Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **1**, No. 5, pp. 664-682

DOI: 10.4208/aamm.09-m0912
October 2009

The Ghost Cell Method and its Applications for Inviscid Compressible Flow on Adaptive Tree Cartesian Grids

Jianming Liu[1,2,3,*], Ning Zhao[1] and Ou Hu[1]

[1] *College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China*

[2] *Department of Mathematics, Xuzhou Normal University, Xuzhou 221116, China*

[3] *Faculty of Technology, De Montfort University, Leicester LE1 9BH, England*

Received 05 January 2009; Accepted (in revised version) 05 June 2009
Available online 30 July 2009

Abstract. In this paper, an immersed boundary algorithm is developed by combining the ghost cell method with adaptive tree Cartesian grid method. Furthermore, the proposed method is successfully used to evaluate various inviscid compressible flow with immersed boundary. The extension to three dimensional cases is also achieved. Numerical examples demonstrate the proposed method is effective.

AMS subject classifications: 65M06, 65M50, 76J20

Key words: Ghost cell method, Cartesian grid, adaptive tree method, inviscid compressible flow.

1 Introduction

This paper focuses on the Ghost Cell method (GCM) and its applications for inviscid compressible flow on adaptive tree Cartesian grids. As we all know, a continuous obstacle of computational fluid dynamics (CFD) for configurations with complex geometry is the problem of mesh generation. Although a variety of grid generation techniques are now available, the generation of a suitable grid for a complicated, multi-element geometry is still a tedious, difficult and time-consuming task.

At present, the spatial discretization methods mainly have three approaches [1,2] for dealing with complicated geometry: unstructured grids, body-fitted curvilinear grids, and Cartesian grids. Unstructured grids mainly use triangles in two dimensional flow, tetrahedrons or prisms in three dimension. The advantages lie in the facility of mesh generation for complicated geometry. But the generation is not toilless,

[*] Corresponding author.
URL: `http://www.global-sci.org/aamm/people/NingZhao.html`
Email: jmliu@xznu.edu.cn (J. Liu), zhaoam@nuaa.edu.cn (N. Zhao), huou1215@163.com (O. Hu)

and still hard to get a good quality grid. Also the memory requirements and computational time are in general high. The main advantage of structured grids follows from the property that the indices i, j, k represent a linear address space (computational space), since it directly corresponds to how the flow variables are stored in the computer memory. Furthermore, more importantly in CFD applications, it gives more accurate results due to the discretisation methods used in most flow solvers. But there are also disadvantages. These are the generation of single structured grids for complex geometries, also time-consuming, and it can produce highly skewed grids too. In order to deal with complicated configurations, multiblock structured grids must be used. However, very long times are still required for the grid generation in the case of complex configurations.

A third alternative is the Cartesian grid approach. Conceptually, this approach is quite simple. Solid bodies are cut out of a single static background mesh and their boundaries represented by different types of cut cell, or solid bodies are equipped with ghost cells using the immersed boundary. Most previous work on Cartesian grids for the compressible Euler equations are based on Cartesian finite volume method [3]. But these methodologies may suffer stability problems when an explicit time step is used, cut cells become very small, and degenerate cells will be encountered. Generally, in two dimensions, a degenerate cell is defined as a cut-cell where the irregularly shaped (embedded) boundary (i) intersects the cell at more than two points or (ii) interacts any cell face at more than 1 point [4]. Some technique must be employed to overcome those problems and time step stability restrictions [4, 5]. Jia et al. [4] present a robust and efficient hybrid cut-cell/ghost-cell method to overcome the degenerate cell, and the heat equations are considered. Several authors [2, 6] use a merging technique, where small irregular cut cell is merged together with a neighboring regular grid cell. By using this merging technique, the conservation is automatically maintained. But this method increases the amount of geometry processing. Other methods include Berger et al. [7, 8] use rotated boxes (h-box method) to enhance stability and, Colella and coworkers [9, 10] use flux-redistribution procedures. Furthermore, embedded or immersed boundary ghost cell methods may be also a good choice, and Cartesian grid finite difference schemes for CFD problems have proven to be quite efficient.

Recently, Sjögreen and Petersson [3] develop an embedded boundary finite difference technique for solving the compressible two- or three-dimensional Euler equations in complex geometries on a Cartesian grid, and slope limiters are used on the embedded boundary to avoid non-physical oscillations near shock waves. Dadone and Grossman [11, 12] provide a novel finite difference ghost cell method on a Cartesian grid, which considers the effect of curvature, and enforces symmetry conditions for entropy and total enthalpy along a normal to the body surface. The results on Cartesian grids indicate that the ghost cell method of [11,12] is remarkably convergent in grid and presents dramatic advantages with respect to the widely used first- and second-order pressure extrapolation techniques on body-fitted polar grids. In above mentioned papers of embedded or immersed boundary ghost cell methods, uniform grid or any grid clustering near the body are used, which must be maintained to the far-field boundary. In [13], Dadone and Grossman give a far-field coarsening and mesh adaptation method for Cartesian grids. Cartesian grids in conjunction with tree data structure are a natural choice for solution-adaptive grids. In this paper, the ghost cell methods with the adaptive tree Cartesian grids, we make a further study for the ghost cell immersed boundary method in inviscid compressible flows, and give some applications of the proposed method. Moreover, the conservation of the method is studied. The extension to three-dimensional flow is also presented.

The remainder of the paper is arranged as follows. In section 2, the high order numerical scheme for Euler equation is described. The boundary treatment is shown in section 3. In section 4, we give the tree data structure and the treatment based grid adaptation. The numerical results obtained using the ghost cell method on the adaptive Cartesian grid are presented in section 5. Concluding remarks are made in the final section.

2 Governing equations and numerical methods

2.1 Governing equations

The inviscid compressible Euler equations can be given in vector form explicitly expressing the conservation laws of mass, momentum and energy. The equations in a Cartesian coordinate system can be written as

$$\frac{\partial U}{\partial t}+\frac{\partial F(U)}{\partial x}+\frac{\partial G(U)}{\partial y}+\frac{\partial H(U)}{\partial z}=0, \tag{2.1}$$

where

$$U=\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ \rho E\end{bmatrix},\quad F=\begin{bmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ \rho uw\\ u(\rho E+p)\end{bmatrix},\quad G=\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ \rho vw\\ v(\rho E+p)\end{bmatrix},\quad H=\begin{bmatrix}\rho w\\ \rho uw\\ \rho vw\\ \rho w^2+p\\ w(\rho E+p)\end{bmatrix}.$$

The variables p,ρ,u,v,w are the pressure, the density, and the three Cartesian components of the velocity vector, respectively, and E represents the total energy per unit mass. The pressure p is obtained using an equation of state for ideal gases

$$p=(\gamma-1)\Big(\rho E-\frac{1}{2}(u^2+v^2+w^2)\Big). \tag{2.2}$$

2.2 Numerical methods

In order to solve the multi-dimensional Euler equations, dimensional splitting is applied. We use one-dimensional Godunov's method in each coordinate direction, respectively. A MUSCL-type extrapolation using a *minmod* slope limiter, with a formal second order accuracy in space, has been applied to extrapolate the conserved variables onto the left- and right-hand sides of each cell face. An approximate Riemann solver is used to get face flux. For every one-dimensional problem, time discretization use the optimal second TVD Runge-Kutta method [15].

