General Mesh Method: A Unified Numerical Scheme

Yuewen Jiang

Department of Engineering Science, University of Oxford, England, UK

email: yuewen.jiang@eng.ox.ac.uk

Abstract

An alternative to three existing types of mesh (structured grid, unstructured grid, and gridfree points) — general mesh — is proposed for Computational Fluid Dynamics (CFD). It includes three existing ones and the arbitrary mixing of mesh and points by using a unified framework. Different from the traditional hybrid grid/gridfree methods requiring separate numerical schemes for the respective grid and points zones, a unified numerical scheme — general volume method — is developed for the general mesh to discretize Navier-Stokes (NS) equations. This innovative scheme naturally solves not only all three existing types of mesh but also the arbitrary mixing of mesh and points without requiring interface treatment. Compared to the existing grid and gridfree based computational methods, the general mesh method is endowed by nature with both the geometric flexibility of points (without geometric connectivity) and physical accuracy of mesh (with node connectivity): 1. the meshing is as general, automated, efficient, and flexible as gridfree points but the numerical solution becomes more stable and accurate than the gridfree method; 2. the numerical scheme is as unified, compact, accurate, and robust as the unstructured grid, whereas the meshing becomes incredibly easier than the latter. Meanwhile, the proposed method is comprehensively validated by both low-speed and transonic high-Reynolds number flows. Furthermore, the simulation of HIRENASD demonstrates the potential to improve the convergence by converting the low-quality mesh elements into meshfree points.

Keywords: general mesh method, general volume method, unified mesh/meshfree points, unstructured grid, Computational Fluid Dynamics (CFD), Navier-Stokes (NS) equations

Highlights:

- An alternative to three existing types of mesh — general mesh — is proposed;
- The general volume method is developed for the general mesh;
- It naturally solves the arbitrary mixing of mesh and points without interface;
- It is as unified and compact as the finite volume scheme of unstructured grid method;

1. Introduction

Experiments that are physically close to the "real" flow that researchers expect. The accuracy depends on the experimental facility. Computational Fluid Dynamics (CFD) is a numerical approximation of real physics in computers. Its accuracy depends on the mesh size and quality and numerical methods. It frequently utilizes the Navier-Stokes (NS) equations as the basic governing equations of Newtonian fluids. These partial differential equations (PDEs) are so complicated that the numerical simulations are generally employed to obtain the solutions. Thanks to the efforts of mathematicians and CFD researchers as well as the improvement of computer science, the CFD solutions have become more and more accurate and efficient. However, despite the remarkable success, it is still challenging to apply the CFD method to more and more complicated geometries.

1.1. Traditional Numerical Methods

At the moment, there exist only three types of mesh we frequently utilize in simulations. These are structured grid, unstructured grid, and gridfree points. The former two types belong to the geometric grid requiring topology connectivity, whereas the gridfree method abandons this constraint. Some researchers may think the gridfree points do not belong to the scope of mesh. From a mathematical point of view, it is considered as a special type of mesh in the framework of the general mesh. The respective numerical methods are briefly discussed below.

1.1.1. Grid-based Methods

The popular spatial discretization methods based on a grid can be essentially classified into three categories: finite difference method, finite element method and finite volume method. The finite difference method is the oldest numerical method which has a simple algorithm and high computational efficiency. It is widely used in flow simulations [1–4]. The finite difference method is mainly applied to structured grid. Although there are similar numerical methods on unstructured grid, the algorithms are complex and the stability is poor.

Different from the finite difference method, the finite element method can easily handle the unstructured grid. One of its most successful applications is computational solid dynamics (CSD). However, the application in CFD is not straightforward, because the discontinuousness of flow field will cause the challenge of stability. Zienkiewicz [5] introduced the finite element method for CFD. Besides, it must be mentioned that the discontinuous Galerkin (DG) methods [6–9] have greatly promoted the development of the finite element method for CFD applications.

The finite volume method [10, 11] is remarkably successful because of the excellent balance between accuracy and robustness. It is the most popular method in CFD applications. Most of the well-known CFD codes are based on this method, such as NASA's CFL3D and FUN3D, Boeing's BCFD, DLR's TAU, Rolls Royce's HYDRA, Stanford's SU^2, OpenFOAM, and Fluent. The discretization method of the finite volume method is as flexible as that of the finite difference method of which the flow solution reconstruction is straightforward. The meshing is as easy as that of the finite element method because it is suitable for both structured and unstructured grids. Moreover, based on the integral form of NS equations, the discretized equations can well preserve the conservation of the original PDEs. Therefore, the unstructured grid based finite volume methods are still fast developing [12–16].

