Chapter 2
High-Order Finite Volume Method for the Compressible Flows

This chapter will introduce the basis framework of FVM and the governing equations are the compressible Navier-Stokes equations. The k-exact reconstruction will be specifically described since it is the key for FVM to achieve high-order accuracy. To solve the viscous flows, a new simple yet an efficient scheme is proposed and the viscous boundary conditions are considered through the constrained least-square method. For faster convergence speed, the dual time stepping with LU-SGS iteration is used for time integration. Also MPI parallelization is implemented in the program for solving large-scale problems. For brevity, only the three-dimensional description for the algorithms below is given below. Those two-dimensional descriptions being distinct from three-dimensional ones are also presented.

2.1 The Framework of the Finite Volume Method

This section will describe the basis framework of cell-center finite volume method on unstructured grids. The governing equations of compressible flows are the Navier-Stokes equations,

$$\frac{\partial U}{\partial t}+\frac{\partial\left(F-F_v\right)}{\partial x}+\frac{\partial\left(G-G_v\right)}{\partial y}+\frac{\partial\left(H-H_v\right)}{\partial z}=0 \tag{2.1}$$

where the conservative variables and inviscid flux vectors are

$$U=\begin{pmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ E\end{pmatrix},\quad F=\begin{pmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ \rho uw\\ u(E+p)\end{pmatrix},\quad G=\begin{pmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ \rho vw\\ v(E+p)\end{pmatrix},\quad H=\begin{pmatrix}\rho w\\ \rho uw\\ \rho vw\\ \rho w^2+p\\ w(E+p)\end{pmatrix}, \tag{2.2}$$

and the viscous flux vectors are

W. Li, *Efficient Implementation of High-Order Accurate Numerical Methods on Unstructured Grids*, Springer Theses, DOI: 10.1007/978-3-662-43432-1_2,

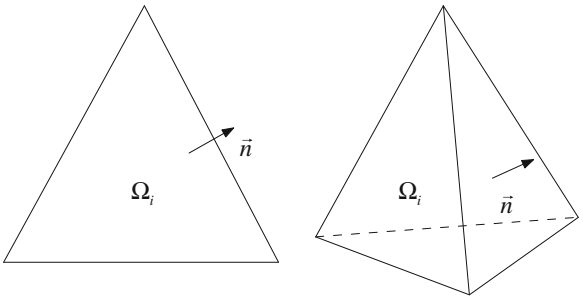

Fig. 2.1 The control volumes on two and three dimensions and the normal vector outwards

$$
\begin{aligned}
F_v &= \left\{0, \tau_{xx}, \tau_{yx}, \tau_{zx}, u\tau_{xx} + v\tau_{yx} + w\tau_{zx} + k\frac{\partial T}{\partial x}\right\}^T, \\
G_v &= \left\{0, \tau_{xy}, \tau_{yy}, \tau_{zy}, u\tau_{xy} + v\tau_{yy} + w\tau_{zy} + k\frac{\partial T}{\partial y}\right\}^T, \\
H_v &= \left\{0, \tau_{xz}, \tau_{yz}, \tau_{zz}, u\tau_{xz} + v\tau_{yz} + w\tau_{zz} + k\frac{\partial T}{\partial z}\right\}^T.
\end{aligned} \tag{2.3}
$$

with the viscous stress

$$
\begin{aligned}
\tau_{xx} &= 2\mu u_x + \lambda(u_x + v_y + w_z), \quad & \tau_{xy} &= \tau_{yx} = \mu(u_y + v_x), \\
\tau_{yy} &= 2\mu v_y + \lambda(u_x + v_y + w_z), \quad & \tau_{xz} &= \tau_{zx} = \mu(u_z + w_x), \\
\tau_{zz} &= 2\mu w_z + \lambda(u_x + v_y + w_z), \quad & \tau_{yz} &= \tau_{zy} = \mu(v_z + w_y).
\end{aligned}
$$

In the above equations, ρ denotes the density, (u, v, w) are the velocity vector, μ is the dynamics viscosity, T is the temperature, E is the total energy and p is the pressure satisfying

$$
E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho\left(u^2 + v^2 + w^2\right)
$$

with a constant ratio of specific heat $\gamma = 1.4$. According to the Stokes hypothesis, one has $\lambda = -2/3$. The heat conductivity is computed using $k = c_p\mu/Pr$, where c_p is the specific heat at constant pressure and Pr is Prandtl number.

For the control volume shown in Fig. 2.1, integrate over the control volume and get

$$
\frac{\mathrm{d}}{\mathrm{d}t}\bar{U}_i + \frac{1}{\bar{\Omega}_i}\oint_{\partial\Omega_i} \mathbf{F}_n \cdot \mathbf{n}\,\mathrm{d}S = 0, \quad \mathbf{F}_n = (F - F_v, G - G_v, H - H_v) \tag{2.4}
$$

where $\bar{\Omega}_i$ is the volume of cell i, $\mathbf{n}$ is the outward unit vector normal to the surface $\partial\Omega_i$, $\bar{U}_i$ is the average for conservative variables on the control volume defined as

Table 2.1 Number of Gaussian points on the triangle and tetrahedron

Order	Triangle			Quadrilateral		
	$K=1$	$K=2$	$K=3$	$K=1$	$K=2$	$K=3$
Surface integral	1	2	2	1	3	4
Volume integral	1	3	4	1	4	5

$$\bar{U}_i = \frac{1}{\bar{\Omega}_i} \int_{\Omega_i} U(x, y, z)\, \mathrm{d}\Omega$$

The surface integral on can be expressed as the summation of integrals on all faces, i.e.,

$$\oint_{\partial\Omega_i} \mathbf{F}_n \cdot \mathbf{n}\, \mathrm{d}S = \sum_{k=1}^{N_F} \int_{\partial\Omega_{i,k}} \mathbf{F}_n \cdot \mathbf{n}\, \mathrm{d}S \approx \sum_{k=1}^{N_F} \left(\sum_{\phi=1}^{N_G} \omega_g \hat{F}_n(U^R(\mathbf{x}_{k,\phi}), U^L(\mathbf{x}_{k,\phi})) \cdot \mathbf{n} S_k \right) \tag{2.5}$$

where the left and right states, U^R, U^L, on the face can be obtained using the high-order k-exact reconstruction in Sect. 2.2, N_F is the number of faces for cell i. The flux integral is computed using the Gaussian integration and N_G is the number of Gaussian integration points. And the number of integration points for the surface and volume integration on different shapes of control volume is shown in Table 2.1. The numerical flux $\hat{F}$ is evaluated using the Riemann solver, for which the Roe and HLL schemes are used in this thesis. The calculation for the viscous flux will be introduced in Sect. 2.3.