A particularly simple and robust approximate Riemann solver, called HLL, was proposed by Harten, Lax and van Leer in [14]. But it has the serious flaw of diffusing contact surfaces. This is mainly because the HLL solver reduces the exact Riemann problem to two pressure waves and therefore neglects the contact surface. An improved version of the HLL Riemann solver, named HLLC, is proposed by Toro [16], which is a modified three waves solver. This HLLC scheme is found to have the following properties [17]: (1) exact preservation of isolated contact and shear waves, (2) positivity preserving of scalar quantity, and (3) enforcement of entropy condition. The HLLC solver is versatile, and has been successfully used in various inviscid or viscous compressible flow on multifarious grids. Due to the significant advantages, in this work, HLLC solver is adopted as the approximate Riemann solver to discrete the convection flux on adaptive tree Cartesian grids.

The HLLC flux is defined by [16, 18]

$$F_{HLLC} = \begin{cases} F_l, & \text{if} \quad S_L > 0, \\ F(U_l^*), & \text{if} \quad S_L \le 0 < S_M, \\ F(U_r^*), & \text{if} \quad S_M \le 0 \le S_R, \\ F_r, & \text{if} \quad S_R < 0, \end{cases} \tag{2.3}$$

where

$$U_l^* = \begin{bmatrix} \rho_l^* \\ (\rho u)_l^* \\ (\rho v)_l^* \\ (\rho w)_l^* \\ (\rho E)_l^* \end{bmatrix} = \Omega_l \begin{bmatrix} \rho_l(S_L - q_l) \\ (S_L - q_l)(\rho u)_l + (p^* - p_l)n_x \\ (S_L - q_l)(\rho v)_l + (p^* - p_l)n_y \\ (S_L - q_l)(\rho w)_l + (p^* - p_l)n_z \\ (S_L - q_l)(\rho E)_l - p_l q_l + p^* S_M \end{bmatrix},$$

$$U_r^* = \begin{bmatrix} \rho_r^* \\ (\rho u)_r^* \\ (\rho v)_r^* \\ (\rho w)_r^* \\ (\rho E)_r^* \end{bmatrix} = \Omega_r \begin{bmatrix} \rho_r(S_R - q_r) \\ (S_R - q_r)(\rho u)_r + (p^* - p_r)n_x \\ (S_R - q_r)(\rho v)_r + (p^* - p_r)n_y \\ (S_R - q_r)(\rho w)_r + (p^* - p_r)n_z \\ (S_R - q_r)(\rho E)_r - p_r q_r + p^* S_M \end{bmatrix},$$

$$F_l^* \equiv F(U_l^*) = \begin{bmatrix} \rho_l^* S_M \\ (\rho u)_l^* S_M + p^* n_x \\ (\rho v)_l^* S_M + p^* n_y \\ (\rho w)_l^* S_M + p^* n_z \\ ((\rho E)_l^* + p^*) S_M \end{bmatrix}, \quad F_r^* \equiv F(U_r^*) = \begin{bmatrix} \rho_r^* S_M \\ (\rho u)_r^* S_M + p^* n_x \\ (\rho v)_r^* S_M + p^* n_y \\ (\rho w)_r^* S_M + p^* n_z \\ ((\rho E)_r^* + p^*) S_M \end{bmatrix},$$

$$\Omega_l \equiv (S_L - S_M)^{-1}, \qquad \Omega_r \equiv (S_R - S_M)^{-1},$$

$$p^* = \rho_l(q_l - S_L)(q_l - S_M) + p_l = \rho_r(q_r - S_R)(q_r - S_M) + p_r,$$

and

$$q \equiv un_x + vn_y + wn_z,$$

with $\vec{n}=[n_x, n_y, n_z]^T$ being the unit normal vector to the interface. Intermediate wave velocity S_M is taken from Batten et al. [19]

$$S_M = \frac{\rho_r q_r (S_R - q_r) - \rho_l q_l (S_L - q_l) + p_l - p_r}{\rho_r (S_R - q_r) - \rho_l (S_L - q_l)}.$$

Signal velocities S_L and S_R are defined as

$$\begin{aligned} S_L &= \min\left(\lambda_1(U_l), \lambda_1(U^{Roe})\right), \\ S_R &= \max\left(\lambda_m(U_r), \lambda_m(U^{Roe})\right), \end{aligned}$$

with $\lambda_1(U^{Roe})$ and $\lambda_m(U^{Roe})$ being the smallest and largest eigenvalues of the Roe matrix.

In the approximate Riemann solvers, a higher-order approximation must be interpreted in terms of flux values to achieve second order accuracy at control-volume boundaries. This paper use van Leer's monotone upstream-centred scheme for conservation laws (MUSCL) approach to get second-order accuracy, and *minmod* limiter to damp spurious oscillation, which are shown as follows [20]

$$\begin{aligned} u^R_{j+\frac{1}{2}} &= u_{j+1} - \frac{1}{4}\left[(1-k)\widetilde{\Delta}_{j+\frac{3}{2}} u + (1+k)\widetilde{\widetilde{\Delta}}_{j+\frac{1}{2}} u\right], \\ u^L_{j+\frac{1}{2}} &= u_j + \frac{1}{4}\left[(1-k)\widetilde{\widetilde{\Delta}}_{j-\frac{1}{2}} u + (1+k)\widetilde{\Delta}_{j+\frac{1}{2}} u\right], \end{aligned} \tag{2.4}$$

where

$$\begin{aligned} &\widetilde{\Delta}_{j+\frac{1}{2}} u = \text{minmod}(\Delta_{j+\frac{1}{2}} u, \omega \Delta_{j-\frac{1}{2}} u), \\ &\widetilde{\widetilde{\Delta}}_{j+\frac{1}{2}} u = \text{minmod}(\Delta_{j+\frac{1}{2}} u, \omega \Delta_{j+\frac{3}{2}} u), \\ &\text{minmod}(x, \omega y) = \text{sgn}(x) \max\{0, \min[|x|, \omega y \text{sgn}(x)]\}, \end{aligned}$$

and k is a coefficient of MUSCL scheme. When set k=1/3 , we can get a third order upwind scheme for uniform grid. ω is a constant specified by user, generally, ω=1.

