/projectile is moving out of the tube, and the propellant gas emerges behind the projectile. As the projectile moves away from the tube, two bow shocks are generated around the top of the sabot, which gradually separate with the sabot. At 14.12 ms the first bow shock combines with the precursor shock to produce a hot-spot. Meanwhile, the second bow shock reaches the axis, and a forward-traveling Mach disk can be seen in the expanding driver gas region. No experimental data are available for comparison purposes for this second example.

V. Conclusions

A method for the computation of unsteady compressible flows involving both fixed and separating bodies has been presented. Based on a Cartesian cut-cell approach and a high-resolution upwind finite volume scheme, the present method proceeds in two steps for moving-body problems. First, a stationary background Cartesian mesh is constructed on the computational domain, and solid bodies are simply cut out of it. In this way the Cartesian cut-cell approach can be seen as a viable alternative to unstructured mesh methods for coping with extremely complicated geometries. Second, solid bodies are allowed to cross the background mesh in relative motion by means of a cell-merging technique. In this way the present Cartesian cut-cell method can deal with complex compressible flows around arbitrarily complicated, multielement geometries, either stationary or in relative motion, without using moving meshes. Problems such as mesh distortion, body motion restriction, etc., that may occur when using other mesh approaches are completely avoided. The computational overheads of the cut-cell approach in terms of the required checks on critical cells and recalculation of cell areas and interface lengths as the bodies move through the static mesh are

typically on the order of 5%. Compared with a conventional body fitted structured grid, cell areas and side-vector quantities are trivial to evaluate and are thus calculated more quickly. A fully unstructured method, on the other hand, would require local remeshing in the vicinity of the moving body together with interpolation of solution data and would thus be more costly than the present cut-cell approach, which only requires changes to cells cut by the body contour.

The method has been applied to muzzle blast and related flows with rapidly moving shocks and separating bodies and has been validated by recourse to relevant experimental data. The computed results suggest that the present approach can provide an accurate and effective numerical prediction tool for the simulation of unsteady compressible flowfields involving both fixed and separating bodies. A three-dimensional version of the present method is currently under development.

References

[1]Wang, J. C. T., and Widhopf, G. F., "Numerical Simulation of Blast Flowfields Using a High Resolution TVD Finite Volume Scheme," *Computers and Fluids*, Vol. 18, No. 1, 1990, pp. 103–137.

[2]Quirk, J. J., "An Alternative to Unstructured Grids for Computing Gas Dynamic Flows Around Arbitrarily Complex Two-Dimensional Bodies," *Computers and Fluids*, Vol. 23, No. 1, 1994, pp. 125–142.

[3]Chiang, Y., van Leer, B., and Powell, K. G., "Simulation of Unsteady Inviscid Flow on an Adaptively Refined Cartesian Grid," AIAA Paper 92-0443, Jan. 1992.

[4]Brenner, P., "Three Dimensional Aerodynamics with Moving Bodies Applied to Solid Propellant," AIAA Paper 91-2304, Jan. 1991.

[5]Batina, J. T., "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex Aircraft Aerodynamic Analysis," *AIAA Journal*, Vol. 29, No. 3, 1991, pp. 327–333.

[6]Lohner, R., and Baum, D., "Adaptive h-Refinement on 3D Unstructured Grids for Transient Problems," *International Journal of Numerical Methods in Fluids*, Vol. 14, No. 12, 1992, pp. 1407–1419.

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

[8]Weatherill, N. P., Hassan, O., Marchant, M. J., and Marcum, D. L., "Adaptive Inviscid Flow Solutions for Aerospace Geometries Using Efficently Generated Unstructured Tetrahedral Meshes," AIAA Paper 93-3390, Jan. 1993.

[9]Trepanier, J. Y., Reggio, M., Paraschivoiu, M., and Camarero, R., "Unsteady Euler Solutions for Arbritrary Moving Bodies and Boundaries," *AIAA Journal*, Vol. 31, No. 10, 1993, pp. 1869–1876.

[10]Voinovich, P., "Three Dimensional Locally Adaptive Unstructured Unsteady Euler Code," Advanced Technology Center, St. Petersburg, Russia, 1994 (unpublished).

[11]Shaw, J. A., Forsey, C. R., Weatherill, N. P., and Rose, K. E., "A Block Structured Mesh Generation Technique for Aerodynamic Geometries," *Numerical Grid Generation in CFD*, edited by J. A. Hauser and C. Taylor, Pineridge Press, Swansea, Wales, UK, 1986, pp. 500–512.

[12]Steger, J. L., Dougherty, F. C., and Benek, J. A., "A Chimera Grid Scheme," *Advances in Grid Generation, American Society of Mechanical Engineers*, American Society of Mechanical Engineers, 1983, pp. 59–69.

