Computers & Fluids 53 (2012) 105–116

Contents lists available at SciVerse ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

Two-dimensional unsteady aerodynamics analysis based on a multiphase perspective

Seung-Jun Lee [a,*], Byoung-Gyu Cho [b], In Lee [c]

[a] Thermal-Hydraulics Safety Research Division, Korea Atomic Energy Research Instituite, Daejeon, Republic of Korea
[b] Transport Machinery Examination Division, Machinery, Metal and Construction Examination Bureau, Korean Intellectual Property Office, Daejeon, Republic of Korea
[c] Division of Aerospace Engineering, School of Mechanical, Aerospace & System Engineering, KAIST, Daejeon, Republic of Korea

ARTICLE INFO

Article history:
Received 8 April 2011
Received in revised form 26 July 2011
Accepted 2 October 2011
Available online 10 October 2011

Keywords:
CIP
CCUP
Steady aerodynamics
Unsteady aerodynamics
Multiphase

ABSTRACT

Compressible aerodynamics is analyzed by the CIP/CCUP (constraint interpolation profile/CIP-combined unified procedure) method. The CIP method is feasible for an analysis of various phase change problems, those associated with compressible and incompressible flow areas. For aeroelastic problems based on CFD (computational fluid dynamics), the use of Lagrangian body-fitted grids is problematic because these grids increase the skewness of a mesh, or can be broken – even during structural motions. Therefore, a method embedding a physical boundary in a fixed Eulerian grid is appropriate compared to Lagrangian grids. In this paper, the CCUP method based on a pressure-based algorithm in which the pressure Poisson equation is modified to deal with compressible flows. A collocated grid system with the velocity components on the cell face obtained from the CIP interpolations is applied. Boundary condition for arbitrary surfaces are newly derived using a simple algebraic relationship based on the immersed boundary method. Far-field boundary condition is replaced with sponge layers. Several numerical benchmarking problems, in this case a wedge or a bump under transonic and supersonic flows, are tested to verify the code. Unsteady motions are directly applied to embedded solid areas. Finally, the steady and unsteady aerodynamics of airfoil sections are calculated and compared to reference data to validate the proposed method.

1. Introduction

A considerable amount of aero-vehicle design research has focused on developing high-performance aero-vehicles with greater safety and durability. For instance, recently developed aircrafts with flexible wings or even missiles with rigid wings are commonly used under severe flight conditions as part of mission. The structural flexibility of an aircraft in a complex aerodynamic environment can cause aeroelastic instability, also known as flutter, which seriously threatens the sustainability of the aircraft. Thus, it is becoming increasingly necessary to consider the interaction between phases in an effort to prevent instability.

Flexible wing structures of aircrafts are deflected owing to the aerodynamic forces acting on the wing surfaces. For CFD-based aeroelastic analyses, however, the use of Lagrangian body-fitted grids is often problematic because such grids increase the skewness of meshes or can be broken during structural deflecting motions. Garcia [1] developed a new method which moves the total CV (control volume) during the structural motion of BO105 wing model. However, even in his paper, it was noted that the controlling parameter must be adjusted carefully to obtain stable results for dynamically unstable responses.

Therefore, a method of embedding a physical boundary in a fixed Eulerian grid is appropriate compared to the use of Lagrangian grids. In this paper, the CIP/CCUP method among various interface tracking algorithms, including the MAC (marker and cell) [2], Front tracking [3], level set (LS) [4] and volume of fluid (VOF) [5] algorithms, is used. Conventionally, diffusion in advection equations is a very significant problem in Eulerian grid systems which are employed by many multiphase solvers. A sharp interface usually encounters diffusion when it goes through grids, whereas it is conserved well in CIP/CCUP methods without any numerical treatment [6].

For the CIP method, Yabe et al. [7–9] suggested a scheme that was less diffusive and less oscillating known as CIP, which became one of the Hermitan interpolation methods used to solve hyperbolic-type equations. First, the initial CIP method uses a cubic polynomial function to construct an interpolation function. Contrary to the typical higher order interpolation functions, which utilize many stencils to achieve proper accuracy, the CIP method uses only two neighboring stencils, although it has third-order accuracy. This is possible owing to the utilization of the point values and

[*] Corresponding author.
E-mail addresses: cosinesj@kaeri.re.kr (S.-J. Lee), busycho@kipo.go.kr (B.-G. Cho), inlee@kaist.ac.kr (I. Lee).

0045-7930/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2011.10.001

their first spatial derivatives at the same time. Consequently, the information inside the discrete grid points is recovered well enough to march in time with a low number of dissipation and dispersion errors. Consequently, the CIP method is highly suitable for a discontinuous front and for an interface between different phases.

The CCUP method based on various grid systems was introduced to solve aero- and hydro-dynamic problems [10]. Many researchers have tested the method by solving standard compressible flow problems, such as Sod's shock tube problem [11], and a forward-facing step problem [12]. Moreover, for an incompressible flow, the lid-driven cavity problem and the dam break sloshing problem [13], and others were solved.

In this research, the CCUP method is modified as Ida's suggests [14,15] for the non-advection phase in an effort to improve the convergence of the pressure Poisson equation and to modify the sharpness of the interfaces. Furthermore, as used in the Soroban grids in Ref. [16], a collocated grid system with the velocity components on the cell face obtained from the CIP interpolations is adopted. This system does not require an additional numerical treatment, as in Rhie and Chow [17]. The far-field boundary condition is replaced by the sponge layer, which is one of the absorbing boundary conditions. The solid surfaces are explicitly defined initially, and the instantaneous position of the surface is reconstructed according to the values of the DOFs (degrees of freedom). In addition, because the solid surface does not fit into a Cartesian grid completely, a simple solid surface boundary condition is derived algebraically for steady and unsteady states. The developed solver is verified and validated by solving benchmarking problems, including the problem of square wave propagation with a variable velocity field and transonic/supersonic flows over bumps and a wedge. Finally, flow fields in steady and unsteady motions of airfoils are calculated and compared with experimental and numerical results.

In the remainder of this paper, we begin by investigating the CIP/CCUP scheme and its modifications. Then, boundary conditions for far-fields and solid surfaces are introduced. Finally, two benchmarking problems and steady/unsteady aerodynamics are presented to verify and validate the method.

2. The CIP/CCUP scheme

2.1. CIP method

The CIP method, which is a the Hermitan interpolation method, utilizes point values and their first spatial derivatives at discrete grid points. Some CIP researchers have developed new versions of the CIP method. Among them are the IDO (interpolated differential operator) method [18] and the VSIAM3 (Volume/Surface Integrated Average-based Multi-Moment Method) [19] method. In the IDO scheme, all of the time derivatives are replaced by spatial derivatives. For VSIAM3, instead of the first spatial derivatives, volume- and surface-integrated intermediate parameters are introduced. Unlike conventional numerical schemes in which numerical errors are reduced by constructing higher order interpolation functions with additional stencils, the typical CIP methods adopt only two neighboring stencils, this making these schemes compact. The simple hyperbolic equations below explain this clearly.

$$\frac{\partial \mathbf{f}}{\partial \mathbf{t}} + \mathbf{u}\frac{\partial \mathbf{f}}{\partial \mathbf{x}} = 0, \tag{1}$$

$$\frac{\partial \mathbf{g}}{\partial \mathbf{t}} + \mathbf{u}\frac{\partial \mathbf{g}}{\partial \mathbf{x}} = 0. \tag{2}$$

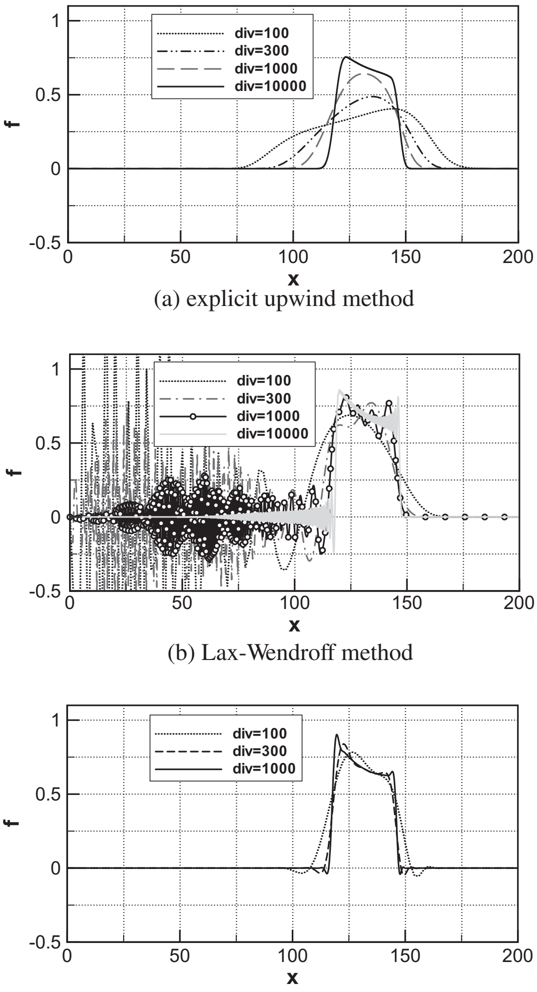
Eqs. (1) and (2) are hyperbolic equations. Here, $\mathbf{u}$ denotes the constant propagation velocity for simplicity. Generally, Eq. (1) was the main subject of the analyses. However, as mentioned previously, the first spatial gradient equation of Eq. (1), that is, Eq. (2), is required. It has the same form as Eq. (1). Thus, **f** and **g** propagate at the same time with the third-order CIP method. As a result, the interpolated profile does not show dissipation or dispersion errors; that is, the time marching solution is numerically stable. Moreover, owing to the low number of required stencils, it is easy to code.

$$\mathbf{f}_i^{n+1} \equiv \mathbf{F}_i^n(\mathbf{x}_i, t+\Delta t) \cong \mathbf{F}_i^n(\mathbf{x}_i - \mathbf{u}\Delta t, t), \tag{3}$$

$$\mathbf{F}_i^n(\mathbf{x}) = \mathbf{a}_i\mathbf{X}^3 + \mathbf{b}_i\mathbf{X}^2 + \mathbf{c}_i\mathbf{X} + \mathbf{d}_i. \tag{4}$$

Here, $\mathbf{f}_i^{n+1}$ denotes the point value of grid i at the $n + 1$ time step. $\mathbf{F}_i^n$ is the interpolation function derived for $\mathbf{x} \in (\mathbf{x}_{i-1}, \mathbf{x}_i)$, $\mathbf{x} > 0$ or $\mathbf{x} \in (\mathbf{x}_i, \mathbf{x}_{i+1})$, and $\mathbf{x} < 0$. X is the local distance of x from the local origin x_i. The coefficients, $\mathbf{a}_i$, $\mathbf{b}_i$, $\mathbf{c}_i$ and $\mathbf{d}_i$, are uniquely defined by four equations, as each stencil has two equations of **f** and **g**. Detailed information pertaining to these coefficients can be found in the literature [12]. As the CIP method is based on the semi-Lagrangian approach, the advected solution, Eq. (3), written as $\mathbf{F}_i^n(\mathbf{x}_i, t+\Delta t)$, is set to be equal to $\mathbf{F}_i^n(\mathbf{x}_i - \mathbf{u}\Delta t, t)$. This can be realized in Eqs. (1) and (2), which do not have diffusion terms.