However, despite the considerable success of grid-based methods, the grid requiring node connectivity is challenging to generate the high-aspect-ratio prism in the boundary layer for the complicated geometries or those with large moving boundaries. Grid generation is also time-consuming if it requires repeated adjustment of meshing parameters in complicated configurations. Moreover, it can not inherently avoid the low-quality and negative-volume elements that result in stability and convergence problems.

1.1.2. Gridfree Methods

Different from grid decomposition, the gridfree method is not constrained by the grid topology. Thus it is flexible to generate the discrete points for complex geometries and moving boundaries. Zhang [17] presented a gridfree method based on the least-squares method for the two-dimensional incompressible flow. It used the Gauss quadrature in the background cells. The boundary condition was enforced by the penalty method. A 'local' radial basis function (RBF) was used on a gridfree scheme [18] for the unsteady incompressible flow. The velocity and pressure were decoupled. The incompressible flow field is continuous in space which is convenient for flow solution construction. Katz and Jameson [19] developed an edge-based gridfree method for compressible inviscid flow simulations. It used the grid-based reconstruction, diffusion, and convergence acceleration schemes in the gridfree method. Wang and Periaux [20] proposed a gridfree method which used the artificial dissipation to solve the 2D Euler equation. The explicit five-stage Runge-Kutta scheme was employed in time marching.

A gridfree method for viscous flow simulation was proposed by Munikrishna and Balakrishnan [21]. It projected the gridfree solver as a viable Cartesian grid methodology. It was used to calculate the viscous flow around 2D multi-element airfoil. Namvar and Jahangirian [22] investigated the gridfree method for 2D compressible turbulent flows. The spatial derivatives were evaluated by the Taylor least squares method. Dehghan and Abbaszadeh [23] developed a fast and efficient local gridfress method to solve shallow water equations. The proper orthogonal decomposition approach was utilized to construct the gridfree method. The results showed excellent efficiency. Jiang [24, 25] proposed an algebraic-volume gridfree method which was straightforward to implement in the traditional finite volume solver.

Although points generation of gridfree method is much more flexible than grid-based methods, the corresponding numerical methods are not comparable to those of grid-based methods such as robustness and convergence of the simulation and accuracy of the solution. Few studies have applied the gridfree method to complicated configurations so far.

1.1.3. Hybrid Grid/Gridfree Methods

In order to further improve the flexibility & efficiency of unstructured grid generation and the numerical accuracy & robustness of gridfree method, the hybrid grid/gridfree method becomes a natural choice to avoid their respective weaknesses. The hybrid Cartesian grid and gridfree method for com-

pressible flows was proposed in [26]. A Cartesian grid was used as the baseline mesh to cover the computational domain, whereas the boundary surfaces were addressed using a gridfree method. The standard cell-centred finite volume formulation was utilized for the Cartesian grid while the dual least-squares approximation for the gridfree points. Fries and Matthies [27, 28] developed a coupled grid/gridfree method for the incompressible Navier-Stokes equations. A gridfree Galerkin method was established for the gridfree zone and the finite element method for the grid zone. Different shape functions were applied for the corresponding methods. In order to simulate the high-Reynolds number flow, Wang [29] proposed a hybrid gridfree and finite volume method. It employed the finite volume method for the grid zone in the boundary layer and the gridfree method for the points zone in the off-wall region.

A hybrid Cartesian grid and gridfree method [30] was developed for turbulent flow calculations. Cartesian grid dominated the fluid domain, whereas the gridfree method was applied near the wall. It combined an off-wall Cartesian grid solver and a near-wall gridfree solver. A coupled grid/gridfree scheme for boundary vibrations was proposed in [31]. The domain comprised of near-wall gridfree points moving with the body and off-wall stationary Cartesian grid. The arbitrary Lagrangian–Eulerian flow equations were solved by a local RBF in finite difference mode on the moving gridfree points. The flow equations in Eulerian formulation were solved by the traditional finite differencing on the stationary Cartesian grid. Jiang [25] presented a hybrid grid/gridfree method to solve store separation. The gridfree points zone solved by the algebraic-volume gridfree method was coupled with two grid zones solved by the cell-centred finite volume method.

Although the approach of hybrid grid/gridfree methods can achieve some flexibility on the computational space subdivision, it causes the difficulty of numerical discretization and code implementation. This kind of hybrid methodology requires different numerical methods to deal with the relevant zones and needs special treatment for the interface between grid and gridfree zones. Compared to a single grid or gridfree method, this hybrid approach tremendously increases the difficulty and efforts to develop and maintain the CFD code that usually has a large scale.