2.2 The Time Integration Schemes

For the semi-discretization schemes of (2.4), the time integration scheme is needed to update the cell average. In this chapter, the explicit time integration is first considered. Specifically, the third- and fourth-order Runge-Kutta methods are chosen to agree with the order of the spatial discretization. For the explicit time integration, the time step of FVM is

$$\Delta t = CFL \frac{\bar{\Omega}_i}{\sum_{f \in \partial\Omega_i} \lambda_{c,f} S_f + 2\lambda_{v,f} S_f^2 / \bar{\Omega}_i}, \quad CFL \leq 1$$

where $\lambda_c = |\mathbf{v} \cdot \mathbf{n}_f| + c$ is the spectral radius of the flux Jacobian matrix at the cell face, c is the speed of sound and $c^2 = \gamma RT$, the diffusion coefficient is $\lambda_v = \max(\frac{4}{3}, \frac{\gamma}{\mathrm{Pr}}) \frac{\mu}{\rho}$. The finite volume scheme (2.4) can be rewritten as the following semi-discretization form

$$\bar{\Omega}_i \frac{\mathrm{d}\bar{U}_i}{\mathrm{d}t} = R(U) \tag{2.6}$$

where the term on the right-hand side is

$$R(U) = -\frac{1}{\bar{\Omega}_i} \oint_{\partial\Omega_i} \mathbf{F}_n \cdot \mathbf{n} \, \mathrm{d}S$$

Then the third order explicit TVD Runge-Kutta methods for (2.6) can be written as

$$\begin{aligned}
U^{(0)} &= \bar{U}_i^n, \\
U^{(1)} &= U^{(0)} + \frac{\Delta t}{\bar{\Omega}_i} \cdot R(U^{(0)}), \\
U^{(2)} &= \frac{3}{4} U^{(0)} + \frac{1}{4} \left(U^{(1)} + \frac{\Delta t}{\bar{\Omega}_i} \cdot R(U^{(1)}) \right), \\
U^{(3)} &= \frac{1}{3} U^{(0)} + \frac{2}{3} \left(U^{(2)} + \frac{\Delta t}{\bar{\Omega}_i} \cdot R(U^{(2)}) \right), \\
\bar{U}_i^{n+1} &= U^{(3)}
\end{aligned} \tag{2.7}$$

and the fourth order Runge-Kutta method as

$$\begin{aligned}
U^{(0)} &= \bar{U}_i^n, \\
U^{(1)} &= U^{(0)} + \frac{\Delta t}{2\bar{\Omega}_i} R\left(U^{(0)}\right), \\
U^{(2)} &= U^{(0)} + \frac{\Delta t}{2\bar{\Omega}_i} R\left(U^{(1)}\right), \\
U^{(3)} &= U^{(0)} + \frac{\Delta t}{\bar{\Omega}_i} R\left(U^{(2)}\right), \\
U^{(4)} &= U^{(0)} + \frac{\Delta t}{6\bar{\Omega}_i} \left(R\left(U^{(0)}\right) + 2R\left(U^{(1)}\right) + 2R\left(U^{(2)}\right) + R\left(U^{(3)}\right) \right), \\
\bar{U}_i^{n+1} &= U^{(4)}.
\end{aligned} \tag{2.8}$$

The defect of the explicit time integration is that it is restricted by the CFL condition, which makes the time step small. In simulating the engineering flows, the grids are always refined near the regions of shock wave, boundary layers and strong vortex, etc. In these situations, the global time step is very small and the explicit time integration is not a good choice in efficiency. On the other hand, the implicit time integration can break the limit of CFL condition. The much larger time step, theoretically speaking, infinitely large for fully implicit scheme, can speed up the computation greatly. The following section will show how to construct the LU-SGS implicit dual time-stepping algorithm for the Navier-Stokes equations [1, 2].

Use the semi-discretization scheme in (2.6) and have

$$\frac{\partial \bar{U}_i^n}{\partial t} = R(U_i^{n+1})/\bar{\Omega}_i$$

Introduce the term of virtual time $\partial \bar{U}_i/\partial \tau$ and use the second-order backward finite difference scheme to discretize the physical time step (for convenience, the subscript for cell i is omitted)

$$\frac{\partial U}{\partial \tau} + \frac{(1+\phi)(U^{n+1} - U^n) - \phi(U^n - U^{n-1})}{\Delta t} = R(U^{n+1})/\bar{\Omega}_i$$

where $\phi = 1/2$. For the virtual time, first-order forward finite difference scheme is used

$$\frac{U^{m+1} - U^m}{\Delta \tau} + \frac{(1+\phi)(U^{m+1} - U^n) - \phi(U^n - U^{n-1})}{\Delta t} = R(U^{n+1})/\bar{\Omega}_i$$

where m is the step for inner iteration. And the flux term on the right-hand side is

$$R(U^{n+1}) = -\sum_{j \in N(i)} \widehat{F}_{ij} S_{ij}$$

with $N(i)$ being the face-neighbor cells of cell i. And the flux approximated using the first-order Lax scheme is

$$\hat{F}_{ij} = \frac{1}{2}\left[F_i + F_j - \lambda_{ij}(U_j - U_i)\right], j \in N(i)$$

where λ_{ij} is the spectral radius of the flux Jacobian matrix at the cell face shared by cell i and j,

$$\lambda_{ij} = \lambda_c + \lambda_v S_f^2/\bar{\Omega}_i$$

Thus, the right-hand side term can be linearized as

$$\begin{aligned} R(U^{n+1}) &= R(U^n) + \frac{\partial R}{\partial U}\Delta U \\ &= R(U^n) + \sum_{j \in N(i)} \frac{S_{ij}}{2}\left[\left(J(U_i, \mathbf{n}_{ij}) + \lambda_{ij}\right)\Delta U_i \right. \\ &\quad \left. + \left(J(U_j, \mathbf{n}_{ij}) - \lambda_{ij}\right)\Delta U_j\right]. \end{aligned}$$

Because each control volume is closed, for cell i we have

$$\sum_{j\in N(i)} \frac{S_{ij}}{2} J(U_i, \mathbf{n}_{ij}) \Delta U_i = 0$$

The final equation needed to solve is

$$\left(\bar{\Omega}_i \left(\frac{1}{\Delta\tau} + \frac{(1+\phi)}{\Delta t} \right) + \sum_j \frac{|S_{ji}|}{2} |\lambda_{ji}| \right) \Delta U_i^m + \sum_j \frac{|S_{ji}|}{2} \left(J(U_j, n_{ji}) - |\lambda_{ji}| \right) \Delta U_j^m = R_t$$

with the term on the right-hand side

$$R_t = R(U^n) - \frac{(1+\phi)(U_i^m - U_i^n) - \phi(U_i^n - U_i^{n-1})}{\Delta t} \bar{\Omega}_i$$

To avoid the explicit computation of Jacobian matrix, the following approximation is used

$$J\Delta U \approx \Delta F = F(U + \Delta U) - F(U)$$

This kind of problem can be solved using the Lower-Upper Symmetric Gauss-Seidel (LU-SGS) method described below.

Forward sweep:

$$D_i \Delta U_i^{m,*} + \sum_{j\in L(i)} \frac{|S_{ji}|}{2} \left(J(U_j, n_{ji}) - |\lambda_{ji}| \right) \Delta U_j = R_t$$

Backward sweep:

$$D_i \Delta U_i^m + \sum_{j\in U(i)} \frac{|S_{ji}|}{2} \left(J(U_j, n_{ji}) - |\lambda_{ji}| \right) \Delta U_j = -D_i \Delta U_i^{m,*}$$

where $L(i)$ and $U(i)$ represent the lower and upper face-neighbor elements with respect to cell i, and the diagonal matrix is

$$D_i = \bar{\Omega}_i \left(\frac{1}{\Delta\tau} + \frac{(1+\phi)}{\Delta t} \right) + \sum_j \frac{|S_{ji}|}{2} |\lambda_{ji}|$$

Usually the so-called re-order technique is used to speed up convergence on unstructured grids [2], i.e., rearrange the cell numbering order so as to make the index of face-neighbor cells close the central cell. Since our grids are generated using the commercial software with reasonable numbering order, the re-order step is not used.