The nonlinear hyperbolic equation is tested as shown in Fig. 1, where the courant number is 0.1 and the entire computational domain is $0 < \mathbf{x} < 200$. The domain is divided by 100, 300, 1000, and 10,000. The initial profile is square; that is, it characterized by a value of 1 when $40 < \mathbf{x} < 60$, and 0 otherwise. Particularly,



(a) explicit upwind method

(b) Lax-Wendroff method

(c) CIP method

Fig. 1. Square wave propagation tests by several schemes.

the propagation velocity varies in the space. As noted in earlier research [9], the variable velocity field is as follows:

$$\mathbf{u}(\mathbf{x}) = 1.0 + 0.5\sin(2\pi\mathbf{x}/100). \tag{5}$$

The final results are attained when $t = 100$. The explicit upwind, the Lax–Wendroff, and the CIP method are used here and compared to each other. Overall, the accuracy increases when the mesh is fine. As shown in Fig. 1, the explicit upwind and the Lax–Wendroff scheme show dissipation and dispersion error, respectively. Even when they are divided by 10,000, the results are not improved much. In contrast, the CIP method shows good performance, even when it is divided by 300. Moreover, when the division number increases, the accuracy is improved significantly.

2.2. CCUP method

The CCUP method [20] was devised to solve fluid dynamics problems using the CIP method for the advection part. Conventionally, conservative forms of N–S (Navier–Stokes) equations have been used to analyze flow fields while satisfying the conservation of mass. However, several studies have found non-conservative form to be more flexibly applicable to various types of differential equations with source terms [18]. Thus, non-conservative variable forms of the N–S equations and, for simplicity, the FD (finite difference) method are used in this paper. Although it does not conserve the mass exactly, the CCUP method has good conservation of mass [21].

$$\frac{\partial \rho}{\partial t} + \vec{\mathbf{u}} \cdot \nabla \rho = -\rho \nabla \cdot \vec{\mathbf{u}}, \tag{6}$$

$$\frac{\partial \vec{\mathbf{u}}}{\partial t} + \vec{\mathbf{u}} \cdot \nabla \vec{\mathbf{u}} = -\frac{\nabla \mathbf{p}}{\rho} + Q_{\vec{\mathbf{u}}}, \tag{7}$$

$$\frac{\partial \mathbf{p}}{\partial t} + \vec{\mathbf{u}} \cdot \nabla \mathbf{p} = -\rho \mathbf{C}_{\mathbf{s}}^2 \nabla \cdot \vec{\mathbf{u}} + \mathbf{Q}_{\mathbf{p}}. \tag{8}$$

Eqs. (6)–(8) are the N–S equations in a non-conservative form. Each equation presents the continuity, momentum, and pressure equation derived from the definition of the speed of sound. Eqs. (6)–(8) are divided into two parts by using the fractional time stepping method. The advection equations are extracted from the abovementioned equations, the remaining equations related to the speed of sound result in the simple pressure Poisson equation. All procedures follow the order in the literature [20]. The process of deriving the pressure Poisson equation is as follows:

$$\frac{\rho^* - \rho^n}{\Delta t} = -\rho^n \nabla \cdot \vec{\mathbf{u}}^*, \tag{9}$$

$$\frac{\vec{\mathbf{u}}^* - \vec{\mathbf{u}}^n}{\Delta t} = -\frac{1}{\rho^*}\nabla \mathbf{p}^{**}, \tag{10}$$

$$\frac{\mathbf{p}^{**} - \mathbf{p}^n}{\Delta t} = -\gamma \mathbf{p}^n \nabla \cdot \vec{\mathbf{u}}^*. \tag{11}$$

The divergence of Eqs. (10) and (11) is combined by removing $\nabla \cdot \vec{\mathbf{u}}^*$. As a result, the final pressure Poisson equation presented in Eq. (12) is produced. After the non-advection phase, the advection phase is solved by the CIP scheme.

$$\nabla \cdot \left(\frac{1}{\rho^n}\nabla \mathbf{p}^{**}\right) = \frac{\mathbf{p}^{**} - \mathbf{p}^n}{\rho^n \mathbf{C}_s^2 \Delta t^2} + \frac{1}{\Delta t}\nabla \cdot \vec{\mathbf{u}}^n. \tag{12}$$

Here, $\mathbf{p}$, ρ and $\mathbf{C}_s$ refer to the pressure, the density and the speed of sound. The superscript, **, in Eq. (12), indicates an intermediate process, and n is the present state. In this notation, the advection phase precedes the non-advection phase.

Although the CCUP method is a pressure-based algorithm of the type typically used in incompressible flows, the first term on the right-hand side of Eq. (12) makes the method applicable to a compressible flow. According to one study [8], this guarantees the continuity of $\nabla \mathbf{p}/\rho$; that is, although the density varies by several orders, the pressure gradient can be estimated accurately. Hence, this makes the CCUP method a universal solver, even for a compressible flow regime. The pressure Poisson equation is solved by the successive over relaxation (SOR) method in this research.

3. Modifications of present schemes

3.1. A collocated grid system with the velocity components on the cell face

In this paper, a cell-centered grid system is used to solve fluid dynamics problems. A staggered grid system is used typically for its strong coupling between the pressure and velocity. It automatically prevents the occurrence of the checkerboard phenomenon associated with collocated grids and improves the numerical stability significantly. However, the complexity embedded in a staggered grid system obstructs its extension to multi-dimensional problems. Despite the fact that the CCUP method is essentially a pressure-based approach based on a staggered grid system, a collocated grid system is preferred in this research due to the straightforward application of boundary conditions.

Fig. 2 presents the collocated grid system. The black circles represent the pressure, density, and the velocities. To avoid odd–even decoupling, the density, pressure, and the velocities at the cross symbols and the white circles are interpolated. As noted in earlier research [11], or in the Soroban grids [16], a repetitive use of the CIP interpolation is performed to maintain the third-order accuracy. These imaginary properties form an imaginary cell, as it is a staggered grid system. The following equations present the numerical procedure:

$$\mathbf{u}_{\mathbf{i}+1/2} = \mathrm{CIP1D}(\mathbf{u}_{\mathbf{i}}, \mathbf{u}_{\mathbf{i}+1}, \partial_{\mathbf{x}}\mathbf{u}_{\mathbf{i}}, \partial_{\mathbf{x}}\mathbf{u}_{\mathbf{i}+1}),$$
$$\mathbf{v}_{\mathbf{j}+1/2} = \mathrm{CIP1D}(\mathbf{v}_{\mathbf{j}}, \mathbf{v}_{\mathbf{j}+1}, \partial_{\mathbf{y}}\mathbf{v}_{\mathbf{j}}, \partial_{\mathbf{y}}\mathbf{v}_{\mathbf{j}+1}).$$

Here, the subscripts $i + 1/2$ and $j + 1/2$ denote the cross symbols and white circles, respectively. ∂_k refers to the spatial gradient in the direction of k, where k is $\mathbf{x}$ and $\mathbf{y}$ in two dimensions. The function CIP1D is the one-dimensional CIP interpolation [11]. Two CIP interpolations are used to obtain $\mathbf{u}_{\mathbf{i}+1/2}$ and $\mathbf{v}_{\mathbf{j}+1/2}$. Although one-dimensional CIP interpolation provides both the cell-face values and their spatial gradient information, the gradient information is out of focus in this process.