1.2. General Mesh Method

Therefore, a new type of mesh — general mesh — is proposed to maximize the respective advantages of grid-based method (physical accuracy of

numerical method) and gridfree method (geometric flexibility of mesh generation) by an innovative strategy on both meshing and numerical aspects. It includes not only all three existing types of mesh (structured grid, unstructured grid, and gridfree points) but also the arbitrary mixing of mesh and points by using a unified framework.

Similar to the revolution from structured grid to unstructured one, general mesh requires a breakthrough of numerical method to solve all these much different types of mesh by using a unified scheme. Otherwise, the consequential numerical method becomes considerably complex like the hybrid grid/gridfree approaches and difficult to implement and maintain. Thus, based on the widely-used finite volume scheme [10, 11] and previous work of algebraic-volume gridfree method [24, 25], a general volume method is developed for general mesh.

2. Definition of General Mesh

Figure 1 (a) shows a 2D unstructured mesh. The nodes are connected to form elements. Each connectivity represents an edge that is depicted as a solid line. The topology connectivity is of great benefit to provide the geometric information for defining mathematical operators like the derivative. These operators are then applied to construct the numerical discretization of PDEs. For the vertex-based finite volume scheme, a control volume is constructed by the surrounding elements. In the figure, control volume m is geometrically closed by $N(m)$ faces (edges in 2D) that is depicted by solid blue lines. The face area vector $\Delta\mathbf{S}_{m,p}$ is shown in the blue solid arrow. Let node n represent the p-th neighbour of node m. Meanwhile, node m also contains $N(m)$ neighbours, namely, n, n_1, n_2, n_3, n_4, and n_5.

Figure 1 (b) demonstrates an example of a point cloud. The virtual connectivity or virtual edge is shown as a dashed line. It is found that the vertex-based data structure of unstructured grid is suitable for point cloud as well. A point cloud is a set of $N(m)$ contributors employed to reconstruct the flow solutions. As shown in the figure, m is the current meshfree point, and its contributors or neighbours are points n, n_1, n_2, n_3, n_4, and n_5. According to the algebraic-volume meshfree method [24, 25], an algebraic volume is constructed by the point cloud. It is formed by $N(m)$ algebraic faces shown in the dashed line. The p-th algebraic face has an algebraic area vector $\Delta\tilde{\mathbf{S}}_{m,p}$ illustrated in the dashed arrow.

The corresponding general mesh is depicted in Figure 1 (c). The meshing strategy published separately is that the mesh is adopted to decompose the entire computational domain, whereas the points are proposed to solve the negative-volume, low-quality, and invalid elements. Thus it is the arbitrary mixing of mesh and meshfree points, namely, a mesh nodes + $(1-a)$ meshfree points with $a \in [0, 1]$ representing the proportion. Both $a = 1$ and $a = 0$ are allowed. The former represents a pure mesh with full node connectivity, whereas the latter stands for a pure meshfree points without any connectivity. In other words, general mesh inherently includes mesh, points and arbitrary mixing of both.

Figure 1 (c) exhibits two general volumes of which control volume and algebraic volume naturally solve two types of mesh even at the interface, respectively. It is found that the interface does not require any extra treatment. The general mesh can use a unified data structure to solve both mesh and meshfree points. Namely, it is interface-free in the unified framework of general mesh such that meshfree points can appear anywhere like the words *arbitrary mixing* without any extra treatment.

On the contrary, the widely-used hybrid grid/gridfree methods utilize the individual grid zone and meshfree zone. There exists an interface between zones and it requires special treatment to exchange the information and maintain the accuracy and stability. Thus it is impossible to arbitrarily mix grid and gridfree points in this hybrid approach.

3. General Volume Method

3.1. Finite Volume Scheme for Mesh Nodes

The paper will focus on a classic computational physics problem — numerical solution of Navier-Stokes (NS) equations which are of great interest in both scientific and engineering study. They are typical advection-diffusion equations which describe the motion of Newtonian fluid. The integral form of NS equations can be written as follows:

$$\frac{\partial}{\partial t}\iiint_{\Omega}\mathbf{Q}dV + \iint_{\partial\Omega}\mathbf{F}(\mathbf{Q})\cdot\mathbf{n}dS = \iint_{\partial\Omega}\mathbf{F}^{vis}(\mathbf{Q})\cdot\mathbf{n}dS \tag{1}$$

where t is time. Ω and $\partial\Omega$ are the space and boundary of a discrete control volume respectively. $\mathbf{n} = (n_x, n_y, n_z)^T$ stands for the unit outward-normal vector of $\partial\Omega$. V is the volume (area in 2D) of Ω. S denotes the area (length

in 2D) of $\partial\Omega$. The conservative flow variable vector $\mathbf{Q}$, convective fluxes $\mathbf{F}(\mathbf{Q})$, and viscous fluxes $\mathbf{F}^{vis}(\mathbf{Q})$ are respectively given by