2.3 High-Order K-Exact Reconstruction

To calculate the left and right states on the faces, we need to reconstruct the distributions using the cell-average values, which is also the key of FVM to achieve high-order accuracy. One of the pioneering work in this area is the FV scheme based on the k-exact reconstruction, which was first proposed by Barth and Fredrichson [3] and later was furthered improved to other applications by Delanaye and Liu [4], Ollivier-Gooch [5, 6] and others. The k-exact reconstruction is of several important properties, namely conservation of the mean, k-exactness and compact support [5, 6]. In this paper, the second-, third-, and fourth-order schemes will be studied which are corresponding to k = 1, 2, and 3. Then the conservation of the mean property requires

$$\frac{1}{\bar{\Omega}_i}\int_{\Omega_i} u_i(x)\,\mathrm{d}\Omega = \bar{u}_i \tag{2.9}$$

where $\bar{u}_i$ is the exact cell average of $u(\mathbf{x})$. Let the center of gravity for Ω_i be $\mathbf{x}_i = (x_i, y_i, z_i)$. The mean preserving property can be automatically satisfied if the reconstruction polynomial is expressed as:

$$u_i(x) = \bar{u}_i + \sum_{l=1}^{DOF(k)} u_i^l \varphi_{l,i}(x), \quad DOF(k) = \frac{(k+1)\cdots(k+d)}{d!} - 1, d = 3 \tag{2.10}$$

where the zero-mean basis φ_k can be expressed as:

$$\begin{aligned}
\varphi_{1,i}(\mathbf{x}) &= \delta x_i,\ \varphi_{2,i}(\mathbf{x}) = \delta y_i, \varphi_{3,i}(\mathbf{x}) = \delta z_i,\\
\varphi_{4,i}(\mathbf{x}) &= \delta x_i^2 - \overline{\delta x_i^2}\\
\varphi_{5,i}(\mathbf{x}) &= \delta x_i \delta y_i - \overline{\delta x_i \delta y_i}\\
\varphi_{6,i}(\mathbf{x}) &= \delta x_i \delta z_i - \overline{\delta x_i \delta z_i}\\
\varphi_{7,i}(\mathbf{x}) &= \delta y_i^2 - \overline{\delta y_i^2}\\
\varphi_{8,i}(\mathbf{x}) &= \delta y_i \delta z - \overline{\delta y_i \delta z_i}\\
\varphi_{9,i}(\mathbf{x}) &= \delta z_i^2 - \overline{\delta z_i^2}\\
&\cdots\\
\delta x_i &= x - x_i, \delta y_i = y - y_i, \delta z_i = z - z_i,
\end{aligned} \tag{2.11}$$

with the moments defined as

$$\overline{\delta x_i^{m_1} \delta y_i^{m_2} \delta z_i^{m_3}} = \frac{1}{\bar{\Omega}_i}\int_{\Omega_i} (x - x_i)^{m_1} (y - y_i)^{m_2} (z - z_i)^{m_3}\,\mathrm{d}x\mathrm{d}y \tag{2.12}$$

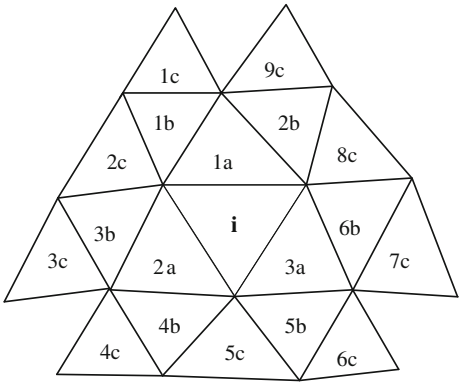

Fig. 2.2 Two-dimensional reconstruction stencils. Second order, $S_i = i, 1a, 2s, 3a$; their order, $S_i = \{i, 1a, 2a, 3a, 1b, 2b, \cdots, 6b\}$; fourth order, $S_i = \{i, 1a, 2a, 3a, 1b, 2b, \cdots, 6b, 1c, 2c, \cdots, 6c\}$. Reprinted from Ref. [7], Copyright 2012, with permission from Elsevier

The property of k-exactness means that $u_i(\mathbf{x})$ reconstructs polynomials of degree k or less exactly, which can be used to determine the coefficients u_i^l. In the neighborhood of Ω_i, there exists a degree k polynomial $P(\mathbf{x})$ that approximates the exact solution $u(\mathbf{x})$ to $O(h^{k+1})$. Because of the k-exactness property of the solution, $u_i(\mathbf{x})$ reconstructs $P(\mathbf{x})$ exactly. Therefore, in the neighborhood of Ω_i represented by a local stencil S_i, we have

$$\frac{1}{\bar{\Omega}_j}\int\limits_{\Omega_j} u_i(\mathbf{x})\,\mathrm{d}\Omega = \frac{1}{\bar{\Omega}_j}\int\limits_{\Omega_j} P(\mathbf{x})\,\mathrm{d}\Omega = \bar{u}_j + O(h^{k+1}), \quad \forall j \in S_i \tag{2.13}$$

where h is the length scale which can be chosen as the maximum radius of the circumcircle of the control volumes, or just be simply defined as $h = \bar{\Omega}^{1/3}$ for non-distorted element. Eq. (2.13) is a system of linear equations of the coefficients u_i^l. To avoid possible singularity, the number of cells in the stencil is required to be larger than the number of unknowns which makes the system over-determined. This system is solved using the method of least squares based on the singular value decomposition.

The left problem is the choosing for the reconstruction stencils S_i. The compact support property implies that the cells in the stencil will not be topologically far from control volume Ω_i. To ensure the systematic and efficient implementation of the k-exact reconstruction, the reconstruction stencil should be as compact as possible. In the present paper, starting from Ω_i, we recursively add the direct face (Von Neumann) neighbors of the cells that are already included in the reconstruction stencil until there is sufficient number of cells in the stencil. Specifically speaking, starting from the central cell as shown in Figs. 2.2 and 2.3, we choose the face-neighbor cells as the first reconstruction stencils, and then take them as the outer border to choose face-neighbor cells as the second reconstruction stencils. The third and fourth reconstruction can be chosen in a similar way.