3.2. Modification of the pressure equation

A new modification of the pressure Poisson equation to improve the numerical stability and convergence was proposed in

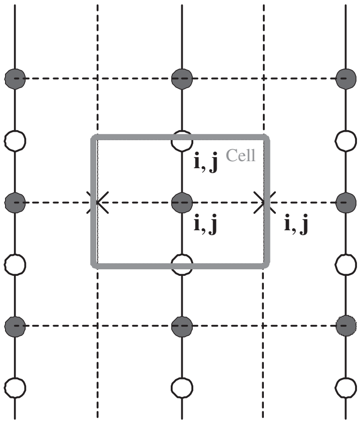


Fig. 2. A collocated grids based on a staggered grids.

an earlier study [15]. The modified pressure Poisson equation is as follows:

$$\frac{\mathbf{p}^{**}-\mathbf{p}^{n}}{\Delta t}=\rho^{n}\mathbf{C}_{\mathbf{s}}^{n^2}\Delta t(\mathbf{G}_1+\mathbf{G}_2)-\rho^{n}\mathbf{C}_{\mathbf{s}}^{n^2}\nabla\cdot\mathbf{u}^{n}, \tag{13}$$

$$\begin{aligned}\mathbf{G}_1&=\alpha(\mathbf{E}_1)\left(\frac{\mathbf{p}_{\mathbf{x}}^{**}}{\rho^{n}}\right)_{\mathbf{x}}+(1-\alpha(\mathbf{E}_1))\left(\frac{\mathbf{p}_{\mathbf{x}}^{n}}{\rho^{*}}\right)_{\mathbf{x}},\\ \mathbf{G}_2&=\alpha(\mathbf{E}_2)\left(\frac{\mathbf{p}_{\mathbf{y}}^{**}}{\rho^{n}}\right)_{\mathbf{y}}+(1-\alpha(\mathbf{E}_2))\left(\frac{\mathbf{p}_{\mathbf{y}}^{n}}{\rho^{n}}\right)_{\mathbf{y}},\end{aligned} \tag{14}$$

$$\begin{aligned}\mathbf{E}_1&=\left(\frac{1}{\rho_{i+1/2,j}^{n}}+\frac{1}{\rho_{i-1/2,j}^{n}}\right)\rho_{i,j}^{n}\left(\frac{\mathbf{C}_{\mathbf{s}i,j}^{n}\Delta t}{\mathbf{h}}\right)^2,\\ \mathbf{E}_2&=\left(\frac{1}{\rho_{i,j+1/2}^{n}}+\frac{1}{\rho_{i,j-1/2}^{n}}\right)\rho_{i,j}^{n}\left(\frac{\mathbf{C}_{\mathbf{s}i,j}^{\mathbf{n}}\Delta t}{\mathbf{h}}\right)^2,\end{aligned} \tag{15}$$

$$\alpha(E)=\frac{1}{1-\mathbf{e}^{-\mathbf{E}}}-\frac{1}{\mathbf{E}}. \tag{16}$$

In this equation, α, the weighting function, is defined in Eq. (16) and $\rho_{i\pm1/2,j\pm1/2}^{n}$ denotes the densities at the cell faces. In addition, $\mathbf{h}$ in Eq. (15) denotes the grid spacing. For a two-dimensional extension, the y-dimensional terms are simply added to Eq. (14). In this research, the face values are interpolated by the CIP method, as shown in Section 3.1. Furthermore, the terms in Eq. (14), $\left(\frac{\mathbf{p}_k^{**}}{\rho^n}\right)_k$, $k=\mathbf{x},\mathbf{y},\mathbf{z}$, are approximated to $\frac{1}{\rho^n}\left(\mathbf{p}_k^{**}\right)_k$ as stated in earlier work [20]. Although Eq. (14) results in slightly higher accuracy, $\left(\frac{\mathbf{p}_k^{**}}{\rho^n}\right)_k$ is not useful for a complex grid systems. Finally, the modified FD form is as follows:

$$\begin{aligned}\mathbf{G}_1=\alpha(\mathbf{E}_1)\frac{1}{\Delta\mathbf{x}_i}&\left\{\frac{2}{\rho_{i+1,j}^{n}+\rho_{i,j}^{n}}\frac{\mathbf{p}_{i+1,j}^{**}-\mathbf{p}_{i,j}^{**}}{\Delta\mathbf{x}_{\mathbf{c}i}}-\frac{2}{\rho_{i,j}^{n}+\rho_{i-1,j}^{n}}\frac{\mathbf{p}_{i,j}^{**}-\mathbf{p}_{i-1,j}^{**}}{\Delta\mathbf{x}_{\mathbf{c}i-1}}\right\}\\ +(1-\alpha(\mathbf{E}_1))\frac{1}{\Delta\mathbf{x}_i}&\left\{\frac{2}{\rho_{i+1,j}^{n}+\rho_{i,j}^{n}}\frac{\mathbf{p}_{i+1,j}^{n}-\mathbf{p}_{i,j}^{n}}{\Delta\mathbf{x}_{\mathbf{c}i}}-\frac{2}{\rho_{i,j}^{n}+\rho_{i-1,j}^{n}}\frac{\mathbf{p}_{i,j}^{n}-\mathbf{p}_{i-1,j}^{n}}{\Delta\mathbf{x}_{\mathbf{c}i-1}}\right\},\end{aligned}$$

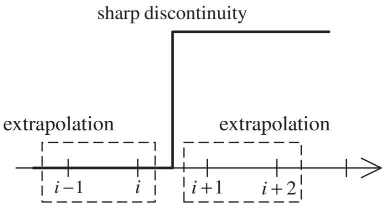


Fig. 3. The improvement in the derivative advancement [15].

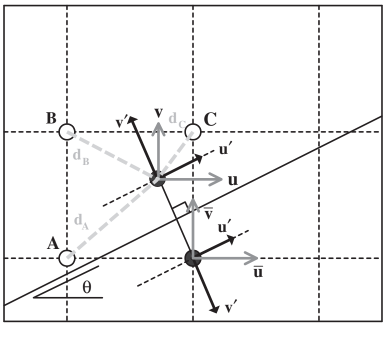


Fig. 4. The inclined surface boundary condition method.

$$\begin{aligned}\mathbf{G}_2=\alpha(\mathbf{E}_2)\frac{1}{\Delta\mathbf{y}_j}&\left\{\frac{2}{\rho_{i,j+1}^{n}+\rho_{i,j}^{n}}\frac{\mathbf{p}_{i,j+1}^{**}-p_{i,j}^{**}}{\Delta\mathbf{y}_{\mathbf{c}j}}-\frac{2}{\rho_{i,j}^{*}+\rho_{i,j-1}^{*}}\frac{\mathbf{p}_{i,j}^{**}-\mathbf{p}_{i,j-1}^{**}}{\Delta\mathbf{y}_{\mathbf{c}j-1}}\right\}\\ +(1-\alpha(\mathbf{E}_2))\frac{1}{\Delta\mathbf{y}_j}&\left\{\frac{2}{\rho_{i,j+1}^{n}+\rho_{i,j}^{n}}\frac{\mathbf{p}_{i,j+1}^{n}-\mathbf{p}_{i,j}^{n}}{\Delta\mathbf{y}_{\mathbf{c}j}}-\frac{2}{\rho_{i,j}^{n}+\rho_{i,j-1}^{n}}\frac{\mathbf{p}_{i,j}^{n}-\mathbf{p}_{i,j-1}^{n}}{\Delta\mathbf{y}_{\mathbf{c}j-1}}\right\}.\end{aligned}$$

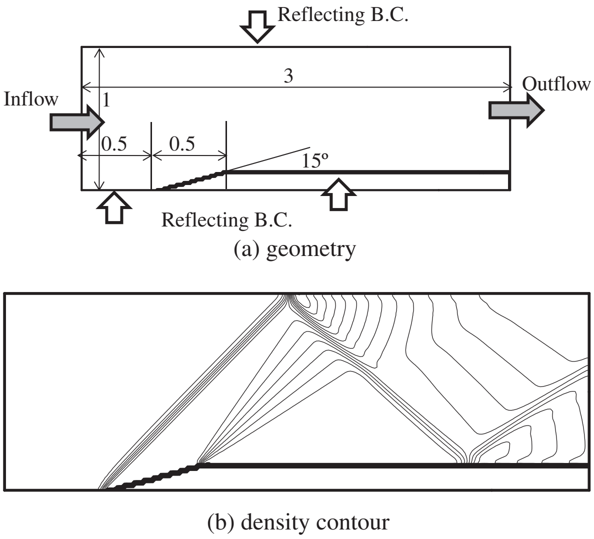
where

$$\begin{aligned}&\Delta\mathbf{x}_i=(\mathbf{x}_{i+1}+\mathbf{x}_{i-1})/2,\quad \Delta\mathbf{x}_{\mathbf{c}i}=\mathbf{x}_i+\mathbf{x}_{i+1},\\ &\Delta\mathbf{y}_j=(\mathbf{y}_{j+1}+\mathbf{y}_{j-1})/2,\quad \Delta\mathbf{y}_{\mathbf{c}j}=\mathbf{y}_j+\mathbf{y}_{j+1}.\end{aligned}$$

3.3. Modification of the derivative advancement

In the literature [15], it was announced that numerical diffusion occurs around shape discontinuity regions when central spatial differencing is used in the non-advection process.

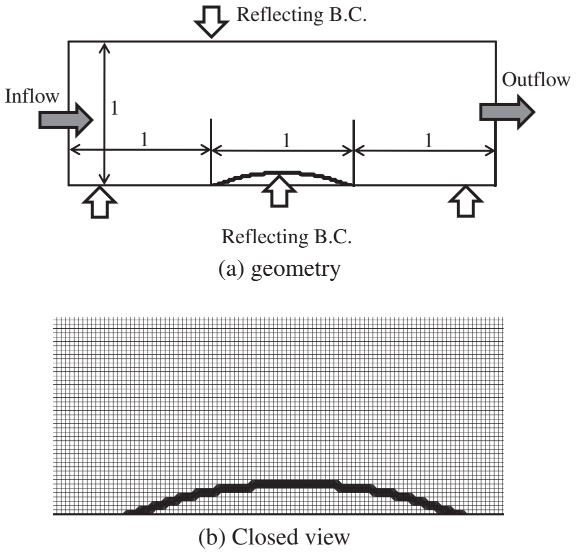
Fig. 3 shows the concept of the enhancement of the stability for the spatial derivative variables by extrapolation. Essentially, the properties at the interface are discontinuous. Thus, differentiating



(a) geometry

(b) density contour

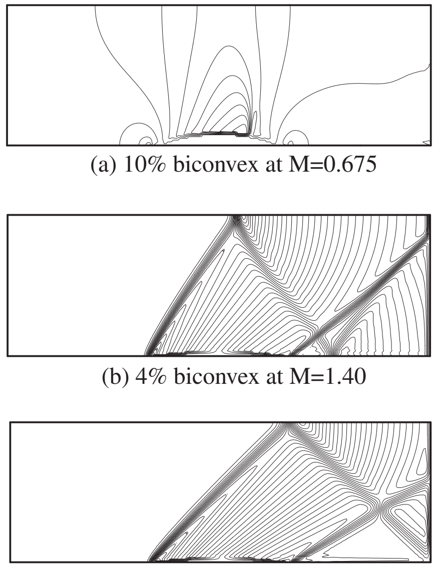
Fig. 5. Density contour of a wedge in a channel.