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e_0 \end{bmatrix}, \mathbf{F}(\mathbf{Q})\cdot\mathbf{n} = \begin{bmatrix} \rho\mathbf{v}\cdot\mathbf{n} \\ \rho u\mathbf{v}\cdot\mathbf{n} + Pn_x \\ \rho v\mathbf{v}\cdot\mathbf{n} + Pn_y \\ \rho w\mathbf{v}\cdot\mathbf{n} + Pn_z \\ (\rho e_0 + P)\mathbf{v}\cdot\mathbf{n} \end{bmatrix}, \mathbf{F}^{vis}(\mathbf{Q})\cdot\mathbf{n} = \begin{bmatrix} 0 \\ \tau_{xx}n_x + \tau_{xy}n_y + \tau_{xz}n_z \\ \tau_{yx}n_x + \tau_{yy}n_y + \tau_{yz}n_z \\ \tau_{zx}n_x + \tau_{zy}n_y + \tau_{zz}n_z \\ \sigma_x n_x + \sigma_y n_y + \sigma_z n_z \end{bmatrix} \tag{2}$$

where ρ, P, and e_0 denote the density, static pressure, and specific total energy per unit mass respectively. $\mathbf{v}$ stands for the velocity vector of which u, v, and w are the components. The viscous stress tensor reads:

$$\begin{cases} \tau_{xx} = 2(\mu+\mu_t)\frac{\partial u}{\partial x} - \frac{2}{3}(\mu+\mu_t)\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \tau_{yy} = 2(\mu+\mu_t)\frac{\partial v}{\partial y} - \frac{2}{3}(\mu+\mu_t)\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \tau_{zz} = 2(\mu+\mu_t)\frac{\partial w}{\partial z} - \frac{2}{3}(\mu+\mu_t)\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \tau_{xy} = \tau_{yx} = (\mu+\mu_t)\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \\ \tau_{xz} = \tau_{zx} = (\mu+\mu_t)\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ \tau_{yz} = \tau_{zy} = (\mu+\mu_t)\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \end{cases} \tag{3}$$

And σ_x, σ_y, and σ_z are respectively given by

$$\begin{cases} \sigma_x = u\tau_{xx} + v\tau_{xy} + w\tau_{xz} + \kappa\frac{\partial T}{\partial x} \\ \sigma_y = u\tau_{yx} + v\tau_{yy} + w\tau_{yz} + \kappa\frac{\partial T}{\partial y} \\ \sigma_z = u\tau_{zx} + v\tau_{zy} + w\tau_{zz} + \kappa\frac{\partial T}{\partial z} \end{cases} \tag{4}$$

with κ representing the thermal conductivity coefficient

$$\kappa = \frac{\gamma R}{\gamma - 1}\left(\frac{\mu}{P_r} + \frac{\mu_t}{P_{rt}}\right) \tag{5}$$

where P_r and P_{rt} are the laminar and turbulent Prandtl numbers and their values are 0.72 and 0.9, respectively. T stands for the static temperature. γ is the ratio of specific heat coefficient. R represents the specific gas constant. μ and μ_t denote the molecular viscosity and turbulent viscosity, respectively. The former is calculated by Sutherland's law, whereas the latter is solved by

an extra turbulent model, such as Spalart-Allmaras (SA) model [32], $k-\omega$ SST model [33], etc.

A key feature of the finite volume scheme is the concept of control volume which is a geometric cell, such as line segments in 1D; triangle, quadrilateral, and polygon in 2D; tetrahedron, prism, pyramid, hexahedron, and polyhedron in 3D. The numerical schemes are designed and implemented over the closed faces of control volume to evaluate the integrals of the convective and viscous fluxes of NS equations. The control volume is created via the geometric information of mesh elements. It is identical with the mesh element in the cell-centred finite volume scheme, whereas the median-dual scheme is frequently employed in the vertex-based finite volume scheme. Figure 1 (a) illustrates a 2D example of the median-dual vertex-based control volume which is closed by the solid blue lines. It is formed by connecting the centroids, face- and edge-midpoints of all elements sharing the particular node. Moreover, the edge-based data structure is an efficient choice for the vertex-based finite volume scheme.