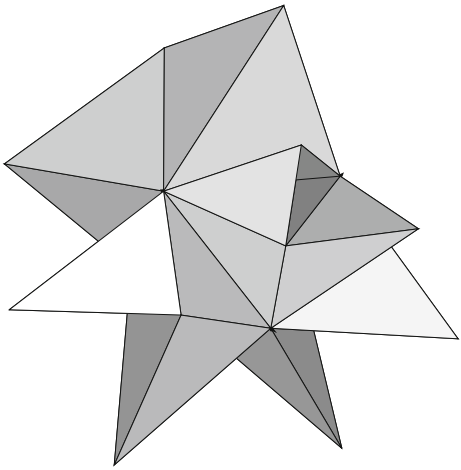

Fig. 2.3 Three-dimensional reconstruction stencils

2.4 Evaluation of Viscous Flux in High-Order Accuracy

In the second-order FVM, we assume the linear distribution for the physical quantities, thus the gradient are constant inside every cell. The evaluation of viscous flux is always first-order accurate, which is not adequate to compute the viscous flow dominated by the boundary layer and vortex structures. In the high-order k-exact reconstruction, the distribution of physical data can be also high order accuracy and brings the possibility to compute the viscous flux in high-order accuracy. Now few references talk about the high-order accuracy viscous flux evaluation. Olliver [5] used the central difference like scheme to compute the viscous flux in the convective-diffusive equation, and Titarev [8] used the same way to compute the viscous flux in Navier-Stokes equations by carrying out the k-exact reconstruction for the terms in viscous flux, such as the gradient of velocity and temperature. In the following sections, we will provide a simple yet an efficient way to compute the viscous flux for the Navier-Stokes equations. This method takes use of the original k-exact reconstruction for the conservative variables to compute the viscous flux directly, and thus reduce the times of reconstruction greatly. The boundary conditions are applied on the k-exact reconstructions through the constrained conditions, which are solved using the constrained least-square methods.

2.4.1 The Viscous Flux Computation

To compute the viscous flux of the face, the gradients, $\partial v_i/\partial x_j$, $\partial T/\partial x_j$, are required to be computed first. Since we only know the reconstructed polynomials of conservative variables, these gradients on the face can be computed by the following two steps.

First, the gradient of conservative variables is computed by simple arithmetic average of the gradients on left and right sides, i.e.,

$$\frac{\partial q}{\partial x_j} = \frac{1}{2}\left(\frac{\partial q_L(\mathbf{x})}{\partial x_j} + \frac{\partial q_R(\mathbf{x})}{\partial x_j}\right) \tag{2.14}$$

where q represents any component of the conservative variables $\bar{U}$, L and R represent left and right cells, respectively. Next, the required gradients are computed by some simple partial operations.

$$\begin{aligned}
&\frac{\partial v_i}{\partial x_j} = \frac{1}{\rho^2}\left(\rho\frac{\partial \rho v_i}{\partial x_j} - \rho v_i\frac{\partial \rho}{\partial x_j}\right),\\
&\frac{\partial p}{\partial x_j} = (\gamma-1)\left(\frac{\partial E}{\partial x_j} - \frac{1}{2}\left(u\frac{\partial \rho u}{\partial x_j} + \rho u\frac{\partial u}{\partial x_j} + v\frac{\partial \rho v}{\partial x_j} + \rho v\frac{\partial v}{\partial x_j} + w\frac{\partial \rho w}{\partial x_j} + \rho w\frac{\partial w}{\partial x_j}\right)\right),\\
&\frac{\partial T}{\partial x_j} = \frac{1}{\rho^2 R}\left(\frac{\rho\partial p}{\partial x_j} - \frac{p\partial \rho}{\partial x_j}\right).
\end{aligned} \tag{2.15}$$

As the required gradients for the velocity and temperature are computed, the viscous flux can be obtained by

$$\oint_{\partial\Omega_i} \mathbf{F}_v\cdot\mathbf{n}\,\mathrm{d}S = \sum_{k=1}^{N_F}\int_{\partial\Omega_{i,k}} \mathbf{F}_v\cdot\mathbf{n}\,\mathrm{d}S \approx \sum_{k=1}^{N_F}\left(\sum_{\phi=1}^{N_G}\omega_g\widehat{F}_v(U(\mathbf{x}_{k,\phi}),\nabla U(\mathbf{x}_{k,\phi}))\cdot\mathbf{n}S_k\right)$$

And the values of conservative variables take the mean values on the face.

2.4.2 On the Implementation of Viscous Boundary Conditions

In this part we enforce the Dirichlet, Neumann-type boundary conditions by constraining the least-squares reconstruction in control volumes adjacent to the boundary. The constrained conditions are the boundary conditions on the Gauss quadrature points.

$$u(\mathbf{x})|_{\mathbf{x}_g} = u_g,\ g = 1,\cdots,N_g$$

where u_s is the boundary condition of variable u on the quadrature points. Substitute the reconstruction polynomial to the above formula and get

$$\bar{u}_i + \sum_{l=1}^{K} u_i^l\varphi_{l,i}(\mathbf{x}_g) = u_g$$

By collecting the constrained condition equations, we have

$$\begin{aligned} \mathbf{C}\mathbf{u}_i &= \mathbf{d} \\ \mathbf{C} &= \left[\varphi_{l,i}(\mathbf{x}_g)\right], \mathbf{d} = \left[u_g - \bar{u}_i\right] \\ l &= 1, \cdots, K, \quad g = 1, \cdots, N_g \end{aligned} \tag{2.16}$$

And in the k-exact reconstruction, we already have

$$\mathbf{A}\mathbf{u}_i = \mathbf{b} \tag{2.17}$$

The problem becomes the over-determined system (2.17) with constrained conditions (2.16), which can be solved using the constrained least-square method introduced below. By introduce the Lagrange multiplier λ, the constrained least-square methods for solving the above equation are to make the following residual minimize

$$f = \sum_{i=1}^{Ns} \left[\sum_{j=1}^{K} a_{ij} x_j - b_i \right]^2 - \sum_{i=1}^{Ng} \lambda_i \left[\sum_{j=1}^{K} c_{ij} x_j - d_i \right]$$

where a_{ij} and c_{ij} are the components of $\mathbf{A}$ and $\mathbf{C}$, b_i, d_i and x_i are the components of vector $\mathbf{b}$, $\mathbf{d}$ and $\mathbf{u_i}$, Ns is the number of reconstruction stencils. Differentiations for the residual f yields

$$\frac{\partial f}{\partial x_i} = 0, \quad \frac{\partial f}{\partial \lambda_j} = 0, \quad i = 1, \cdots, K, \; j = 1, \cdots, N_g$$

and final equations for computed $\mathbf{x}$, λ becomes

$$\begin{bmatrix} 2\mathbf{A}^T\mathbf{A} & -\mathbf{C}^T \\ \mathbf{C} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_i \\ \lambda \end{bmatrix} = \begin{bmatrix} 2\mathbf{A}^T\mathbf{b} \\ \mathbf{d} \end{bmatrix}$$

To save the computational effort, the inverse of left matrix and matrix A are stored. Actually we only need matrix and vector multiplication operator to compute the coefficients $\mathbf{u}_i$.