(a) geometry

(b) Closed view

Fig. 6. Geometry for the transonic/supersonic flow problem.

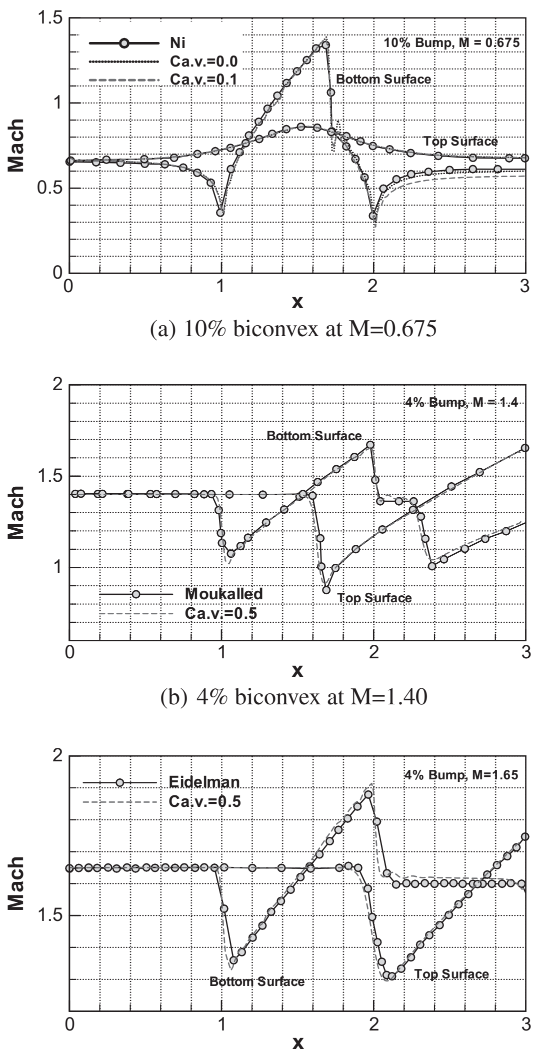


(a) 10% biconvex at M=0.675

(b) 4% biconvex at M=1.40

(c) 4% biconvex at M=1.65

Fig. 7. Density contour for the transonic/supersonic flow problem.



(a) 10% biconvex at M=0.675

(b) 4% biconvex at M=1.40

(c) 4% biconvex at M=1.65

Fig. 8. Local Mach number plots for the upper and lower surface.

spatially across an interface gives the surrounding stencils unphysical values. In terms of continuity, the surrounding stencils should be extrapolated from each phase.

$$\partial_{\mathbf{x}}\mathbf{f}_{i,j}^{n+1} - \partial_{\mathbf{x}}\mathbf{f}_{i,j}^{*} = \frac{1}{2}\left(\frac{\mathbf{d}_{i+1,j} - \mathbf{d}_{i,j}}{\mathbf{h}} + \frac{\mathbf{d}_{i,j} - \mathbf{d}_{i-1,j}}{\mathbf{h}}\right), \tag{17}$$

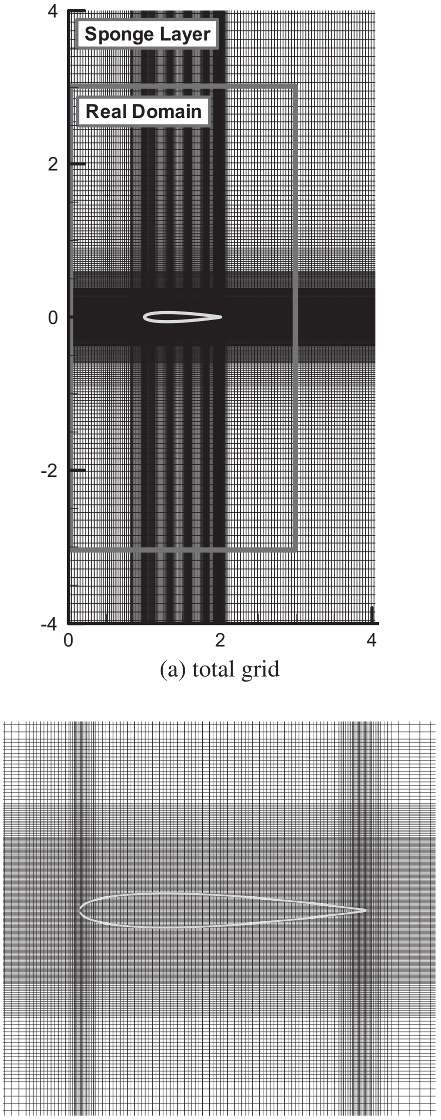
Here, $\mathbf{d}_{i,j} = \mathbf{f}_{i,j}^{n+1} - \mathbf{f}_{i,j}^{*}$. Eq. (17) is modified by introducing an indicator parameter, $\mathbf{H}$, characterized by the color function, ϕ, [15].

$$\mathbf{H}_{i+1/2,j} = \begin{cases} 1 & \text{for } \phi_{i+1,j}^{n+1} \cdot \phi_{i,j}^{n+1} > 0, \\ 0 & \text{otherwise}, \end{cases}$$

$$\mathbf{H}_{i,j+1/2} = \begin{cases} 1 & \text{for } \phi_{i,j+1}^{n+1} \cdot \phi_{i,j}^{n+1} > 0, \\ 0 & \text{otherwise}, \end{cases}$$

$$\partial_{\mathbf{x}}\mathbf{f}_{i,j}^{n+1} - \partial_{\mathbf{x}}\mathbf{f}_{i,j}^{*} = \frac{1}{2}\left(\mathbf{H}_{i+1/2,j}\frac{\mathbf{d}_{i+1,j} - \mathbf{d}_{i,j}}{\mathbf{h}} + \mathbf{H}_{i-1/2,j}\frac{\mathbf{d}_{i,j} - \mathbf{d}_{i-1,j}}{\mathbf{h}}\right),$$

$$\partial_{\mathbf{y}}\mathbf{f}_{i,j}^{n+1} - \partial_{\mathbf{y}}\mathbf{f}_{i,j}^{*} = \frac{1}{2}\left(\mathbf{H}_{i,j+1/2}\frac{\mathbf{d}_{i,j+1} - \mathbf{d}_{i,j}}{\mathbf{h}} + \mathbf{H}_{i,j-1/2}\frac{\mathbf{d}_{i,j} - \mathbf{d}_{i,j-1}}{\mathbf{h}}\right).$$



(a) total grid

(b) closed view

Fig. 9. Cartesian grid system for steady and unsteady aerodynamics analyses.

4. Boundary conditions

4.1. Solid surface boundary condition for an arbitrary surface

The CCUP method used in this paper adopts a Cartesian grid system. The Cartesian grid system eliminates the coordinate transformation procedure so that the accuracy of the scheme can be maintained. Moreover, the physical meaning of the properties can easily be obtained, as it does not transform the governing equations into another form.

However, the mismatch between the surface and grids results in severe numerical errors in the solution. In this case, fine meshes are not a proper remedy. In this paper, we assume that solid subjects do not change their phase. The shape is not deteriorated or generated in time. Thus, a solid surface is defined explicitly and updated in a step-by-step manner.

Therefore, the solid surface boundary condition depicted in Fig. 4 is updated after the solid surface is confirmed.

Fig. 4 shows a schematic diagram of an inclined surface. Here, the black circles represent the grid point, in which a special boundary condition should be applied. The reflecting boundary condition and a slip or non-slip boundary condition can be applied. In this paper, because Euler equations are used, the slip boundary condition is used. In Fig. 4, the black circle below the solid line or solid surface has a reflected point above the solid line. In fact, the reflected point contains the actual flow properties, while the original point has their reflected values. Because the reflected point does not coincide with the grids, the properties must be interpolated by the surrounding fluid points.

$$\mathbf{f} = \sum_{i=1}^{\mathbf{N}} \left(\frac{\sum_{j=1}^{\mathbf{N}} \mathbf{d}_j - \mathbf{d}_i}{(\mathbf{N}-1)\sum_{j=1}^{\mathbf{N}} \mathbf{d}_j} \cdot \mathbf{f}_i \right). \tag{18}$$

Eq. (18) shows the interpolated values at the reflected point, where N refers to the total number of surrounding fluid points excluding the solid region. Here, N is 2 or 3. If N exceeds 3, the CIP method is used to estimate the properties of the reflected point. $\mathbf{d}_i$ refers to the distance from the reflected point to the surrounding point. $\mathbf{f}$ and $\mathbf{f}_i$ are the flow properties of the reflected point and surrounding points, respectively.

After the flow properties are calculated from Eq. (18), the slip boundary condition is then applied.