For a polygonal control volume m depicted in Figure 1 (a) (polyhedron in 3D), the surface area integral of convective fluxes is discretized by

$$\iint_{\partial\Omega} \mathbf{F}(\mathbf{Q}) \cdot \mathbf{n} dS = \sum_{p \in \Xi(m)} \mathbf{F}(\mathbf{Q}_{m,p}) \cdot \Delta \mathbf{S}_{m,p} \tag{6}$$

and that of viscous fluxes is given by

$$\iint_{\partial\Omega} \mathbf{F}^{vis}(\mathbf{Q}) \cdot \mathbf{n} dS = \sum_{p \in \Xi(m)} \mathbf{F}^{vis}(\mathbf{Q}_{m,p}) \cdot \Delta \mathbf{S}_{m,p} \tag{7}$$

where $\mathbf{Q}_{m,p}$ stands for the vector of reconstructed flow solutions at the centroid of its p-th face. It represents the averaged value of face. $\Xi(m)$ denotes the set of index for control volume m. $\Delta \mathbf{S}_{m,p}$ denotes the area vector of the p-th face of control volume m. The vector $\Delta \mathbf{S}_{m,p} = \mathbf{n}_{m,p} \Delta S_{m,p}$ with $\Delta S_{m,p}$ standing for the area of p-th face. Thus the finite volume discretization of NS Equations (1) can be expressed as follows:

$$\frac{\mathrm{d}(\mathbf{Q}_m V_m)}{\mathrm{d}t} + \sum_{p \in \Xi(m)} \mathbf{F}(\mathbf{Q}_{m,p}) \cdot \Delta \mathbf{S}_{m,p} = \sum_{p \in \Xi(m)} \mathbf{F}^{vis}(\mathbf{Q}_{m,p}) \cdot \Delta \mathbf{S}_{m,p} \tag{8}$$

where $\mathbf{Q}_m$ is the vector of conservative flow variables at the centroid of a

control volume. It represents the averaged value of control volume m. V_m is the volume of control volume m.

In the discretization form (8), the surface integral is approximated by a summation of the fluxes crossing the faces of the control volume. The geometric parameters, such as faces set $\Xi(m)$, volume V_m, and area vector $\Delta\mathbf{S}_{m,p}$, are the key variables to construct the finite volume scheme. They are inherently contained in the geometric control volume created from the mesh elements surrounding node m. But they are not included in the meshfree points which do not have the geometric topology connectivity between points. Here the algebraic-volume meshfree method is employed to reconstruct these relevant parameters for the meshfree points.

3.2. Algebraic Volume Scheme for Meshfree Points

An algebraic-volume meshfree method was proposed in the previous work [24, 25]. It is briefly introduced here with some modifications. The differential form of NS equations can be written as follows:

$$\frac{\partial \mathbf{Q}}{\partial t} + \left(\frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{F}}{\partial z}\right) = \left(\frac{\partial \mathbf{F}^{vis}}{\partial x} + \frac{\partial \mathbf{F}^{vis}}{\partial y} + \frac{\partial \mathbf{F}^{vis}}{\partial z}\right) \tag{9}$$

Compared to Equation (8) of finite volume scheme, we can find that the face area vectors of control volume are equivalent to a transform matrix which converts face fluxes in the integral form into flux derivatives in the differential form. The face area vectors play a key role in the finite volume scheme. The algebraic-volume meshfree method proposes an effective way to construct this kind of transform matrix for the meshfree points.

Suppose that the computational domain is decomposed by points. For point m illustrated in Figure 1 (b), the set of index for point m is $\tilde{\Xi}(m)$. The symbol $\sim$ represents the difference from those of the finite volume scheme. The total number of neighbours is N_m. And $N_m > d+1$ with d is the number of dimensions. $\mathbf{x}_m = (x_m, y_m, z_m)^T$ represents the coordinate of point m. The midpoint of point m and its p-th neighbor n is given by

$$\mathbf{x}_{m,p} = \frac{1}{2}(\mathbf{x}_m + \mathbf{x}_n) \tag{10}$$

The Taylor series expansion of midpoint fluxes $\mathbf{F}_{m,p}$ at point m is

$$\mathbf{F}_{m,p} = \mathbf{F}_m + \nabla\mathbf{F}_m \cdot (\mathbf{x}_{m,p} - \mathbf{x}_m) + o(\Delta\mathbf{x})^2, \quad p \in \tilde{\Xi}(m) \tag{11}$$