3 Ghost cell methods on Cartesian grids

Recently, Dadone and Grossman [11, 12] present some systemic results about a novel ghost cell method for static body on Cartesian grids. In their papers, the ghost cell values are developed from an assumed flow field model in vicinity of the wall consisting of a vortex flow with locally symmetric distribution of entropy S and total enthalpy H per unit mass along a surface normal, and take into account the effect of curvature. If we make R the signed local radius of curvature of the wall, V_t the velocity component tangential to the body surface, then this flow model satisfies the normal momentum equation

$$\frac{\partial p}{\partial \vec{n}} = -\rho \frac{(V_t)^2}{R}. \tag{3.1}$$

The above equation is equipped with the nonpenetration boundary condition V_n=$\mathbf{V}\cdot\vec{n}$=0, and the antisymmetric normal derivative $\partial S/\partial \vec{n}$ and $\partial H/\partial \vec{n}$ along the surface normal to the body in the vicinity of the body. These entropy and total enthalpy distributions will produce zero normal derivatives when the flow is irrotational. The method has shown to produce superior accuracy compared to the classical surface boundary condition in [11,12]. In view of many good characters, the method has been successfully used for unstructured grid by Wang and Sun in [21].

3.1 Two-dimensional ghost cell method on Cartesian grids

In this study, the body is immersed into the Cartesian volume grid. In order to implement the ghost cell methodology for an immersed boundary, we need to first identify cells whose centers are inside the solid, and then determine the ghost cells. Solid cells can be identified by ray-tracing. Here, two closest rows of solid cells near the body surface are identified as the ghost cells (GC). Those cells are shown in Fig. 1. Following this, we need to devise a scheme that will allow us to compute the value of the variables at each of these ghost cell centers such that the boundary condition on the immersed boundary in the vicinity of the ghost-cell is satisfied. Here, the ghost cell method for two dimensional inviscid flow on Cartesian grids use the following equations described by [11]

$$p_{GC} = p_{IP} - \rho_{IP}\frac{V_{tIP}^2}{R}\Delta n, \tag{3.2}$$

$$\rho_{GC} = \rho_{IP}\left(\frac{p_{GC}}{p_{IP}}\right)^{\frac{1}{\gamma}}, \tag{3.3}$$

$$V_{tGC}^2 = V_{tIP}^2 + \frac{2\gamma}{\gamma-1}\left(\frac{p_{IP}}{\rho_{IP}} - \frac{p_{GC}}{\rho_{GC}}\right), \tag{3.4}$$

$$V_{nGC} = -V_{nIP}, \tag{3.5}$$

where IP denotes the image point concerning the relevant ghost cell (GC), and Δn indicates the distance between the IP and GC. Then, the nonpenetration boundary condition is satisfied automatically. Using the surrounding fluid cells, the values at the image points IP can be achieved by a bilinear interpolation for two dimensional flow, a trilinear interpolation for three dimensional flow. In the above procedure, a situation may be encountered where one of the fluid cells surrounding the image point is the ghost cell itself [22,23], see for instance the point IP1 shown in Fig. 1. In this case, we only use a linear interpolation. It is also the case that the surrounding interpolation stencil for image point of ghost cell contains other ghost cells. This situation only takes

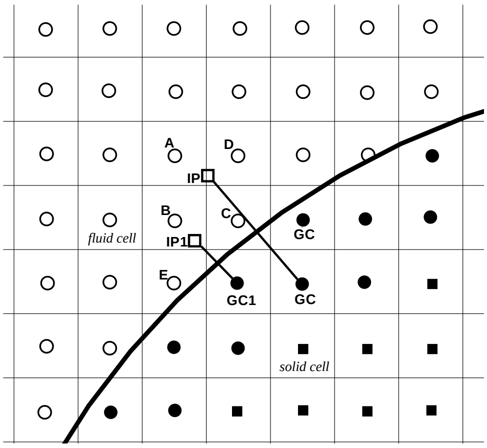

Figure 1: 2D schematic describing ghost cell method used in the current solver.

into account the fluid cells to achieve the value. In this special case, the interpolation formula at the image point uses the following equation

$$\phi_{IP} = \left(\sum_{i=1}^{m} \frac{1}{r_i}\phi_i\right) \Bigg/ \left(\sum_{i=1}^{m} \frac{1}{r_i}\right), \tag{3.6}$$

where r_i indicates the distance between the image point IP and the fluid cell surrounding IP, m is the total number of the fluid cells.

In order to evaluate the pressure p_{GC} at ghost cell in Eq. (3.2), for an arbitrary curved boundary, we also need to estimate the local curvature. Here we adopt the method suggested by Wang and Sun [21] for two dimensional flow. Consider a curved boundary shown in Fig. 2. To estimate the curvature for body surface A-B, firstly we use A-B and the point to the left of the boundary face (point Lft) to make one estimate,

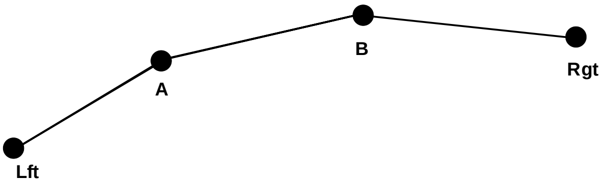

Figure 2: Estimation of local curvature for 2d body.

and then we use A-B and the right point Rgt to perform the other estimate, which is given in [21]. Then, a simply average of the two estimates may obtain the final radius. The local averaged radius is used as the final approximation of the local curvature. However, if either point A or B is a sharp corner, then we only use the estimate from the three points, avoiding the sharp corner.

3.2 Three-dimensional ghost cell method on Cartesian grids

For three-dimensional flow, the extension is as follows [12]

$$p_{GC} = p_{IP} - \rho_{IP}\frac{\widetilde{V}_{IP}^2}{R}\Delta n, \tag{3.7}$$

$$\rho_{GC} = \rho_{IP}\left(\frac{p_{GC}}{p_{IP}}\right)^{\frac{1}{\gamma}}, \tag{3.8}$$

$$V_{nGC} = V_{GC}\cdot\vec{n} = -V_{IP}\cdot\vec{n}, \tag{3.9}$$

$$\widetilde{V}_{GC}^2 = \widetilde{V}_{IP}^2 + \frac{2\gamma}{\gamma-1}\left(\frac{p_{IP}}{\rho_{IP}} - \frac{p_{GC}}{\rho_{GC}}\right), \tag{3.10}$$

$$\widehat{V}_{GC} = 0, \tag{3.11}$$

where the vector $\vec{n}$ is normal to the body surface, $\widetilde{V}=V-(V\cdot\vec{n})\vec{n}$ is in the direction of the streamline projected onto the surface, and $\widehat{V}$ the velocity component normal to $\vec{n}$ and $\widetilde{V}$.