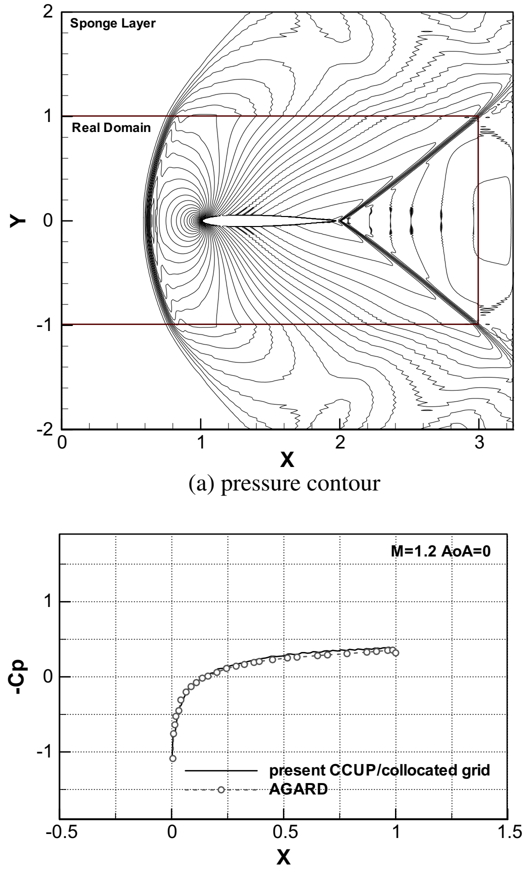
$$\bar{\mathbf{u}} = \frac{1}{1+\tan^2\theta}\{(1-\tan^2\theta)\cdot\mathbf{u} + 2\tan\theta\cdot\mathbf{v}\}, \tag{19}$$

$$\bar{\mathbf{v}} = \frac{1}{1+\tan^2\theta}\{2\tan^2\theta\cdot\mathbf{u} + (\tan^2\theta - 1)\cdot\mathbf{v}\}. \tag{20}$$

As with typical reflecting boundary conditions, because the pressure and density do not have a direction, their values are simply shifted to the original point. However, the velocity components defined on Cartesian grids are converted into the inclined surface coordinates and transformed again into Cartesian grids, as presented in Eqs. (19) and (20).

4.2. Far-field boundary condition

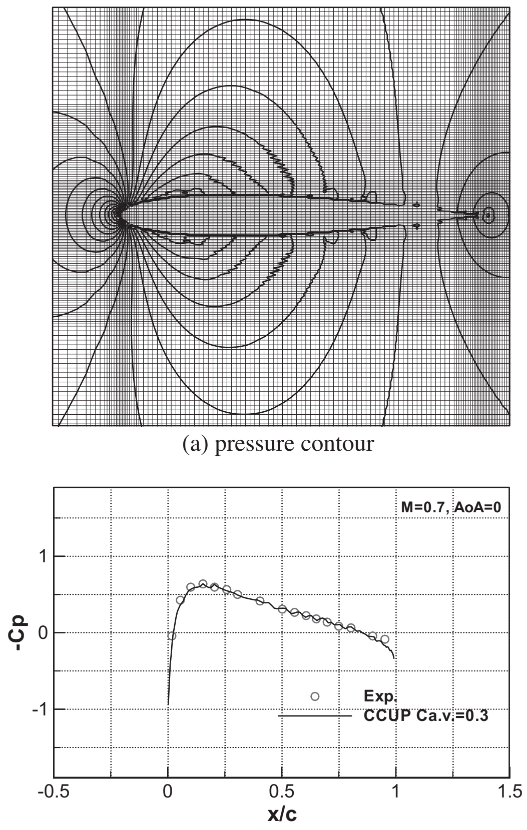
To reduce the computational region, the sponge layer method, an absorbing layer method, is introduced. Compared to other absorbing layer methods, the sponge layer method has been thoroughly investigated on account of its simplicity. However, it is important to construct a proper positive damping profile. Similar



(a) pressure contour

(b) C_p plot

Fig. 10. Steady aerodynamics of NACA0012 at Mach = 1.2 and AoA = 0.0°.



(a) pressure contour

(b) C_p plot

Fig. 11. Steady aerodynamics of NACA0012 at Mach = 0.7 and AoA = 0.0°.

to earlier work [22], the sponge layer effect is achieved here by introducing an absorbing boundary condition into the momentum equations, Eqs. (7) and (8).

In earlier work [23], the auxiliary term for the absorbing boundary condition is defined as follows:

$$\overrightarrow{\mathbf{F}}_{sponge\,Layer} = \alpha(\mathbf{x}) \cdot (\vec{\mathbf{u}}_{ref} - \vec{\mathbf{u}}). \tag{21}$$

The damping profile α is defined by the tangent hyperbolic function.

$$\alpha(\mathbf{x}) = \alpha \cdot \left\{1 - \frac{1}{2}\left(\tanh\frac{8}{\varepsilon}\left(\mathbf{x} - \mathbf{x}_L - \frac{\varepsilon}{2}\right) - \tanh\frac{8}{\varepsilon}\left(\mathbf{x} - \mathbf{x}_R + \frac{\varepsilon}{2}\right)\right)\right\} \tag{22}$$

Here, $\varepsilon = \mathbf{L} \cdot \frac{n_{Layer}}{n}$. This profile has a globally smooth characteristic and is advantageous when using spatial discretization [22].

In this paper, Eq. (21) is extended to multiple dimensions.

$$\mathbf{F}_{sponge\,Layer} = \alpha \cdot (\phi_{ref} - \phi). \tag{23}$$

The damping profiles are presented below.

$$\alpha = \alpha_{\max} \cdot \left\{1 - \frac{1}{2}\left(\tanh\frac{8}{\varepsilon}\left(\mathbf{x} - \mathbf{x}_L - \frac{\varepsilon}{2}\right) - \tanh\frac{8}{\varepsilon}\left(\mathbf{x} - \mathbf{x}_R + \frac{\varepsilon}{2}\right)\right)\right\},$$

$$\alpha = \alpha^{\mathbf{y}}_{\max} \cdot \left\{1 - \frac{1}{2}\left(\tanh\frac{8}{\varepsilon}\left(\mathbf{y} - \mathbf{y}^{+}_{\text{center}} - \frac{\varepsilon}{2}\right) - \tanh\frac{8}{\varepsilon}\left(\mathbf{y} - \mathbf{y}_{jmax} + \frac{\varepsilon}{2}\right)\right)\right\},$$

$$\alpha = \alpha^{\mathbf{y}}_{\max} \cdot \left\{1 - \frac{1}{2}\left(\tanh\frac{8}{\varepsilon}\left(\mathbf{y} - \mathbf{y}_0 - \frac{\varepsilon}{2}\right) - \tanh\frac{8}{\varepsilon}\left(\mathbf{y} - \mathbf{y}^{-}_{\text{center}} + \frac{\varepsilon}{2}\right)\right)\right\}.$$

In these equations, ϕ includes all flow properties. $\alpha_{\mathbf{max}}$, the user-defined parameter, is 1 for the sponge layer and 0 otherwise. $\mathbf{y}^{+}_{\text{center}}$ and $\mathbf{y}^{-}_{\text{center}}$ are the center locations.

5. Numerical analyses

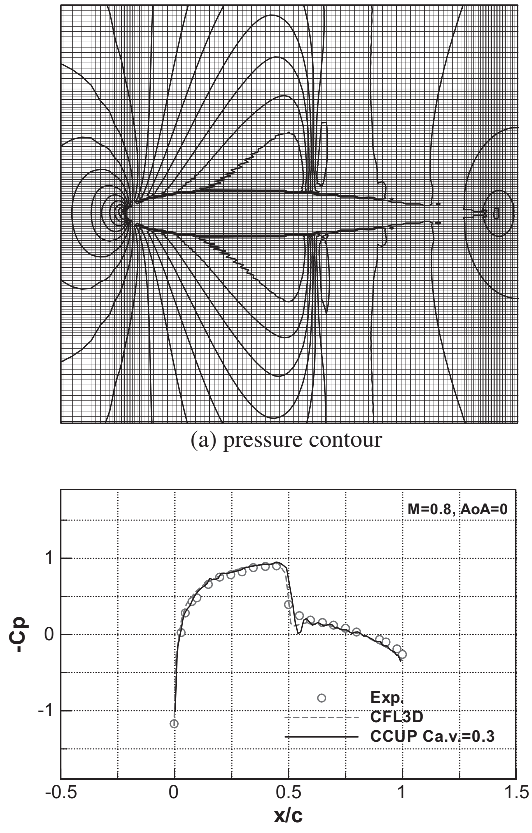
5.1. Benchmarking test: flow over a wedge

To consider compressible flow problems, an artificial viscosity condition is implemented [24]. Unlike shock capturing schemes such as TVD (total variation diminishing) and ENO (essentially non-oscillatory), which use implicit artificial viscosity, explicit artificial viscosity is very simple.

$$\vec{\mathbf{q}} = \mathbf{c}\left\{-\rho\left(\mathbf{C_s} - \frac{\gamma+1}{2}\Delta\vec{\mathbf{u}}\right)\right\}. \tag{24}$$

Eq. (24) is valid only for $\Delta\vec{\mathbf{u}} \leqslant 0$; otherwise, $\vec{\mathbf{q}} = \vec{0}$. This represents a linear combination of linear viscosity and quadratic Von Neumann–Richtmyer viscosity [24].

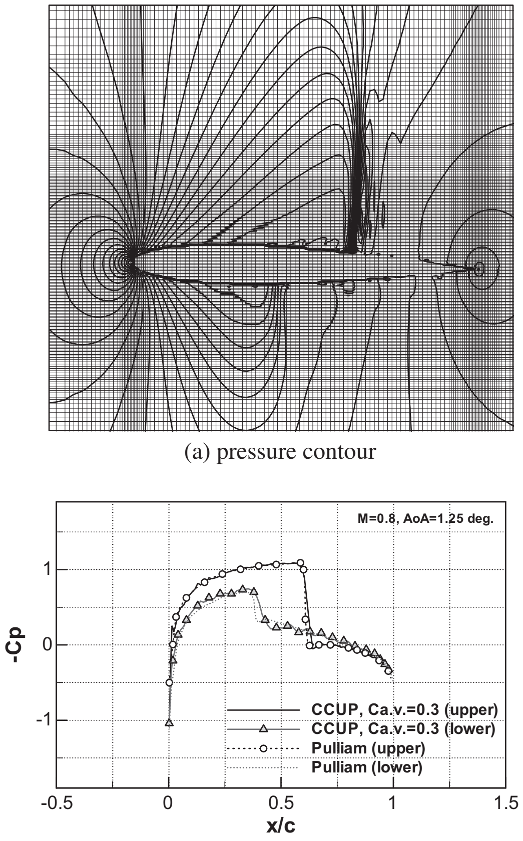
Fig. 5 shows the shock wave reflections on the solid surface. The entire computational domain has three length units in width and one length unit in height. A supersonic flow at Mach 2.0 comes in from the left end and goes out through the right end of the channel. Fig. 5a shows the geometry information. The wedge is located 0.5 length units from the left end and the wedge width is 0.5 length units. The slope of the wedge is 15°. A reflecting boundary condition is applied to all surfaces except for the inlet and outlet surfaces, and the inclined surface boundary condition presented in Section 4.1 is applied to the wedge surface. The computational domain is discretized at 240 × 80. The density contour is plotted in Fig. 5b. According to Fig. 5b, an oblique shock on the top surface of the channel is located at around 1.45 length units. A second reflection of the shock wave on the solid surface is located at around 2.37 length units. The result shows good agreement with earlier results [25].