2.5 Parallelization for Large-Scale Computation

In the practical engineering applications, large-scale grids are required to make the calculation more accurate, which always need to work on several computers simultaneously. In the computation, we need to divide the computational grids into partitions and one process is responsible for one partition. The most challenge is the data exchange in the interfaces of partitions. Comparing to the second-order scheme, the parallelization for the high-order FVM is more difficult since we need to create a

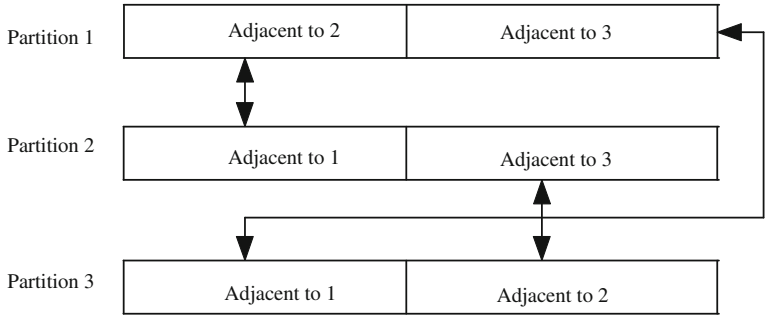

Fig. 2.4 Data transfer model for the parallel program

large set of virtual grids related to the adjacent partitions and transfer much larger data on these regions.

First we use the METIS software [9] to divide the computational grids into several partitions and obtain the partition index for each cell. To exchange data between two adjacent partitions, virtual grids large enough are setup to store the data of neighbor partitions. This is to make the interfacial cells work exactly as the inner cells. According to the algorithms, the data exchange is required on the k-exact reconstruction and flux evaluation. Especially the k-exact reconstruction needs data on large stencils. With the partition information, the interfacial cells can be identified. We create the virtual grids for the reconstruction stencils of the interfacial cells, which are on different partitions from that of the interfacial cells.

With these virtual grids, data exchange must be carried on during the computation. According to the numerical algorithm, the mean values of physical data are exchanged before the k-exact reconstruction and the reconstruction polynomials are exchanged to compute the left and right state used for the numerical flux evaluation. The data exchange is realized using the Message Passing Interface (MPI) standard library. As shown in Fig. 2.4, every partition is adjacent to several other partitions, thus the MPI non-standard message transfer model is used, i.e., every partition firstly sends all data to target partitions, then receive the data from other partitions and set the values on the virtual grids.

2.6 Numerical Tests

This section will solve the two- and three-dimensional Navier-Stokes equations using second- to fourth-order FVMs. The accuracy of high-order FVM will be validated first, then a series of compressible viscous flows are used to verify the correctness of the schemes.

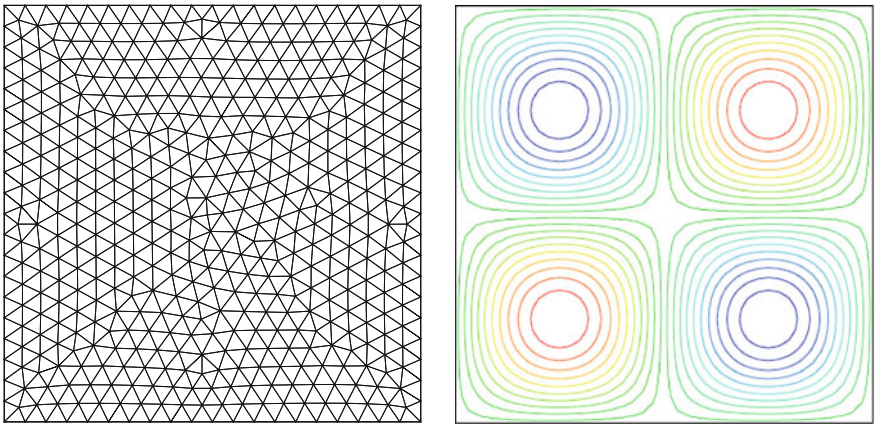

Fig. 2.5 Computational grids (*left*) and initial conditions (*right*) for the two-dimensional convective problem

2.6.1 Accuracy Validation

The following two standard tests are carried out to test the accuracy of the high-order FVM.

2.6.1.1 Two-Dimensional Linear Convective Equation

The two-dimensional linear convective equation can be written as

$$u_t + \mathbf{a} \cdot \nabla u = 0$$

where $\mathbf{a}$ is the constant convective velocity taken as $(1, 2)$ in this test. The convective flux is calculated using the upwind schemes. The initial wave is taken as

$$u_0(x, y) = \sin(2\pi x) \sin(2\pi y)$$

The left picture of Fig. 2.5 shows the computational grids on region $[0, 1] \times [0, 1]$ with grid size $h = 1/20$. The boundaries are periodic. The right picture of Fig. 2.5 shows the computational results of fourth order FVM at $t = 1$ on $h = 1/40$ grids, which return to the initial values. A series of successive refined grids are used to assess the accuracy of the high-order FVM, which are shown in Table 2.2. From this table we can observe that all the second- to fourth-order schemes can achieve the expected orders.

Table 2.2 Accuracy test for the linear convective equation

Scheme	Grid	L1 error	Order	L error	Order
2nd order	1/20	1.14E-02		4.63E-02	
	1/40	2.56E-03	2.16	1.16E-02	2
	1/80	6.00E-04	2.09	2.80E-03	2.05
	1/160	1.39E-04	2.11	8.07E-04	1.8
	1/320	3.55E-05	1.97	2.62E-04	1.62
3rd order	1/20	2.07E-02		3.73E-02	
	1/40	2.69E-03	2.95	4.81E-03	2.95
	1/80	3.37E-04	2.99	6.01E-04	3
	1/160	4.22E-05	3	7.50E-05	3
	1/320	5.27E-06	3	9.37E-06	3
4th order	1/20	1.26E-03		2.67E-03	
	1/40	7.51E-05	4.06	1.56E-04	4.1
	1/80	4.64E-06	4.02	9.33E-06	4.06
	1/160	2.89E-07	4	5.71E-07	4.03
	1/320	1.82E-08	3.99	3.57E-08	4

2.6.1.2 The Constructed Solutions for the Three-Dimensional Euler and Navier-Stokes Equations

These two solutions are used to test the accuracy of the three-dimensional high-order FVM. These are unsteady cases. Give an unsteady vector solutions $U_p(\mathbf{x}, t)$ and substitute them into the right-hand side of the equations,

$$\frac{\partial U}{\partial t} + \frac{\partial (F - F_v)}{\partial x} + \frac{\partial (G - G_v)}{\partial y} + \frac{\partial (H - H_v)}{\partial z} = S.$$

Next we will define two sets of initial flow fields and the corresponding source terms, for which the governing equations are the Euler and Navier-Stokes equations. The computational grids are shown in Fig. 2.6.