For a viscous flow simulation, the high-aspect-ratio point distribution is usually generated near the viscous wall. A weight ω is added to both sides of the equation to enhance the numerical accuracy in the solution of the inverse matrix. Neglecting the high order part $o(\Delta\mathbf{x})^2$, we can obtain the linear equations for all N_m neighbors

$$\mathbf{A}_m\mathbf{B}_m = \mathbf{C}_m \tag{12}$$

with the abbreviations given by

$$\mathbf{A}_m = \begin{bmatrix} \omega_{m,1}(x_{m,1}-x_m) & \omega_{m,1}(y_{m,1}-y_m) & \omega_{m,1}(z_{m,1}-z_m) & \omega_{m,1} \\ \vdots & \vdots & \vdots & \vdots \\ \omega_{m,p}(x_{m,p}-x_m) & \omega_{m,p}(y_{m,p}-y_m) & \omega_{m,p}(z_{m,p}-z_m) & \omega_{m,p} \\ \vdots & \vdots & \vdots & \vdots \\ \omega_{m,N_m}(x_{m,N_m}-x_m) & \omega_{m,N_m}(y_{m,N_m}-y_m) & \omega_{m,N_m}(z_{m,N_m}-z_m) & \omega_{m,N_m} \end{bmatrix} \tag{13}$$

$$\mathbf{B}_m = \begin{bmatrix} \frac{\partial \mathbf{F}_m}{\partial x} & \frac{\partial \mathbf{F}_m}{\partial y} & \frac{\partial \mathbf{F}_m}{\partial z} & \mathbf{F}_m \end{bmatrix}^T \tag{14}$$

$$\mathbf{C}_m = \begin{bmatrix} \omega_{m,1}\mathbf{F}_{m,1} & \dots & \omega_{m,p}\mathbf{F}_{m,p} & \dots & \omega_{m,N_m}\mathbf{F}_{m,N_m} \end{bmatrix}^T \tag{15}$$

The inverse of distance is chosen as the weight ω in the paper.

The equation is solved by

$$\mathbf{A}_m^T\mathbf{A}_m\mathbf{B}_m = \mathbf{A}_m^T\mathbf{C}_m \tag{16}$$

It is the least-squares solution of Equation (12). If matrix $\mathbf{A}_m^T\mathbf{A}_m$ is singular, point m is moved to avoid the issue. Thus we obtain the solution

$$\mathbf{B}_m = [(\mathbf{A}_m^T\mathbf{A}_m)^{-1}\mathbf{A}_m^T]\mathbf{C}_m \tag{17}$$

Let $\Delta\tilde{S}^i_{m,p}$ represent the p-th element of the i-th row of matrix $[(\mathbf{A}_m^T\mathbf{A}_m)^{-1}\mathbf{A}_m^T]$, where superscript i stands for the i-th row of the vector. We can obtain

$$B^i_m = \sum_{p\in\tilde{\Xi}(m)} C_{m,p}\Delta\tilde{S}^i_{m,p} \tag{18}$$

According to the relation (14), the first three elements of vector $\mathbf{B}_m$ are the derivatives of flux. Therefore, we obtain the derivatives of flux at point m

$$\left\{\begin{array}{l}\frac{\partial \mathbf{F}_m}{\partial x}=\sum_{p\in\tilde{\Xi}(m)}\omega_{m,p}\mathbf{F}_{m,p}\Delta\tilde{S}^1_{m,p}\\ \frac{\partial \mathbf{F}_m}{\partial y}=\sum_{p\in\tilde{\Xi}(m)}\omega_{m,p}\mathbf{F}_{m,p}\Delta\tilde{S}^2_{m,p}\\ \frac{\partial \mathbf{F}_m}{\partial z}=\sum_{p\in\tilde{\Xi}(m)}\omega_{m,p}\mathbf{F}_{m,p}\Delta\tilde{S}^3_{m,p}\end{array}\right. \tag{19}$$

Let

$$\Delta\tilde{\mathbf{S}}_{m,p}=\omega_{m,p}\left[\Delta\tilde{S}^1_{m,p}\quad\Delta\tilde{S}^2_{m,p}\quad\Delta\tilde{S}^3_{m,p}\right]^T \tag{20}$$