(a) pressure contour

(b) C$_\text{p}$ plot

Fig. 12. Steady aerodynamics of NACA0012 at Mach = 0.8 and AoA = 0.0°.



(a) pressure contour

(b) C$_\text{p}$ plot

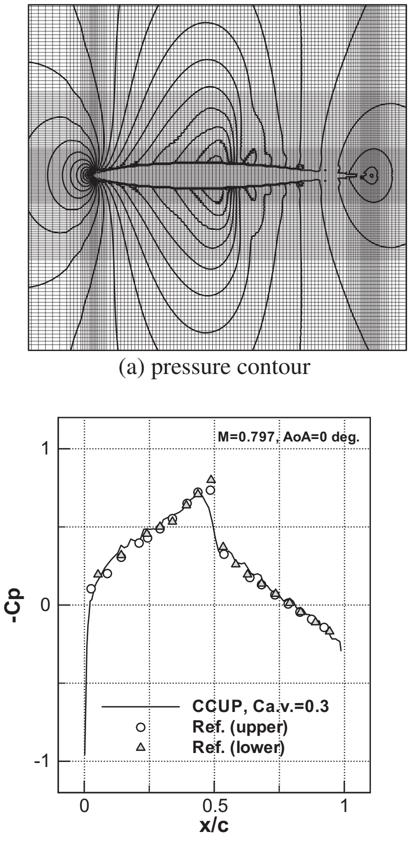
Fig. 13. Steady aerodynamics of NACA0012 at Mach = 0.8 and AoA = 1.25°.

5.2. Benchmarking test: flows over bumps

For flows over biconvex shape bumps, the well-known transonic/supersonic flows problem is tested.

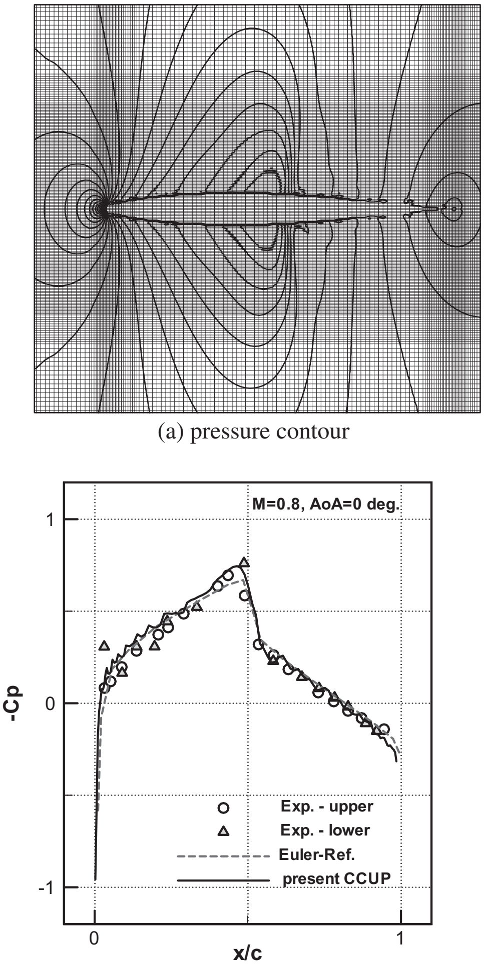
Fig. 6a represents a schematic diagram of a channel with a bump inside. Fig. 6b shows a closed view of a biconvex bump in a Cartesian grid. Steady transonic and supersonic flows at Mach = 0.675, 1.40, and 1.65 come in from the left end of the channel and come out at the right end of the channel. A solid biconvex bump with a maximum height of 4% or 10% is located in the middle of the channel. The computational domain has three length units in width and one length unit in height. The computational domain is discretized at 240 × 80. A reflecting boundary condition is applied to all solid surfaces. An inclined surface boundary condition is applied to the biconvex surface. Three cases, a 10% biconvex bump at M = 0.675, a 4% biconvex bump at M = 1.40, and a 4% biconvex bump at M = 1.65, are tested.

Fig. 7 shows the density contour for the three cases. Fig. 7a is the result of case 1, the 10% biconvex bump at M = 0.675. According to Fig. 7a, a normal shock wave is generated at a chord position of approximately 71%. Figs. 7b and c illustrate case 2 and case 3, respectively. All cases capture shock waves well and show good agreement with the reference values [26–28]. Fig. 8 presents the local Mach number plot for the top and bottom surface. Fig. 8a–c represents case 1, case 2, and case 3, respectively. All cases are compared with the reference values in the literature [26–28]. In Fig. 8a, the local Mach number coincides well with that in the literature [26]. Although a low estimation of the local Mach number in Fig. 8a is observed in the downstream region, the Mach number on the biconvex surface is well predicted. Following a published value [29], the low estimation is due to the rotational flow after the normal shock near a wall. If the artificial viscosity coefficient is zero, numerical oscillation appears after a shock wave discontinuity. However, a slight change of the coefficient reduces the oscillation. Although the artificial viscosity is explicitly controlled, the Mach contour on the bottom surface does not vary greatly. Furthermore, the Mach contour on the top surface is predicted accurately compared to the reference. For Fig. 8b, the oblique shock wave reflected initially on the top surface is reflected on the bottom surface again. The Mach contour is in good agreement with


(a) pressure contour

(b) C_p plot

Fig. 14. Steady aerodynamics of NACA64a010 at Mach = 0.797 and AoA = 0.0°.


(a) pressure contour

(b) C_p plot

Fig. 15. Steady aerodynamics of NACA64a010 at Mach = 0.800 and AoA = 0.0°.

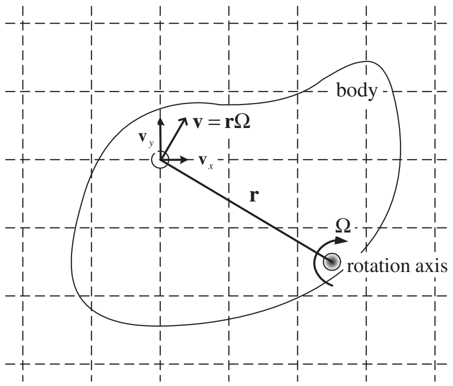


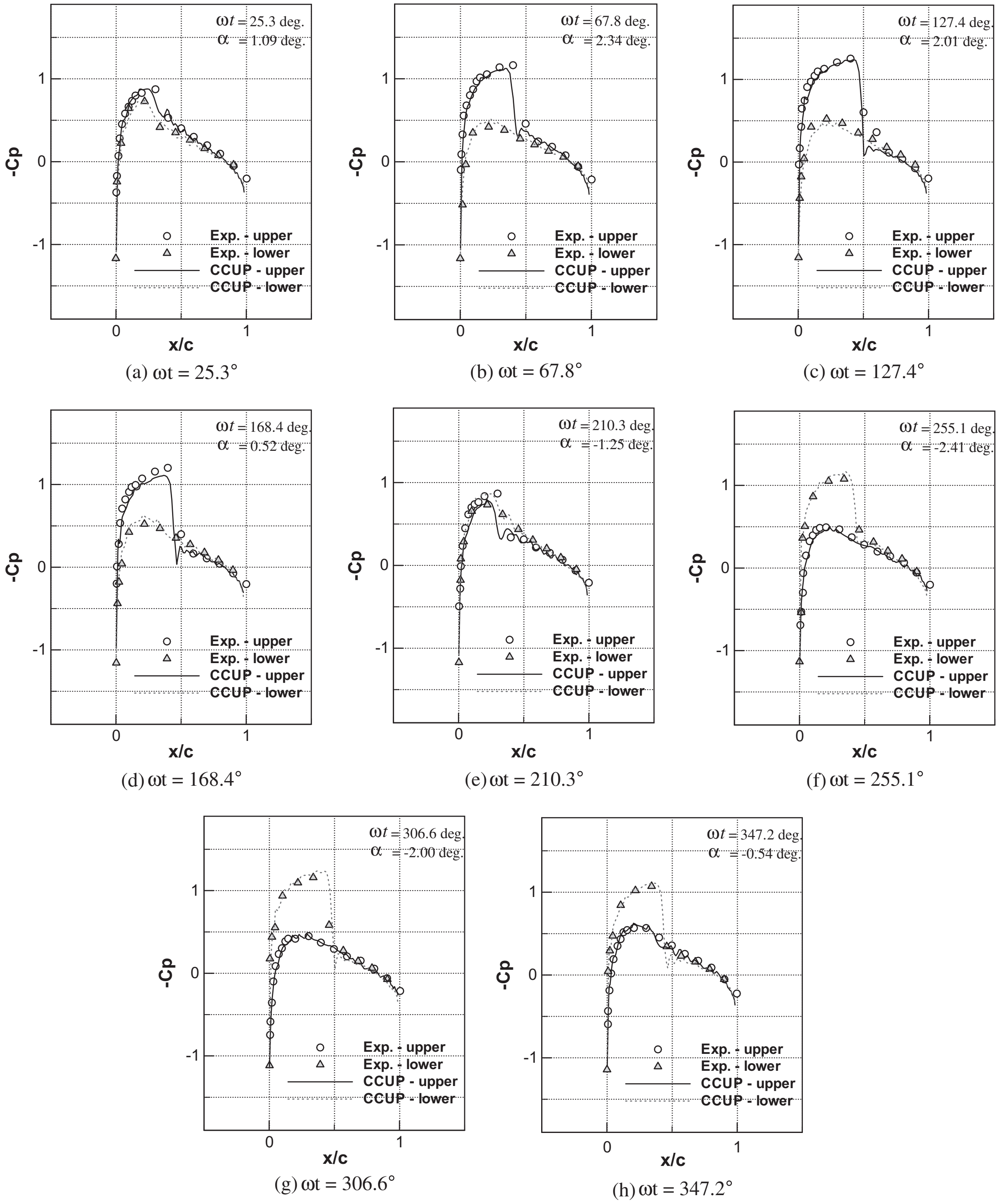
Fig. 16. Unsteady body boundary condition.

the literature value [27]. For Fig. 8c, the Mach contour on the biconvex bump surface is similar to that shown in case 2. However, the oblique shock generated at the leading edge of the biconvex goes out the computational domain. As shown in Fig. 8c, the shock wave is estimated precisely. A thin viscous layer affects the after-shock region, as shown in Fig. 8a. However, the effect is not severe. Hence, all results show good agreement with the reference results.