For the inviscid Euler equations, the exact solution and the corresponding source terms are defined as

$$U_p(\mathbf{x}, t) = \begin{pmatrix} \rho \\ \mathbf{V} \\ p \end{pmatrix} = \begin{pmatrix} 2 + A_0 \sin(\omega t - \mathbf{kx}) \\ \mathbf{0} \\ 2 + A_0 \sin(\omega t - \mathbf{kx}) \end{pmatrix}$$

and

$$\mathbf{S} = \begin{pmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \end{pmatrix} (\mathbf{x}, \mathbf{t}) = \begin{pmatrix} \omega A_0 \cos(\omega t - \mathbf{kx}) \\ -k_x A_0 \cos(\omega t - \mathbf{kx}) \\ -k_y A_0 \cos(\omega t - \mathbf{kx}) \\ -k_z A_0 \cos(\omega t - \mathbf{kx}) \\ \frac{1}{\gamma - 1} \omega A_0 \cos(\omega t - \mathbf{kx}) \end{pmatrix}$$

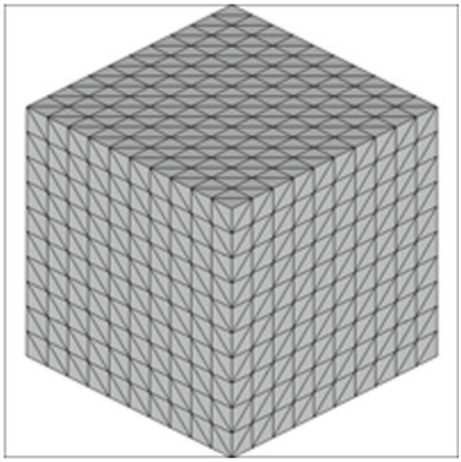

Fig. 2.6 The three-dimensional computational grids for accuracy test

Table 2.3 The accuracy test for the three-dimensional Euler equations

Scheme	Grid	L1 error	Order	L error	Order
3rd order	1/10	3.28E-03		9.84E-03	
	1/20	4.10E-04	3	1.26E-03	2.96
	1/30	1.22E-04	3	3.71E-04	3.02
	1/40	5.13E-05	3	1.56E-04	3
4th order	1/10	2.15E-04		4.07E-04	
	1/20	4.27E-05	3.99	8.34E-05	3.91
	1/30	1.35E-05	4.01	2.65E-05	3.99
	1/40	5.51E-06	4.01	1.08E-05	4.01

where the constant $A_0 = 1$ and $\mathbf{k} = k_x, k_y, k_z, k_x = k_y = k_z = \omega = 2\pi$. The computational time is $t = 0.25$. The 3rd and 4th order FV schemes are used and the error using L_1 and L_∞ are shown in Table 2.3, where we can find the convergence speeds of errors can achieve the expected ones.

For the viscous flows, the exact solution are defined as

$$U_p(\mathbf{x}, t) = \begin{pmatrix} \rho \\ \mathbf{v} \\ p \end{pmatrix} = \begin{pmatrix} 1 \\ \mathbf{0} \\ 2 + A_0 \sin(\omega t - \mathbf{kx}) \end{pmatrix}$$

with temperature

$$T = p/\rho = 2 + A_0 \sin(\omega t - \mathbf{kx})$$

where the constant $A_0 = 1$ and $\mathbf{k} = k_x, k_y, k_z, k_x = k_y = k_z = \omega = 2\pi$. The computational time is $t = 0.25$. The viscous viscosity is taken as $\mu = 1$ and Prantle number as 1. The source terms are

Table 2.4 The accuracy test for the three-dimensional Navier-Stokes equations

Scheme	Grid	L_1 error	Order	L_∞ error	Order
3rd order	1/10	3.68E-03		7.12E-03	
	1/20	7.00E-04	2.4	9.96E-04	2.84
	1/30	2.87E-04	2.2	4.30E-04	2.07
	1/40	1.57E-04	2.11	2.39E-04	2.05
4th order	1/10	4.68E-04		8.05E-04	
	1/20	9.92E-05	3.83	1.79E-04	3.71
	1/30	3.03E-05	5.39	5.57E-05	5.23
	1/40	1.18E-05	4.2	2.25E-05	4.07

$$S = \begin{bmatrix} 0 \\ -k_x A_0 \cos(\omega t - \mathbf{kx}) \\ -k_y A_0 \cos(\omega t - \mathbf{kx}) \\ -k_z A_0 \cos(\omega t - \mathbf{kx}) \\ \frac{1}{\gamma - 1}\omega A_0 \cos(\omega t - \mathbf{kx}) + k_{\text{conduct}} A_0 \sin(\omega t - \mathbf{kx}) \left(k_x^2 + k_y^2 + k_z^2\right) \end{bmatrix}$$

The 3rd and 4th order FV schemes are used and the error using L_1 and L_∞ are shown in Table 2.4, where we can find the convergence speeds of errors can achieve the expected ones.

2.6.2 The Flat Plate Boundary Layer

We consider the laminar boundary over an adiabatic flat plate at a free-stream Mach number of 0.3 and a Reynolds number of 15,000 based on the free-stream velocity and the length of the flat plate. The computation domain is $[-0.03, 0.1] \times [0., 0.03]$ with 24,700 element, as shown in Fig. 2.7. The flat plate lies from $x = 0$ to 0.1.

The second- to fourth-order FV schemes are used in the computation. The $u\&v$ nondimensional velocities in the normal directions are shown in Fig. 2.8, where we can observe that the second- to fourth-order results can agree with the exact Blasius solutions quite well. This test case validates the correctness of using the mean values of conservative variables to compute the viscous flux and the constrained least-square method to treat the viscous boundary conditions.

2.6.3 The Subsonic Flow Around NACA0012 Airfoil

This test case deals with a subsonic flow past a NACA0012 airfoil at a Mach number of 0.5, and an angle of attack 0^o, and a Reynolds number of 5,000 based on the

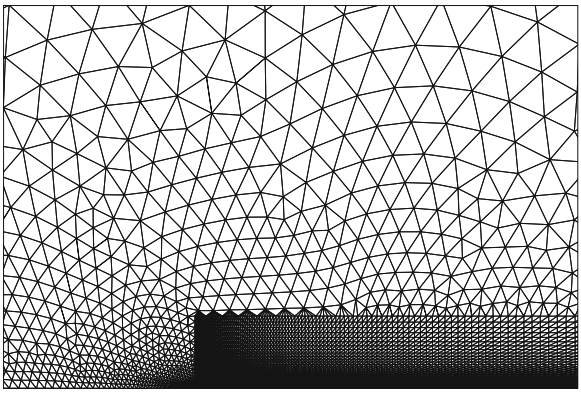

Fig. 2.7 Regions and grids for the boundary layer plate

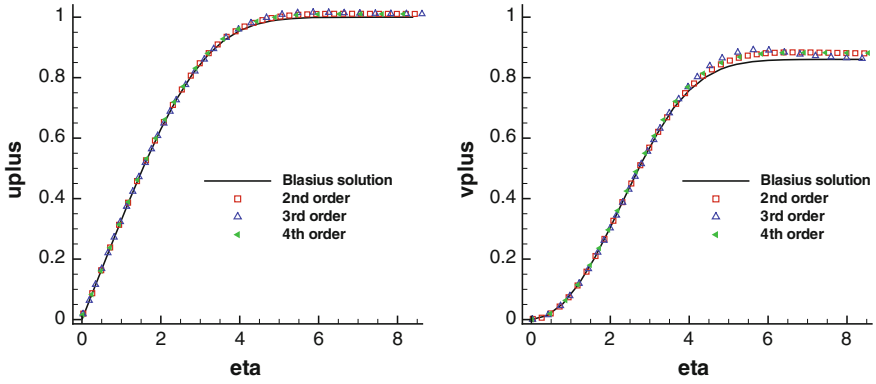

Fig. 2.8 The u and v velocity distribution in the normal directions

free-stream velocity and the chord length of the airfoil. An adiabatic wall is assumed in this case. The computational grids are shown in Fig. 2.9.