We obtain the summation of derivatives of convective fluxes

$$\frac{\partial \mathbf{F}_m}{\partial x}+\frac{\partial \mathbf{F}_m}{\partial y}+\frac{\partial \mathbf{F}_m}{\partial z}=\sum_{p\in\tilde{\Xi}(m)}\mathbf{F}_{m,p}\cdot\Delta\tilde{\mathbf{S}}_{m,p} \tag{21}$$

Compared to the finite volume discretization (Equation (6)), we call the reconstructed vector $\Delta\tilde{\mathbf{S}}_{m,p}$ an algebraic area vector. Similarly, we can get the summation of derivatives of viscous fluxes

$$\frac{\partial \mathbf{F}^{vis}}{\partial x}+\frac{\partial \mathbf{F}^{vis}}{\partial y}+\frac{\partial \mathbf{F}^{vis}}{\partial z}=\sum_{p\in\tilde{\Xi}(m)}\mathbf{F}^{vis}(\mathbf{Q}_{m,p})\cdot\Delta\tilde{\mathbf{S}}_{m,p} \tag{22}$$

Therefore, the discretization of differential form of NS Equations (9) by the algebraic-volume meshfree method is as follows

$$\frac{\mathrm{d}(\mathbf{Q}_m\tilde{V}_m)}{\mathrm{d}t}+\sum_{p\in\tilde{\Xi}(m)}\mathbf{F}(\mathbf{Q}_{m,p})\cdot\Delta\tilde{\mathbf{S}}_{m,p}=\sum_{p\in\tilde{\Xi}(m)}\mathbf{F}^{vis}(\mathbf{Q}_{m,p})\cdot\Delta\tilde{\mathbf{S}}_{m,p} \tag{23}$$

where $\tilde{V}_m$ represents the "volume" of algebraic volume, and $\tilde{V}_m\equiv 1$.

3.3. General Volume Scheme for General Mesh

It can be seen that there are no essential differences between the expression of finite volume scheme (Equation (8)) and that of algebraic-volume meshfree method (Equation (23)). Therefore, the numerical discretization of NS equations can be written in a general form as follows:

$$\frac{\mathrm{d}(\mathbf{Q}_m\mathcal{V}_m)}{\mathrm{d}t}+\sum_{p\in\Xi(m)}\mathbf{F}(\mathbf{Q}_{m,p})\cdot\Delta\mathcal{S}_{m,p}=\sum_{p\in\Xi(m)}\mathbf{F}^{vis}(\mathbf{Q}_{m,p})\cdot\Delta\mathcal{S}_{m,p} \tag{24}$$

where the key symbols are given by

1. $\Xi(m)$ represents the set of general faces. For a mesh node, it is the geometric faces set $\Xi(m)$ of control volume; for a meshfree point, that represents the point cloud set $\tilde{\Xi}(m)$.
2. $\mathcal{V}_m$ denotes the "volume" of general volume. For a mesh node, it is volume V_m of control volume calculated by the geometric formula; for a meshfree point, "volume" $\tilde{V}_m$ of algebraic volume is constant 1.
3. $\Delta\mathcal{S}_{m,p}$ stands for the general face area vector. For a mesh node, it is the geometric face area vector $\Delta\mathbf{S}_{m,p}$ of control volume; for a meshfree point, the algebraic area vector $\Delta\tilde{\mathbf{S}}_{m,p}$ is solved by the weighted least-squares method.

Furthermore, it is found that:

- All these space-related parameters Ξ, $\mathcal{V}$, and $\Delta\mathcal{S}$ of general volume (arbitrary mixing of control volume and algebraic volume) are the fundamental information only calculated once in the pre-processor of flow solver. The mesh-based information is obtained by the geometric formula, whereas the meshfree point employs the weighted least-squares reconstruction.

- All those physics-related parameters ($\mathbf{Q}$, t, $\mathbf{F}(\mathbf{Q})$, and $\mathbf{F}^{vis}(\mathbf{Q})$) and numerical schemes of spatial and temporal discretization are the same for both mesh nodes and meshfree points of the general mesh. Therefore, the numerical discretization is as unified and compact as a single type of mesh. The resulting major CFD code is general.

- The proposed numerical scheme does not need to treat the interface between mesh node and meshfree point exhibited in Figure 1 (c). The general volume naturally includes both control volume and algebraic volume. Thus the general volume scheme solves both without requiring any interface treatment.

Equation (24) is the general volume scheme of NS equations. In this general form, a point cloud becomes a special "mesh element" of which the traditional geometric information of control volume, such as face area vectors, is reconstructed by the weighted least-squares solution. It inherently includes both finite volume and meshfree method. It solves both mesh and meshfree points by a unified scheme. Furthermore, the incredible benefit is interface-free such that the mesh and points can be mixed arbitrarily.