The method of [12] for curvature in Eq. (3.7) is very complicated, which need to find the intersection of the body surface with the plane formed by the tangent to the surface streamline and the surface normal. If the geometry is a sphere, the radius of the local curvature is simply the radius of the sphere. For an arbitrary 3D body, some methods must be developed to estimate the local curvature. In practical computation, for simplicity, we can use local averaged radius as the approximation of the curvature near the body surface. Given arbitrary four noncoplanar points A, B, C and D, we can find the radius by solving the following equations

$$(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2 = R^2, \quad i = A, B, C, D.$$

If we set the volume of the tetrahedron ABCD is Ω,

$$|AB| = a, \quad |AC| = b, \quad |AD| = c, \quad |BC| = r, \quad |BD| = q, \quad \text{and} \quad |CD| = p,$$

then

$$R = \frac{1}{24\Omega}\sqrt{(ap+bq+cr)(-ap+bq+cr)(ap-bq+cr)(ap+bq-cr)}, \tag{3.12}$$

where Ω can be computed from the following equation

$$\Omega = \frac{1}{3}\sum_{m=1}^{4}(\vec{r}_{mid}\cdot\vec{S})_m.$$

Here $(\vec{r}_{mid})_m$ denotes the midpoint of the tetrahedron face m, and $\vec{S}_m$ the face vector (outward directed) at face m, respectively. The local averaged radius is used as the final approximation of the local curvature.

If reflection boundary conditions are used on a solid boundary in the above method and the effective of curvature is considered to be neglected, the variables on ghost cell can be obtained by

$$p_{GC} = p_{IP}\,, \tag{3.13}$$

$$\rho_{GC} = \rho_{IP}\,, \tag{3.14}$$

$$V_{GC} = V_{IP} - 2(V_{IP}\cdot\vec{n})\vec{n}\,. \tag{3.15}$$

4 Solution-based mesh refinement on Cartesian grids

Cartesian grids in conjunction with tree data structure is a natural choice for solution-adaptive grids. In this work, we use a generalized binary tree (e.g., quad-tree in 2D, octree in 3D, etc.) data structure [24].

A solution-based approach to mesh refinement or coarsening is applied, i.e., sensors are employed to detect and localize physical flow phenomena. We adopt the sensors concerning curl and divergence of velocity suggested by De Zeeuw [24] as follows

$$\tau_{ci} = |\nabla\times V|d_i^{\frac{3}{2}}, \qquad \tau_{di} = |\nabla\cdot V|d_i^{\frac{3}{2}},$$

where i=$1,2,\cdots,N$, with N being the total number of cells and d=$\sqrt{\Omega}$ (Ω is the cell volume). The standard deviation of both parameters are computed as

$$\sigma_c = \sqrt{\frac{\sum\limits_{i=1}^{N}\tau_{ci}^2}{N}}, \qquad \sigma_d = \sqrt{\frac{\sum\limits_{i=1}^{N}\tau_{di}^2}{N}}.$$

A cell is flagged for refinement or coarsening if one of two possible conditions hold: (1) if either $\tau_{ci}>\sigma_c$ or $\tau_{di}>\sigma_d$, the cell is flagged for refinement; (2) if both $\tau_{ci}<1/10\sigma_c$ and $\tau_{di}<1/10\sigma_d$, the cell is flagged for coarsening.

When a cell is flagged for refinement, we must present the value at the center of a child cell. In this study, we use a linear interpolation polynomial to get the child cell's value, with *minmod* limiter damping the oscillation near shock wave. A cell is flagged for coarsening, the parent cell only use a conservative average to obtain the value. When an adaptive refinement is used, we may encounter nonuniform grids shown in Fig. 3. If we want to advance the value at the point A, we should find the value at the grid point E firstly. For this case, we find his surrounding cells A, B, C, D, then the Eq. (3.6) is used to interpolate the vale at the point E. At the interface between coarse and fine grids , we must impose correction of conservation law to make the fluxes into the fine grid across a coarse/fine cell boundary equal the flux out of the coarse cell [25]. Here, we make use of the effective method given in [25] to ensure global conservation.

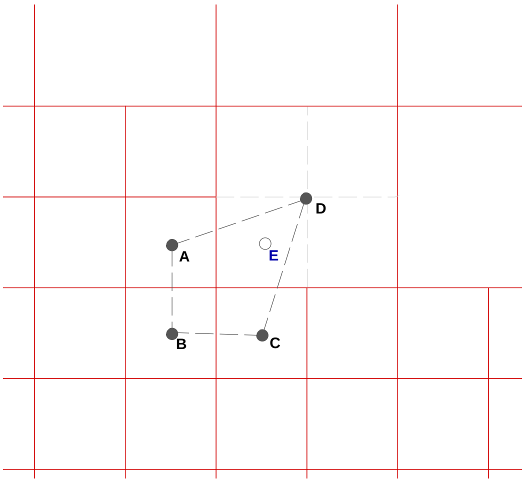

Figure 3: The treatment of coarse-fine grid interface.

5 Results for ghost cell methods on Cartesian grids

The main purpose of the test cases is to demonstrate the capability of the developed overall algorithm. The proposed method has been applied to compute two- and three-dimensional compressible flows for various configurations.

5.1 Supersonic flow past a circular cylinder

Firstly, we compute supersonic two-dimensional compressible flow around a circular cylinder of radius r=1 at Ma_1=3. In this simulation, initial grids (initial largest spatial step h=r/4) have been refined three times near the body firstly, then three levels of solution-based refinement have been done. The final mesh is shown on the left of Fig. 4. The total number of the computational grid cell is 37977. In this computation, the number of the grid is far less than the one used in [3] for combining the ghost cell method Eqs. (3.2)-(3.5) with the adaptive Cartesian grid method. Numerical density contours obtained for Ma_1=3 is presented on the right of Fig. 4. This figure shows a sharp resolution near the shock. From the theoretical and experimental results, the distance Δ of the shock from the obstacle, measured on the stagnation is approximated in [26, 27] by

$$\frac{\Delta}{r} = 0.386 \exp\left(\frac{4.67}{M_1^2}\right). \tag{5.1}$$

Eq. (5.1) gives the theoretical value Δ=0.6485. Using our ghost cell method on the adaptive Cartesian grids, the numerical value of the mean shock detachment is given by Δ=0.685.