We use a well-tested second-order scheme to compute the pressure and skin friction coefficient on a fine grids and take the results as the reference values. The Mach number contour is shown in the right side of Fig. 2.9. The 3rd and 4th order schemes are used to compute the pressure and skin friction distributions. The results are shown in Fig. 2.10, where the coefficients can match the reference values quite well.

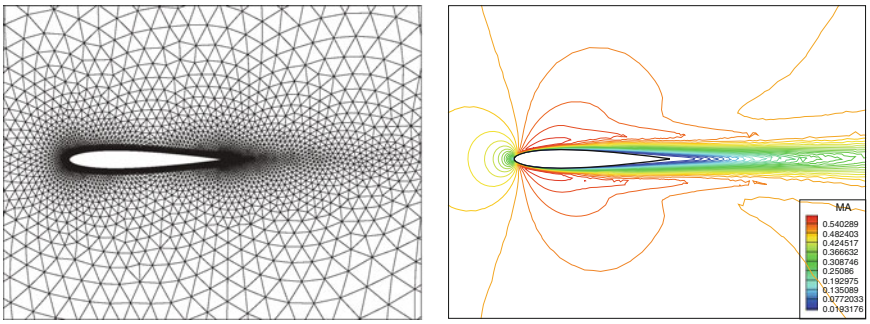

Fig. 2.9 Grid and mach number contour for NACA0012

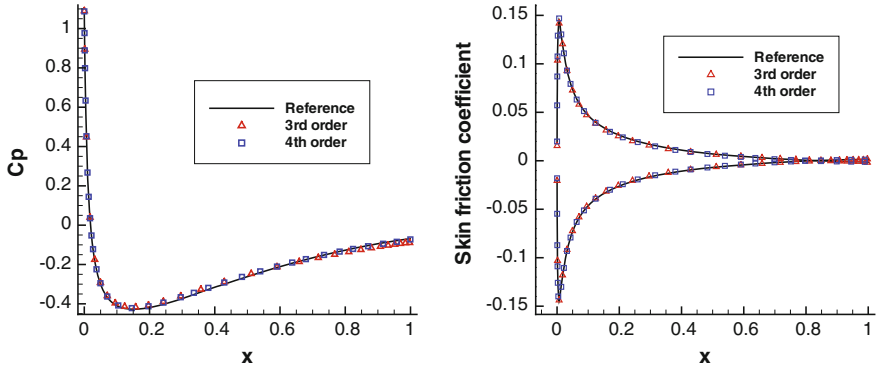

Fig. 2.10 Grid and Mach number contour for NACA0012

2.6.4 Sound Field Generated by the Flow Around a Circular Cylinder

This test case concerns about the sound field generated by the unsteady Kármon street vortex. We have the cylinder diameter $d = 1$, the Reynolds number based on the cylinder diameter and free stream parameters $\mathrm{Re} = \rho_\infty u_\infty d/\mu_\infty = 150$ and the inlet Mach number $M_\infty = 0.2$. Since the pressure fluctuation computation requires high accuracy numerical schemes, the 4th order FV scheme is used in the computation where the curved boundary is approximated using 3rd order polynomials for each segment. The grids are shown in the left of Fig. 2.11 with 83,936 triangular grids. The sound field caused by the Kármon street vortex is shown in the right of Fig. 2.11, which agrees with the results of Müller [10] and Dumbser [11] quite well. And the Mach contour in the vortex is shown in Fig. 2.12. The historical lift and drag coefficients are shown in Fig. 2.13 and the corresponding Strouhal number can be computed $St = fd/\mu_\infty = 0.182$, which agrees with the results of Müller [10] and Dumbser [11] $St = 0.182$ quite well.