5.3. Steady aerodynamics based the multiphase concept

Fig. 9 presents the two-dimensional Cartesian grids consisting of the real domain and the sponge layer. A solid airfoil is embedded in the middle of the grids in Fig. 9a. Fig. 9b shows a closed view of all of the grids in Fig. 9a. The total size of the grids is 4 × 3, but this is not fixed. It can vary according to the problem. The sponge layer surrounds three faces of the real domain, the top, under, and downstream region faces. The top and bottom surface of the real domain are located at 3 and -3 in the y-direction, respectively, and at 3 in the case of the downstream surface. The damping profile of the x-component is used when $x > 3$. The damping profile of the y^{+}-component is used when $y > 3$ and $x < 3$, and the y^{-}-component is applied when $y < -3$ and $x < 3$.

The larger gradient part of an airfoil is discretized manually without any numerical treatment, because the sensitivity of the



(a) ωt = 25.3°

(b) ωt = 67.8°

(c) ωt = 127.4°

(d) ωt = 168.4°

(e) ωt = 210.3°

(f) ωt = 255.1°

(g) ωt = 306.6°

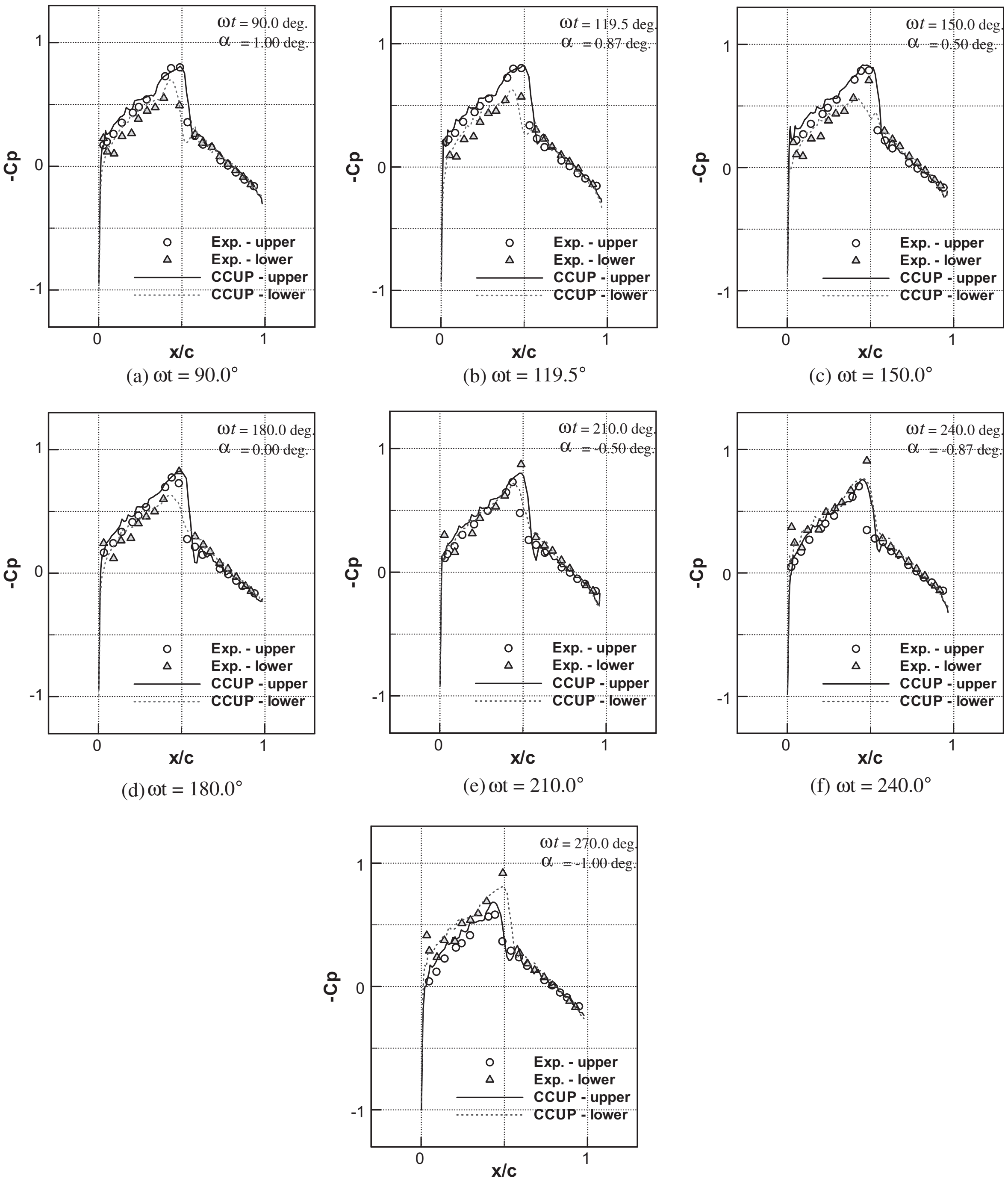
(h) ωt = 347.2°

Fig. 17. Pitching unsteady aerodynamics of NACA0012 at Mach = 0.755 and AoA = 0.0°.

results for the grid system is relatively low compared to typical compressible flow solvers. However, to improve the accuracy around the airfoil surface, the grids are clustered around it. The coefficient of the artificial viscosity is 0.3. The time step is 0.001–0.002. As shown in Fig. 9, there is no specific airfoil grid here, but the solid flow properties are given inside the total flow region and both the fluid and the solid part are solved together in one code, as if it is a multiphase flow.

Figs. 10–15 show the pressure coefficient of the airfoil surface and the pressure contour of the flow field around the airfoil for various steady aerodynamic conditions. In Figs. 10–15, the upper figures are the pressure contour and the lower figures present the pressure coefficient comparisons.

For Fig. 10, a NACA0012 airfoil, which has a thickness of 12% about the chord length at Mach 1.2 and an AoA (angle of attack) of 0.0°, is tested. The total domain is 3.25 × 4. The outside of the real domain is a sponge layer when $x > 3$, $y > 1$, or $y < -1$. For a supersonic case, strong shock waves are generated at the leading and trailing edge of the airfoil and are reflected from the leading edge. The pressure coefficient on the surface in Fig. 10b was compared with that in the literature [30], showing a very accurate result. The user-defined damping factor, $\alpha_{\mathbf{max}}$, is 1.



(a) ωt = 90.0°

(b) ωt = 119.5°

(c) ωt = 150.0°

(d) ωt = 180.0°

(e) ωt = 210.0°

(f) ωt = 240.0°

(g) ωt = 270.0°

Fig. 18. Pitching unsteady aerodynamics of NACA64a010 at Mach = 0.8 and AoA = 0.0°.

For Fig. 11, the same airfoil used in Fig. 10 is tested with Mach 0.7 and an AoA of 0.0° as a subsonic case. The total computational domain is identical to that shown in Fig. 9. The damping factor is 20, and the coefficient of the artificial viscosity is 0.3. However, the shock wave is generated as shown in Figs. 11a and b. The pressure coefficient is validated with experimental data from the literature [31].

For Fig. 12, the Mach number is increased to 0.8 as a transonic case. The damping factor and artificial viscosity parameter are 20, 0.3, respectively. A strong shock wave is appeared at around 50% of the chordwise location in Fig. 12a and b. When compared with experimental and numerical data, CFL3D, which is based on N–S equations [32], the location and strength of the shock wave are accurately predicted.

Contrary to the cases of Figs. 10–12, Fig. 13 shows the results at Mach 0.8 and an AoA of 1.25° as a typical non-symmetric case. The parameters of the damping factor and the artificial viscosity are identical to those in Fig. 12. Strong shock waves are developed on both the upper and lower surface. In this case, the inlet boundary condition does not change, but the solid phase is inclined. For the upper surface, the shock wave is around 60% in the chordwise location. For the lower surface, the shock wave chordwise location is 40%. The result is comparable to that in earlier work [33].

For Figs. 14 and 15, an airfoil with 10% of the thickness, i.e., NACA64a010, was analyzed. Analyses of the steady aerodynamics at Mach number of 0.797 for Fig. 14 and 0.8 for Fig. 15 were done. For all cases, the angle of attack and the damping factor were 0, but the artificial viscosity parameter was unchanged. The result presents a shock wave at around 50% of the chordwise location, showing good agreement with the reference values [34,35].