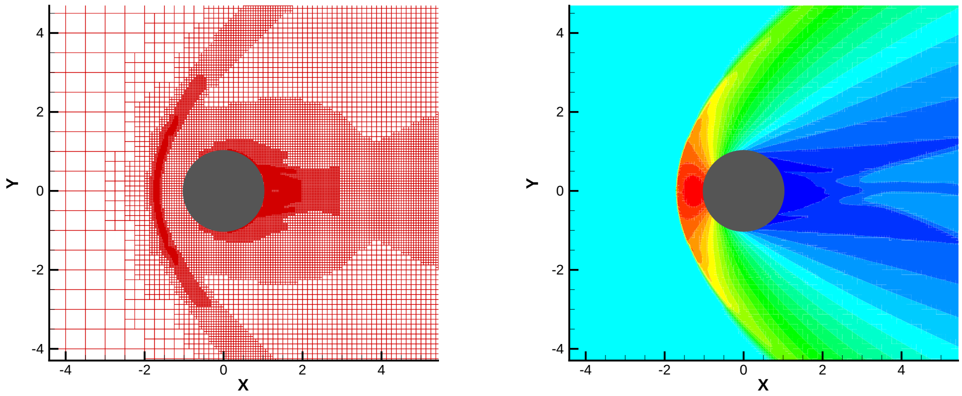

Figure 4: Example 5.1. Left: zoom of the adaptive grid. Right: density contours.

5.2 Steady-state supersonic flow around a triangle

In this subsection, we consider the same case with the one in [27], where a supersonic flow past a solid body of triangular shape with height h=0.5 and half angle θ=20°, as shown on the left of Fig. 5. The free stream Mach number is 2. In this geometry, a special procedure will be encountered, which is the problem of multi-valued ghost points [3, 11–13]. It is often found near unresolved thin surfaces and sharp corners, such as the cases at the sharp trailing edge of an airfoil or near the apex of triangle as Fig. 5. At sharp corners, one cell center inside the geometry may be the ghost cell center for one side of the corner surface, as well as for the other side. Also, a ghost cell center pertaining to one side of a corner surface may be located inside the flow field

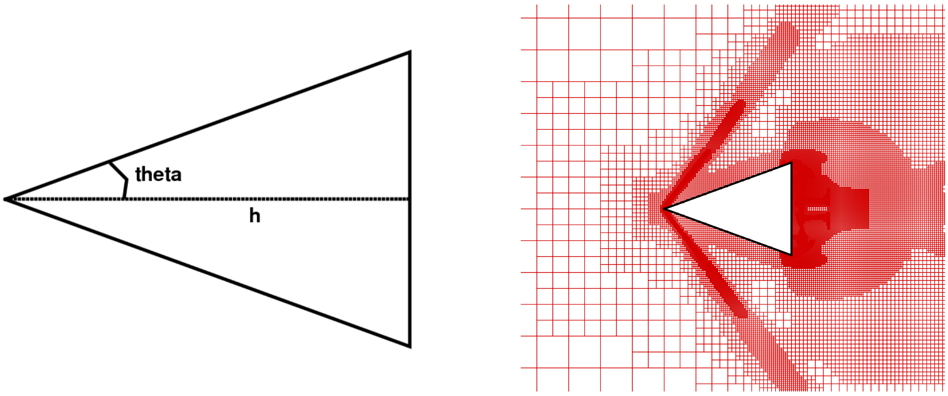

Figure 5: Example 5.2. Left: geometrical configuration for oblique shock wave analysis. Right: zoom of adaptive Cartesian grids.

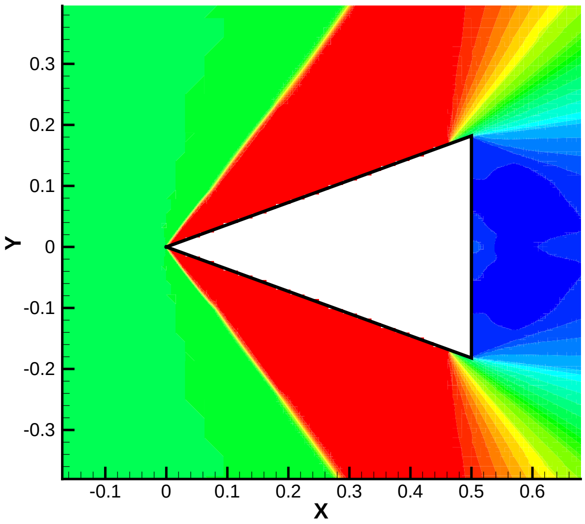

Figure 6: Example 5.2. Density contours for $Ma_1 = 2$.

on the other side of the corner. To handle this case, considering the used numerical scheme of time evolution, we firstly find the fluid cells which need multiple valued ghost points to evolute in time, and set flags. Generally, we give ghost boundary conditions before time evolution. In this special case, the multiple valued ghost points are evaluated during the time evolution, which accords to the fluid cell to find the corresponding ghost cell value. The rest fluid cells are computed as usual.

We solve the problem using the presented adaptive ghost cell Cartesian grids method on the domain $[-4,4] \times [-4,4]$. Initial grids (initial largest spatial step h=0.5) have been refined six times near the body firstly, then make three times solution adaption. The final mesh is shown on the right of Fig. 5. The total number of grid points is 40233. Fig. 6 shows contours of the density obtained with the presented method. The figure clearly indicates the position of the attached shock.

An oblique shock can be attached or detached depending on the values of the deflection angle θ and the upstream Mach number Ma_1. If the shock is attached to the triangle, its angle with the horizontal, β, can be computed through following relation [27,28]

$$\tan\theta = 2\cot\beta\Big(\frac{M_1^2\sin^2\beta - 1}{M_1^2(\gamma+\cos 2\beta)+2}\Big). \tag{5.2}$$

For M_1=2 and θ=20°, Boiron et al. [27], according to Eq. (5.2), compute the angle β=53.46°. In [27], the authors also give some numerical values of β by using a high-resolution penalization method to solve viscous Navier-Stokes equations. Their numerical value of β is 54.13° on 512^2 grid points, 53.70° on 1024^2 grid points and 53.56° on 512^2 grid points. In our computation for Euler equations on 40233 grid points, the mean shock angle β=54.20°. The numerical result is comparable with theirs.