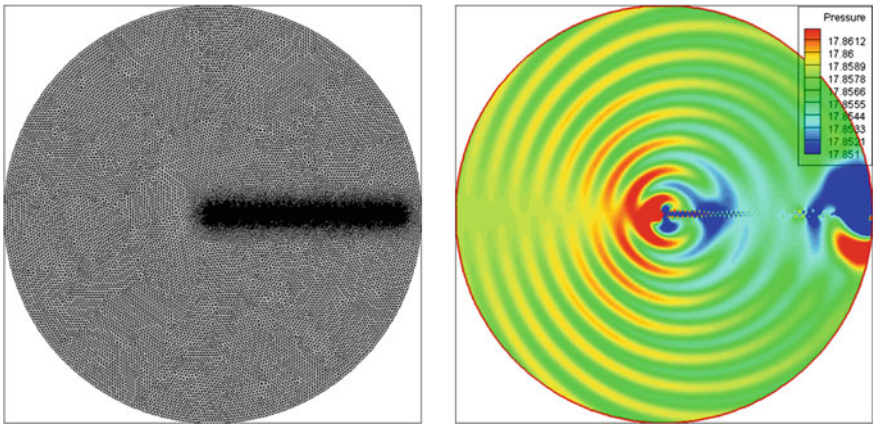

Fig. 2.11 Grid and Mach number contour for NACA0012

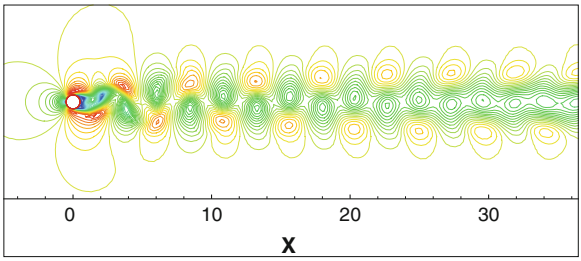

Fig. 2.12 The Mach contour line in the flow around cylinder

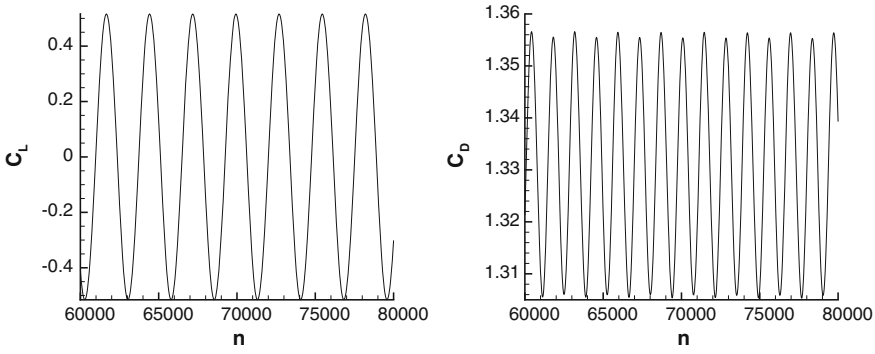

Fig. 2.13 Grid and Mach number contour for NACA0012

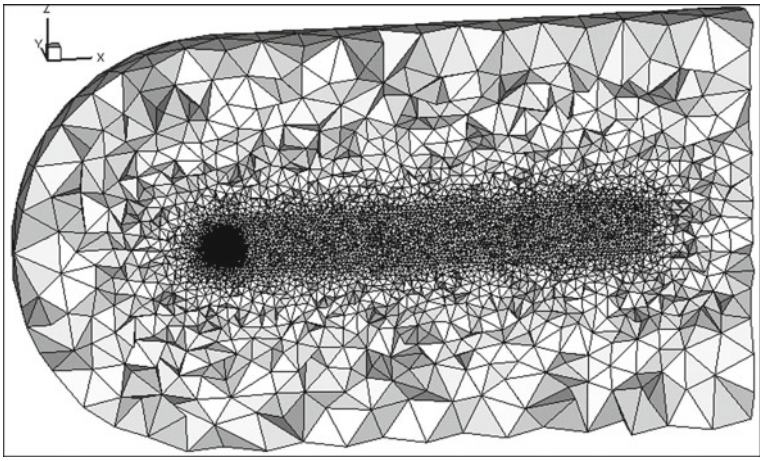

Fig. 2.14 The grids for computing viscous flows around sphere

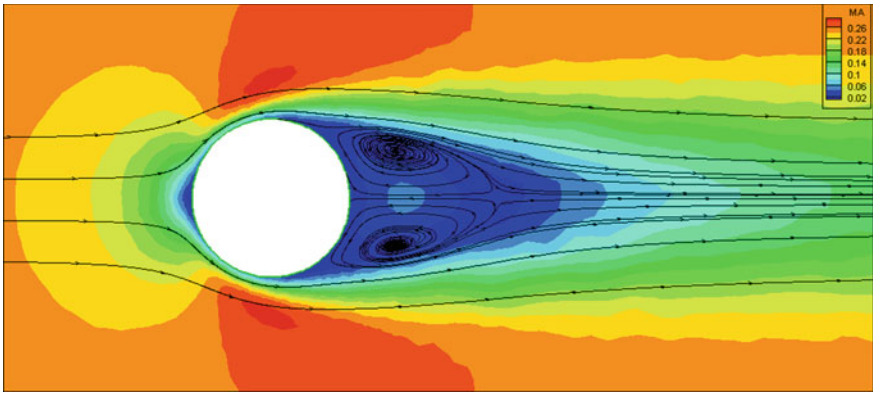

Fig. 2.15 The Mach number contour with stream lines at $z = 0$ plane computed using 4th order FVM

2.6.5 The Steady Viscous Flow Around the Sphere

To validate the correctness of high-order FVM in solving the three-dimensional compressible flows, this section is to compute the steady viscous flow around the sphere. The experimental and computational conditions are taken from reference [12, 13] for comparison. The Reynolds number based on the spherical diameter is 118 with the inlet Mach number 0.2535. The spherical diameter is 1 and the computational region is combined by a half sphere of diameter 10 and a cylinder of height 25. The computational grids are shown in Fig. 2.14 with 458,915 tetrahedrons and 77,844 grid points. The mesh size near the sphere is taken as $h = 0.04$. The 3rd and 4th

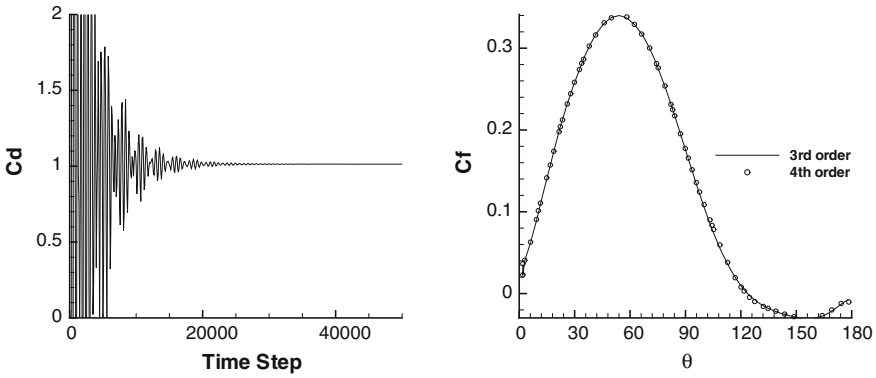

Fig. 2.16 The historical drag coefficients (*left*) and the skin friction coefficients (*right*, θ is angle)

order FV schemes are used in the computations. The Mach number and stream lines can be clearly observed in the wake of the sphere shown in Fig. 2.15. And Fig. 2.16 shows the converged drag coefficient $C_d = 1.01$, which agrees with the results 1.0 in Refs. [12, 13] quite well. Also the order-convergence skin friction coefficients using the 3rd and 4th order FV schemes agree with the Ref. [12] well.

2.7 Conclusions

This chapter completes the framework of high-order finite volume schemes in solving compressible Navier-Stokes equations. Based on the k-exact reconstruction, the high-order quadrature on the faces is carried out using the Gaussian quadrature rules and the FVM is generalized to high-order accuracy. To compute the diffusive flux in high-order accuracy, the key element is the evaluation of the gradients on the face, which is chosen to be the mean value of gradient of reconstruction polynomials on left and right cells. This procedure is very efficient and does not bring in additional reconstruction steps. Also we consider the viscous boundary condition by incorporating it into the k-exact reconstruction step through the constrained least-square method. Two and three-dimensional numerical results show that the high-order scheme can predict the viscous flow accurately.

References

1. Jameson A, Yoon S (1987) Lower-upper Implicit schemes with multiple grids for the Euler equations. AIAA J 25:929–935
2. Sharov D, Nakahashi K (1997) Reordering of 3D hybrid unstructured grids for vectorized LU-SGS Navier-Stokes computations. AIAA Paper 97–2102

3. Barth TJ, Frederichson PO (1990) Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. AIAA 90–0013
4. Delanaye M, Liu Y (1999) Quadratic reconstruction finite volume schemes on 3D arbitrary unstructured polyhedral grids. AIAA 99–3259
5. Ollivier-Gooch C, Altena MV (2002) A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation. J Comput Phys 181:729–752
6. Ollivier-Gooch CF (1997) Quasi-ENO schemes for unstructured meshes based on unlimited data-dependent least-square reconstruction. J Comput Phys 133:6–17
7. Li WA, Ren YX (2012) The multi-dimensional limiters for solving hyperbolic conservation laws on unstructured grids II: extension to high order finite volume schemes. J Comput Phys 231(11):4053–4077
8. Titarev VA, Drikakis D (2011) Uniformly high-order schemes on arbitrary unstructured meshes. Comput Fluids 46(1):467–471
9. Karypis G, Kumar V (1998) Multilevel k-way partitioning scheme for irregular graphs. J Parallel Distrib Comput 48:96129
10. Müller B (2008) High order numerical simulation of aeolian tones. Comput Fluids 37:450–462
11. Dumbser M (2011) Arbitrary high order PNPM schemes on unstructured meshes for the compressible Navier-Stokes equations. Comput Fluids 39:60–76
12. Haga T, Gao H, Wang ZJ (2010) A high-order unifying discontinuous formulation for 3D mixed grids. AIAA 2010–540
13. Taneda S (1956) Experimental investigations of the wake behind a sphere at low Reynolds numbers. J Phys Soc Jpn 11:1104–1108

http://www.springer.com/978-3-662-43431-4

Efficient Implementation of High-Order Accurate Numerical Methods on Unstructured Grids

Li, W.

2014, XV, 148 p. 100 illus., 11 illus. in color., Hardcover
ISBN: 978-3-662-43431-4