5.4. Unsteady aerodynamics based on the multiphase concept

For unsteady aerodynamics, Eqs. (19) and (20) must be modified to include the rotational and translational motion of a rigid body, as follows:

$$\bar{\mathbf{u}} = \frac{1}{1+\tan^2\theta}\left\{(1-\tan^2\theta)\cdot\mathbf{u} + 2\tan\theta\cdot\mathbf{v}\right\} + \mathbf{v}_x^{\mathbf{R}} + \mathbf{v}_x^{\mathbf{T}}, \qquad (25)$$

$$\bar{\mathbf{v}} = \frac{1}{1+\tan^2\theta}\left\{2\tan\theta\cdot\mathbf{u} + (\tan^2\theta-1)\cdot\mathbf{v}\right\} + \mathbf{v}_y^{\mathbf{R}} + \mathbf{v}_y^{\mathbf{T}}. \qquad (26)$$

As shown in Fig. 16, the *x*- and *y*-component of the rotational velocities, $\mathbf{v}_x$ and $\mathbf{v}_y$, of a rigid body are defined around the rotational axis, Ω. These are denoted as $\mathbf{v}^{\mathbf{R}}$ in Eqs. (25) and (26). Furthermore, $\mathbf{v}^{\mathbf{T}}$ represents the translational velocity of a body.

The unsteady aerodynamic forces are achieved by pitching a body sinusoidally according to the following function:

$$\alpha = \alpha_m + \alpha_0\sin(\omega t) = \alpha_m + \alpha_0\sin(\mathbf{k}\tau). \qquad (27)$$

In this equation, α is the pitch angle, α_m is the averaged pitch angle, and α_0 is the pitch amplitude. In this paper, the unsteady motion is assumed to behave as a sine wave with frequency ω. In Eq. (27), the reduced frequency $\mathbf{k}$ is introduced. It is defined as follows:

$$\mathbf{k} = \frac{\omega\mathbf{b}}{\mathbf{V}_\infty}, \quad \tau = \frac{\mathbf{V}_\infty t}{\omega}$$

Here, $\mathbf{v}_\infty$ refers to the freestream velocity and $\mathbf{b}$ is the half-chord value. τ is the non-dimensional time.

Fig. 17 shows the unsteady aerodynamics in the pitching motion of NACA0012. The Mach number is 0.755. α_m and α_0 are 0.016° and 2.51°, respectively. The reduced frequency, $\mathbf{k}$, is 0.0841. In this case, an initial calculation for several cycles to avoid numerical transient responses is not performed. Fig. 17 expresses the instantaneous pressure coefficient distribution on the surface of an airfoil. These figures were obtained at phase angles of 25.3°, 67.8°, 127.4°, 168.4°, 210.3°, 255.1°, 306.1° and 347.2°. In the figures, the solid black line denotes the upper surface, while the gray line indicates the lower surface. The symbols denote the experimental data. All pressure coefficients are compared with the experimental data in a specific study [36]. The results show an accurate and sharp prediction of the shock waves on upper and lower surface and their transient motions.

Fig. 18 shows the unsteady aerodynamics in the pitching motion of NACA64a010. The Mach number is 0.800. α_m and α_0 are 0.0° and 1.01°, respectively. The reduced frequency $\mathbf{k}$ is 0.202. The initial numerical transient responses are not handled in this case either. The instantaneous pressure coefficients on the surfaces are obtained at phase angles of 90°, 150°, 180°, 210°, 240° and 270°. When the pressure coefficients are compared with those from the literature [37], the results are in good agreement with the experimental data.

6. Conclusions

In this research, the improved CIP/CCUP method is used to analyze steady and unsteady aerodynamics including solids. Unlike the conventional aerodynamics fields, the multiphase concept is applied. Although the solid surface did not precisely match the Eulerian grids as Lagrangian grid systems do, the simple ghost-cell-based arbitrary boundary condition works well enough with the CIP/CCUP method to predict accurate shock waves on airfoils. For a far-field boundary condition, the sponge layer method was implemented. It does not affect the real domain inside, even for time-marched solutions such as unsteady sinusoidal motions. In conclusion, the multiphase concept is applicable to compressible aerodynamics problems including the structural motions, which sometimes break or seriously distort the CV. Furthermore, the extension to the three-dimensional application is straightforward. Although the remained difficulties are to solve three-dimensional object within a reasonable computing cost maintaining good accuracy, we are developing a three-dimensional multiphase code combined with the structural dynamics.

Acknowledgment

This work was supported by the second stage of the Brain Korea 21 Project in 2011. The authors are grateful for this support.

References

[1] Garcia JA. Numerical investigation of nonlinear aeroelastic effects on flexible high-aspect-ratio wings. J. Aircraft 2005;42(4):1025–36.

[2] Harlow FH, Welch JE. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. Phys Fluids 1965;8(12):2182–9.

[3] Unverdi SO, Tryggvason G. Computations of multi-fluid flows. Physica D 1992;60:70–83.

[4] Sussman M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two-phase flow. J Comput Phys 1994;114:146–59.

[5] Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. J Comput Phys 1981;39(1):201–25.

[6] Yabe T et al. Challenge of CIP as a universal solver for solid, liquid and gas. Int J Numer Method Fluid 2005;47:655–76.

[7] Takewaki H, Nishiguchi A, Yabe T. Cubic interpolated pseudo-particle method (CIP) for solving hyperbolic-type equations. J Comput Phys 1985;61(2):261–8.

[8] Yabe T, Xiao F, Utsumi T. The constrained interpolation profile method for multiphase analysis. J Comput Phys 2001;169:556–93.

[9] Tanaka R, Nakamura T, Yabe T. Constructing exactly conservative scheme in a non-conservative form. Comput Phys Commun 2000;126:232–43.

[10] Yabe T, Xiao F. A method to trace sharp interface of two fluids in calculations involving shocks. Shock Waves 1994;4:101–8.

[11] Yabe T, Aoki T. A universal solver for hyperbolic equations by cubic-polynomial interpolation I. one-dimensional solver. Comput Phys Commun 1991;66:219–32.

[12] Yabe T et al. A universal solver for hyperbolic equations by cubic-polynomial interpolation II. Two- and three-dimensional solvers. Comput Phys Commun 1991;66:233–42.

[13] Yoon SY, Yabe T. The unified simulation for incompressible and compressible flow by the predictor–corrector scheme based on the CIP method. Comput Phys Commun 1999;119:149–58.

[14] Ida M. An improved unified solver for compressible and incompressible fluids involving free surfaces. Part I. Convection. Comput Phys Commun 2000;132:44–65.

[15] Ida M. An improved unified solver for compressible and incompressible fluids involving free surfaces. Part II. Multi-time-step integration and applications. Comput Phys Commun 2003;150:300–22.

[16] Takizawa K et al. Computation of free-surface flows and fluid–object interactions with the CIP method based on adaptive meshless Soroban grids. Comput Mech 2007;40:167–83.

[17] Rhie CM, Chow WL. Numerical study of the turbulent flow past an airfoil with trailing edge separation. AIAA J 1983;21(11):1525–32.

[18] Aoki T. Interpolated differential operator (IDO) scheme for solving partial differential equations. Comput Phys Commun 1997;102:132–46.

[19] Xiao F, Ikebata A, Hasegawa T. Numerical simulations of free-interface fluids by a multi-integrated moment method. Comput Struct 2005;83:409–23.

[20] Yabe T, Wang PY. Unified numerical procedure for compressible and incompressible fluid. J Phys Soc Jpn 1991;60(7):2105–8.

[21] Nakamura T et al. Exactly conservative semi-Lagrangian scheme for multi-dimensional hyperbolic equations with directional splitting technique. J Comput Phys 2001;174:171–207.

[22] Hu C, Kashiwagi M. A CIP-based method for numerical simulations of violent free-surface flows. J Mar Sci Technol – Japan 2004;9(4):143–57.

[23] Hagstrom T, Appelö D. Experiments with hermite methods for simulating compressible flows: Runge–Kutta time-stepping and absorbing layers. AIAA-2007-3505.

[24] Ogata Y, Yabe T. Shock capturing with improved numerical viscosity in primitive Euler representation. Comput Phys Commun 1999;119:179–93.

[25] Chandar DDJ, Rao SVR, Deshpande SM. A one point shock capturing kinetic scheme for hyperbolic conservation laws. In: The third international conference on computational fluid dynamics, ICCFD3; 2004.

[26] Ni RH. A multiple grid scheme for solving the Euler equations. AIAA J 1982;20:1565–71.

[27] Moukalled F, Darwish M. A high-resolution pressure-based algorithm for fluid flow at all speeds. J Comput Phys 2001;168:101–33.

[28] Eidelman S, Colella P, Shreeve RP. Application of the Godunov method and its second-order extension to cascade flow modeling. AIAA J 1984;22(11):1609–15.

[29] Taghaddosi F et al. An adaptive least-squares method for the compressible Euler equations. Int J Numer Method Fluid 1999;31:1121–39.

[30] Yoshihara H, Sacher P. Test cases for inviscid flow field methods. AGARD AR-211; 1986.

[31] Report of the fluid dynamics panel working group 04. Experimental data base for computer program assessment. AGARD AR-138; 1979.

[32] Robinson BA, Batina JT, Yang HTY. Aeroelastic analysis of wings using the Euler equations with a deforming mesh. AIAA-90-1032-CP:1510–18.

[33] Pullian T. Artificial dissipation models for Euler equations. AIAA J 1986;24(12):1931–40.

[34] Howlett J. Calculation of viscous effects on transonic flow for oscillating airfoils and comparisons with experiments. NASA-TP 2731; 1987.

[35] Chen ZB, Bradshawt P. Calculations of viscous transonic flow over airfoils. AIAA J 1984;22(2):201–5.

[36] Kirshman DJ, Liu F. Flutter prediction by an Euler method on non-moving Cartesian grids with gridless boundary conditions. Comput Fluids 2006;35:571–86.

[37] Chyu WJ, Davis SS, Chang KS. Calculation of unsteady transonic flow over an airfoil. AIAA J 1981;19(6):684–90.