5.3 Conservation properties

In [3], Sjögreen and Petersson analyzed the conservation properties of their Cartesian embedded boundary method. To investigate the quantity of possible loss of mass from the embedded boundary, they compute a steady subsonic flow in a channel with an elliptic obstacle. We will also use the same case to study the mass loss by the present method. In the investigation, the domain is of size $[-3,3] \times [-2,2]$ and the ellipse has the axis length 0.5 and 0.3 in the x- and y-directions respectively. The free stream Mach number is 0.5. The upper and lower boundaries are solid walls where slip boundary conditions are imposed. The conditions are identical with the ones of [3].

The conservation of mass is measured by comparing the total mass flux across the left inlet with the total mass flux across the right outlet boundary. According to [3], the mass flux over the grid line x_i is approximated by the sum

$$F_i = \sum_{j=1}^{N-1} h\rho_{i,j+1/2}u_{i,j+1/2},$$

where N is the number of grid points in the j direction, and $\rho_{i,j+1/2} = (\rho_{i,j+1} + \rho_{i,j})/2$. The discrepancy between the influx F_1 and the outflux F_M shows the mass loss, which is ΔF=$F_M - F_1$. For the present ghost cell method, 304 × 204 grid points are used. In this situation, the mass loss ΔF is 3.32×10^{-3}, and the relative loss is 1.65×10^{-3}. In [3], using the limiter at embedded boundary, the authors showed that the mass loss ΔF is 1.5×10^{-2}, and the relative loss is 5.4×10^{-3} with 301 × 301 grid points. Without the limiter, the authors of [3] showed that the mass loss ΔF is 1.2×10^{-2}, and the relative loss is 4.3×10^{-3}. It is interesting that we can also get smaller mass loss (the mass loss ΔF=8.541×10^{-3}, and the relative loss is 5.36×10^{-3}) if we make use of the reflection boundary conditions Eqs. (3.13)-(3.15), which shows the results are slightly put into the shade than the boundary conditions Eqs. (3.2)-(3.5) concerning the effect of curvature, locally symmetric distribution of entropy and total enthalpy per unit mass along a surface normal. This demonstrates some advantage of the present method, see also [11]. Fig. 7 presents the numerical contour image for the pressure.

5.4 Flow past a NACA0012 airfoil

Now, let us consider an application of the proposed method to deal with airfoil. An airfoil has a sharp trailing edge, which is difficult to define the ghost cell boundary as the aforesaid triangle. In this case, we also used our adaptive ghost cell method to handle this question and get some desired results.

In the transonic flow computation, the Mach number is 0.8 and the angle of attack 1.25° around the NACA0012 airfoil. This case is run for outer boundary of the mesh at an 10-chord radius. The initial mesh was firstly refined six times near the body

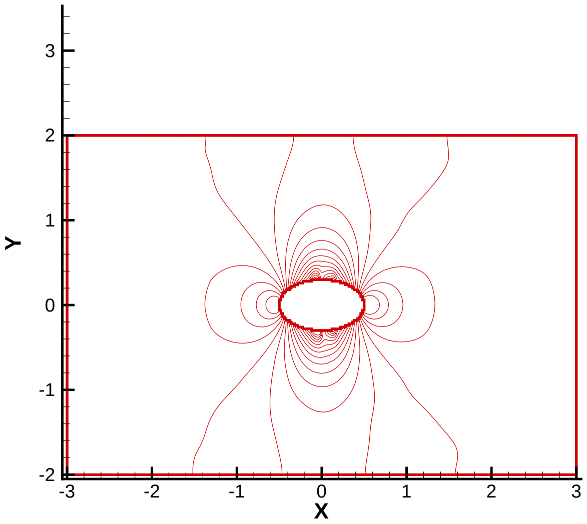

Figure 7: Example 5.3. Pressure contours for Mach 0.5 flow in a channel with obstacle.

boundary (the largest spacial interval h=0.5, airfoil chord=1). Three levels of solution-based refinement are done during the time evolution. The mesh is shown in Fig. 8. In the process of solution-based refinement, we only use the sensor of the divergence of velocity σ_d. The pressure contours are shown on the left of Fig. 9, and the pressure coefficient is plotted on the right. As observed in Fig. 9, both the strong shock and the weak shock are well resolved, and the positions of shock are also well located. The relations between convergence history and the number of iterations are shown in Fig. 10.

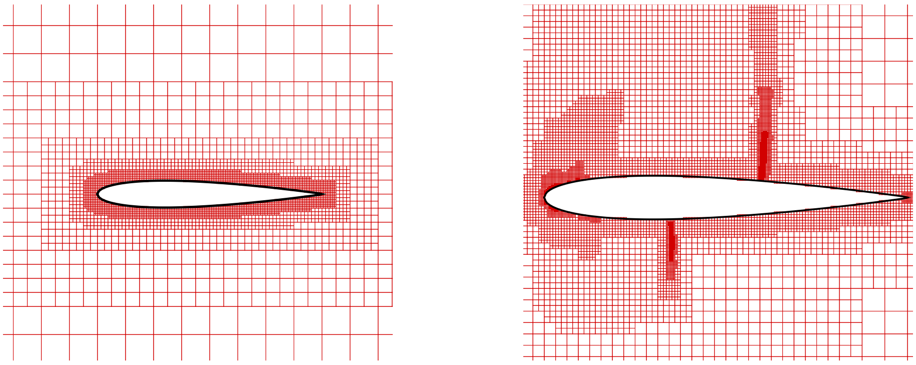

Figure 8: Example 5.4. Left: the initial grid of NACA0012 airfoil. Right: zoom of adaptive Cartesian grids.

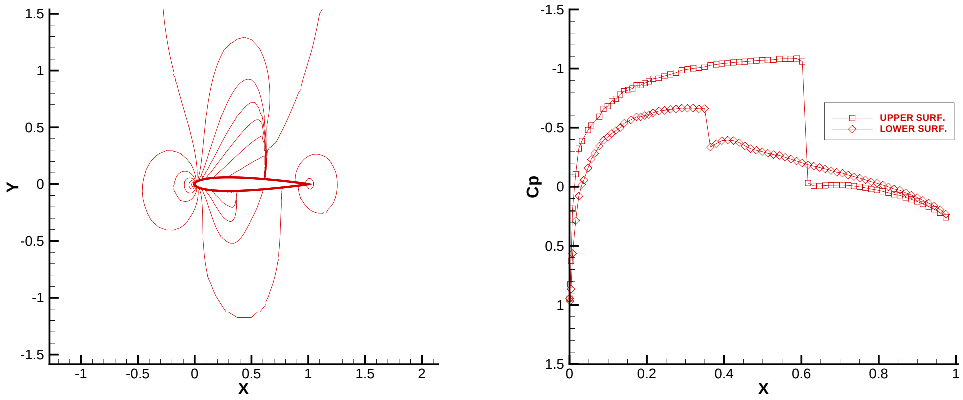

Figure 9: Example 5.4. Left: pressure contours. Right: distribution of pressure around the surface of the airfoil (Mach 0.8 at angle of attack 1.25°).