[13]Meakin, R. L., "An Efficent Means of Adaptive Refinement Within Systems of Overset Grids," AIAA Paper 95-1722, June 1995.

[14]Blaylock, T. A., and Onslow, S. H., "Application of the FAME Method to Store Release Prediction," *Computational Fluid Dynamics '94*, Wiley, 1994.

[15]Prodert, E. J., Hassan, O., Morgan, K., and Peraire, J., "An Adaptive Finite Element Method for Transient Compressible Flows with Moving Boundaries," *International Journal for Numerical Methods in Engineering*, Vol. 32, No. 4, 1991, pp. 751–765.

[16]Lohner, R., "Adaptive Remeshing for Transient Problems," *Computer Methods in Applied Mechanics and Engineering*, Vol. 75, No. 1–3, 1989, pp. 195–214.

[17]Clarke, D. K., Salas, M. D., and Hassan, H. A., "Euler Calculations for Multielement Airfoils Using Cartesian Grids," *AIAA Journal*, Vol. 24, No. 3, 1986, pp. 353–358.

[18]Epstein, B., Luntz, A. L., and Nachson, A., "Cartesian Euler Method for Arbitrary Aircraft Configurations," *AIAA Journal*, Vol. 30, No. 3, 1992, pp. 679–687.

[19]LeVeque, R. J., "High Resolution Finite Volume Methods on Arbritary Grids via Wave Propagation," *Journal of Computational Physics*, Vol. 78, 1988, pp. 36–63.

[20]De Zeeuw, D., and Powell, K. G., "An Adaptive Refined Cartesian Mesh Solver for the Euler Equations," *Journal of Computational Physics*, Vol. 104, No. 1, 1993, pp. 56–68.

[21]Chaing, Y., van Leer, B., and Powell, K. G., "Simulation of Unsteady Inviscid Flow on an Adaptively Refined Cartesian Grid," AIAA Paper 93-3335, June 1993.

[22]Berger, M. J., Aftosmis, M. J., and Melton, J. E., "Adaptation and Surface Modelling for Cartesian Mesh Methods," AIAA Paper 95-1725, 1995.

[23]Coirier, W. J., and Powell, K. G., "An Accuracy Assessment of Cartesian Mesh Approaches for the Euler Equations," *Journal of Computational Physics*, Vol. 117, 1995, pp. 121–131.

[24]Yang, G., Causon, D. M., Ingram, D. M., Saunders, R., and Batten, P., "A Cartesian Cut Cell Method for Compressible Flows—Part A: Static Body Problems," *Aeronautical Journal*, Vol. 101, No. 1002, 1997, pp. 47–56.

[25]Berger, M. J., and LeVeque, R. J., "An Adaptive Cartesian Mesh Algorithm for the Euler Equations in Arbitrary Geometries," AIAA Paper 89-1930, June 1989.

[26]Quirk, J. J., "An Alternative to Unstructured Grids for Computing Gas Dynamic Flows Around Arbitrarily Complex Two-Dimensional Bodies," *Computers and Fluids*, Vol. 23, No. 1, 1994, pp. 125–142.

[27]Slack, D. C., and Walters, R. W., "An Interactive Remeshing Algorithm for the Two Dimensional Euler Equations," AIAA Paper 90-0331, Jan. 1990.

[28]Falcovitz, J., Alfandary, G., and Hanoch, G., "A Two Dimensional Conservation Laws Scheme for Compressible Flows with Moving Boundaries," *Journal of Computational Physics*, Vol. 138, 1997, pp. 83–102.

[29]Quirk, J. J., "Godunov-Type Schemes Applied to Detonation Flows," NASA CR-191447/ICASE Rept. 93-15, Inst. for Computer Application in Science and Engineering, Hampton, VA, Jan. 1993.

[30]Clarke, D. K., Salas, M. D., and Hassan, H. A., "Euler Calculations for Multielement Airfoils Using Cartesian Grids," *AIAA Journal*, Vol. 24, No. 3, 1986, pp. 353–358.

[31]Van Leer, B., "On the Relation Between the Upwind-Differencing Schemes of Godunov, Engquist-Osher and Roe," *SIAM Journal on Scientific and Statistical Computing*, Vol. 5, No. 1, 1984, pp. 1–20.

[32]Toro, E. F., Spruce, M., and Speares W., "Restoration of the Contact Surface in the HLL Riemann Solver," *Shock Waves*, Vol. 4, No. 1, 1994, pp. 25–34.

[33]Schmidt, E. M., and Duffy, S., "Noise from Shock Tube Facilities," AIAA Paper 85-0049, June 1985.

[34]Widhopf, G. F., Buell, J. C., and Schmidt, E. M., "Time-Dependent Near Field Muzzle Brake Flow Simulations," AIAA Paper 82-0973, Jan. 1982.

D. S. McRae
Associate Editor