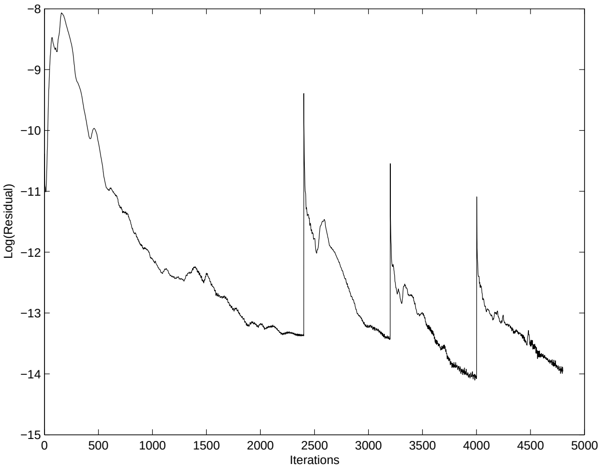

Figure 10: Example 5.4. Convergence histories vs the number of iterations.

5.5 Three-dimensional results

Finally, let us consider a simple three-dimensional supersonic flow past a sphere to show the capability of the present adaptive Cartesian grid ghost cell method. The sphere has radius r=1. We adopt octree data structure for mesh generation. We show the final refined grids on the left of Fig. 11. In this study, the initial mesh is firstly refined two times near the body boundary, and two levels of solution-based refinement are done during the time evolution (the largest spacial interval h=0.5). We also indicate the density distribution for the steady Mach 3 flow past a sphere in Fig. 11. From the figure, although the spacial interval is coarser, the bow shock is well resolved, which indicates that our three-dimensional code is effective. The computation was run time accurately until the bow shock moved into the correct position.

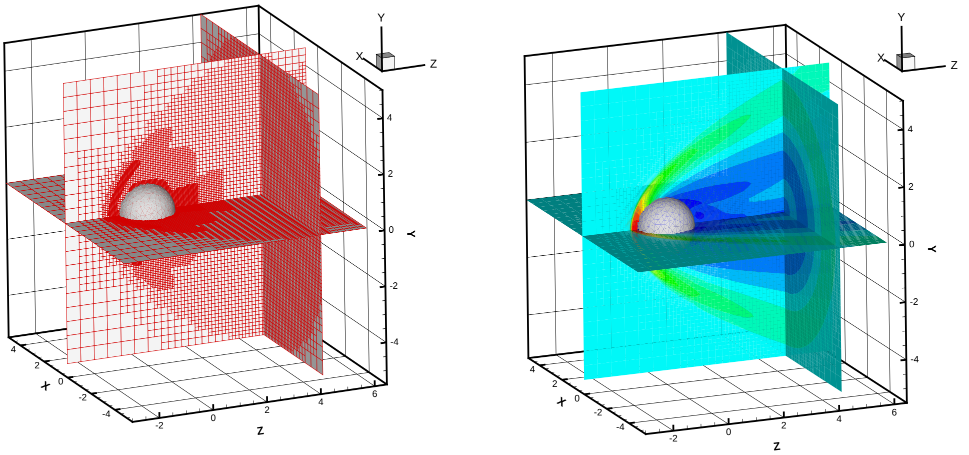

Figure 11: Example 5.5. Left: zoom of Cartesian adaptive grids. Right: density contours (Mach 3 flow past a sphere).

From the theoretical and experimental results, the distance Δ of the shock from the sphere, measured on the stagnation, is approximated in [26] by

$$\frac{\Delta}{r} = 0.143 \exp\left(\frac{3.24}{M_1^2}\right). \tag{5.3}$$

Eq.(5.3) gives the theoretical value of Δ=0.205. Using our ghost cell method on adap-

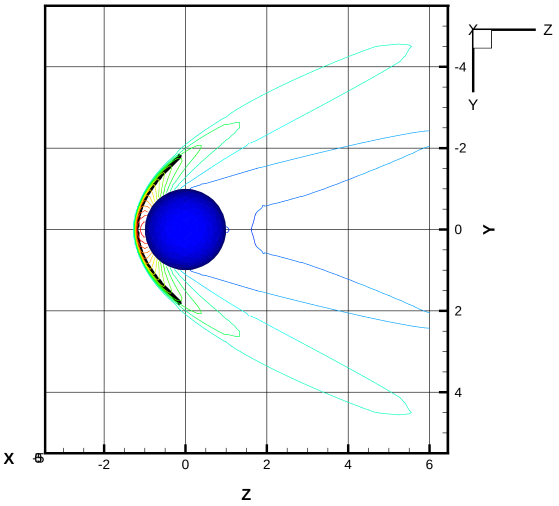

Figure 12: Example 5.5. Comparison of theoretical by Eq. (5.4) (*thick black line*) and computed shock wave shape by CFD for flow over sphere.

tive Cartesian grids, the numerical value of mean shock detachment Δ is 0.213 approximately.

In order to evaluate the effectiveness of our method, we also study the shape of shock at x=0. Billig [26], assumed that the detached shock wave is a hyperbola that is asymptotic to the freestream Mach angle or in the case of a cone or wedge afterbody, the attached shock angle θ. The equation presented by Billig for the coordinates of the shock at x=0 is

$$z=-\Big\{r+\Delta-r_c cotan^2\theta\Big[\Big(1+\frac{y^2tan^2\theta}{r_c^2}\Big)^{\frac{1}{2}}-1\Big]\Big\}. \tag{5.4}$$

The approximate relationship for the vertex radius of the curvature r_c of the sphere-cone and the shock shape is presented as follows

$$\frac{r_c}{r}=1.143\exp\Big[\frac{0.54}{(M_1-1)^{1.2}}\Big]. \tag{5.5}$$

The freestream Mach angle may be gotten by

$$\theta=arcsin\frac{1}{M_1}. \tag{5.6}$$

Using Eqs. (5.5) and (5.6), the shape of shock is approximated by Eq. (5.4). The isobars of pressure at x=0, and the approximate shape (*thick black line*) of shock wave achieved by Eq. (5.4), are plotted in Fig. 12. From this figure, the computed shock shape by the present Cartesian grid method is well consistent with the theoretical result.

6 Conclusions

In this paper, we developed a new kind of immersed boundary methods which combine the ghost cell method with adaptive tree Cartesian grid method. The proposed method has been applied successfully to solve steady state problems of the Euler equations for a variety of two- and three- dimensional internal and external flows. Numerical tests included a supersonic flow past a triangle, circular cylinder, three-dimensional sphere, a transonic NACA airfoil, and a subsonic flow in a channel with obstacle. Numerical results showed that the present method can achieve correct shock positions, high resolution, and some correct physical characters. Despite the violation of the conservation in the immersed boundary methods, the loss of mass of the present method is smaller than other works for the same cases. It is demonstrated that the present method is very effective. The extension to handle moving body and viscous compressible Navier-Stokes solver is under development.

Acknowledgments

The authors would like to thank the referees and editors for their helpful suggestions which improve this article. We would also like to thank Dr. Chonghai Wu, College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics, for his help in revising this paper. This research was supported partly by National Science Foundation of China (10728026) and National Basic Research Program of China (2007CB714600).

References

[1] J. Blazek, Computational Fluid Dynamics: Principles and Applications, Elsevier, 2001.

[2] G. Yang, D. M. Causon, D. M. Ingram, R. Saunders and P. Batten, *A Cartesian cut cell method for compressible flows Part A: static body problems*, The Aeronautical Journal, Feb. (1997), pp. 47–56.

[3] B. Sjögreen and N. A. Petersson, *A Cartesian embedded boundary method for hyperbolic conservation laws*, Commun. Comput. Phys. , 2 (2007), pp. 1199–1219.

[4] H. Jia, F.-S. Lien and E. Yee, *A robust and efficient hybrid cut-cell/ghost-cell method with adaptive mesh refinement for moving boundaries on irregular domains*, Comput. Methods Appl. Mech. Engrg., 198 (2008), pp. 432–448.

[5] H. Forrer and R. Jeltsch, *A higher-order boundary treatment for Cartesian-grid methods*, J. Comput. Phys. , 140 (1998), pp. 259–277.

[6] W. J. Coirier and K. G. Powell, *An accuracy assessment of Cartesian-mesh approaches for the Euler equations*, J. Comput. Phys. , 117 (1995), pp. 121–131.

[7] M. J. Berger, C. Helzel and R. J. LeVeque, *H-box methods for the approximation of hyperbolic conservation laws on irregular grids*, SIAM J. Numer. Anal. , 41 (2003), pp. 893–918.

[8] C. Helzel, M. J. Berger and R. J. LeVeque, *A high-resolution rotated grid method for conservation laws with embedded geometries*, SIAM J. Sci. Comput. , 26 (2005), pp. 785–809.

[9] R. B. Pember, J. B. Bell, P. Colella, W. Y. Crutchfield and M. L. Welcome, *An adaptive Cartesian grid method for unsteady compressible flow in irregular regions*, J. Comput. Phys. , 120 (1995), pp. 278–304.

[10] P. Colella, D. T. Graves, B. J. Keen and D. Modian, *A Cartesian grid embedded boundary method for hyperbolic conservation laws*, J. Comput. Phys. , 211 (2006), pp. 347–366.

[11] A. Dadone and B. Grossman, *Ghost-cell method for inviscid two-dimensional flows on Cartesian grids*, AIAA Journal, 42 (2004), pp. 2499–2507.

[12] A. Dadone and B. Grossman, *Ghost-cell method for inviscid three-dimensional flows on Cartesian grids*, Comput. Fluids, 36 (2007), pp. 1513–1528.

[13] A. Dadone and B. Grossman, *Ghost-cell method with far field coarsening and mesh adaptation for Cartesian grids*, Comput. Fluids, 35 (2006), pp. 676–687.

[14] A. Harten, P. D. Lax and B. van Leer, *On upstream differencing and Godunov-type scheme for hyperbolic conservation laws*, SIAM Review, 25 (1983), pp. 35–61.

[15] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes*, J. Comput. Phys. , 77 (1988), pp. 439–471.

[16] E. F. Toro, M. Spruce and W. Speares, *Restoration of the contact surface in the HLL Riemann solver*, Shock Waves, 4 (1994), pp. 25–34.

[17] H. Luo, J. D. Baum and R. Löhner, *Extension of Harten-Lax-van Leer scheme for flows at all speeds*, AIAA Journal, 43 (2005), pp. 1160–1166.

[18] P. Batten, M. A. Leschziner and U. C. Goldberg, *A verage-state Jacobians and implicit methods for compressible viscous and turbulent flows*, J. Comput. Phys. , 137 (1997), pp. 38–78.

[19] P. Batten, N. Clarke, C. Lambert and D. M. Causon, *On the choice of wavespeeds for the HLLC Riemann solver*, SIAM J. Sci. Comput. , 18 (1997), pp. 1553–1570.

[20] Z. Q. Zhu, Z. N. Wu, J. Li, C. Yan and Z. M. Chen, Applied Computaional Fluid Dynamics, Beijing University of Aeronautics & Astronautics Press, 1998 (in Chinese).

[21] Z. J. Wang and Y. Sun, *Curvature-based wall boundary condition for the Euler conditions on unstructured grids*, AIAA Journal, 41 (2003), pp. 27–33.

[22] R. Ghias, R. Mittal and H. Dong, *A sharp interface immersed boundary method for compressible viscous flows*, J. Comput. Phys. , 225 (2007), pp. 528–553.

[23] R. Mittal, H. Dong, M. Bozkurttas, F. M. Najjar, A. Vargas and A. von Loebbecke, *A versatile sharp interface immersed boudanry method for incompressible flows with complex boundaries*, J. Comput. Phys. , 227 (2008), pp. 4825–4852.

[24] D. L. De Zeeuw, *A quadtree-based adaptively-refined Cartesian-grid algorithm for solution of the Euler equations*, PhD Thesis, The University of Michigan, 1993.

[25] M. J. Berger, *Local adaptive mesh refinement for shock hydrodynamics*, J. Comput. Phys ., 82 (1989), pp. 64–84.

[26] F. S. Billig, *Shock-wave shapes around spherical- and cylindrical-nosed bodies*, J. Spacecraft and Rockets, 6 (1967), pp. 822–823.

[27] O. Boiron, G. Chiavassa and R. Donat, *A high-resolution penalization method for large Mach number flows in the presence of obstacles*, Comput. Fluids, 38 (2009), pp. 703–714.

[28] C. Hirsch, Numerical Computation of Internal and External Flows, Wiley, 1988.