Therefore, both data structure and numerical scheme are as unified and compact as unstructured grid. The general mesh can be defined as a new type of mesh that is more geometrically flexible.

For comparison, the hybrid grid/gridfree method needs to deal with the interface of grid zone and gridfree zone such that it is impossible to arbitrarily mix grid nodes and gridfree points like the general mesh method. Both data structure and numerical scheme are not unified and thus make the hybrid method complicated.

3.3.1. Discretization of the Convective Fluxes

Because of the characteristics of convection, the discretization scheme of convective fluxes affects the stability significantly. It is evaluated by

$$\mathbf{F}(\mathbf{Q}_{m,p}) \cdot \Delta\mathcal{S}_{m,p} = \mathbf{F}^c(\mathbf{Q}^L_{m,p}, \mathbf{Q}^R_{m,p}) \cdot \Delta\mathcal{S}_{m,p} \tag{25}$$

where $\mathbf{Q}^L_{m,p}$ and $\mathbf{Q}^R_{m,p}$ represent the flow values of left- and right-hand sides of the general face, respectively. $\mathbf{F}^c$ stands for the numerical discretization scheme for the convective fluxes, such as central schemes with artificial dissipation [10], flux-vector splitting schemes (Van Leer's [34], AUSM [35]), flux-difference splitting schemes (Roe's [36]), and so on.

It can only reach the first order if we directly take the flow solution of each general volume. To achieve the second-order approximation, we have to reconstruct the left- and right-hand flow solutions at the centre of general face. Assume that the solution is piecewise linearly distributed in each general volume. For a flow variable q, the values of left- and right-hand sides are extrapolated by

$$\begin{cases} q^L_{m,p} = q_m + \phi_m[\nabla q_m \cdot (\mathbf{x}_{m,p} - \mathbf{x}_m)] \\ q^R_{m,p} = q_n + \phi_n[\nabla q_n \cdot (\mathbf{x}_{m,p} - \mathbf{x}_n)] \end{cases} \tag{26}$$

where $\mathbf{x}_{m,p}$ is the coordinate vector of face centre. $\mathbf{x}_m$ stands for the coordinate vector of the centroid of general volume m. Subscript n represents the p-th neighbor of general volume m. ϕ is the limiter function. Its value should tend to be 1 in the smooth region and 0 in the discontinuous region. Two popular limiters are adopted: Barth & Jespersen limiter [37] and Venkatakrishnan limiter [38]. ∇q stands for the gradient of variable q. It is

obtained either by the Green-Gauss method or by the weighted least-squares reconstruction [39]. The former is solved by

$$\nabla q_m = \frac{\sum_{p\in\Xi(m)} q_{m,p}\cdot\Delta\mathcal{S}_{m,p}}{\mathcal{V}_m} \tag{27}$$

3.3.2. Discretization of the Viscous Fluxes

The viscous fluxes are solved by the central difference scheme. The spatial derivatives of velocity and temperature on the general face are calculated by the weighted average method. Since the flow gradient of each general volume has been obtained with the calculation of convective fluxes, it is natural to evaluate the face gradient by averaging.

$$\nabla q_{m,p} = \frac{1}{2}(\nabla q_m + \nabla q_n) \tag{28}$$

where n represents the p-th neighbor of control volume m. As mentioned in [40], this stencil may introduce the decoupling of the solution on quadrilateral or hexahedral cells. Here the weighted average method combines two parts of values: averaged part $\nabla q^{av}_{m,p}$ and corrected part $\delta\nabla q^{cor}_{m,p}$

$$\nabla q_{m,p} = \nabla q^{av}_{m,p}\cdot\mathbf{t}_\perp\frac{\mathbf{t}_\perp}{\|\mathbf{t}_\perp\|^2} + \delta\nabla q^{cor}_{m,p} \tag{29}$$

where $\mathbf{t}_\perp$ stands for the vector perpendicular to the direction of corrected part. According to the study [41], we can get Table 1, where $\mathbf{n}_{m,p}$ represents the outward face direction. The parameters listed in the table do not mean that each row has a correspondence relationship. Any combination of the averaged part and corrected one is feasible.

4. Implicit Time-Marching Method

The semi-discrete form of Equation (24) can be written as follows:

$$\frac{\mathrm{d}(\mathbf{Q}_m\mathcal{V}_m)}{\mathrm{d}t} = -\bar{\mathbf{R}}(\mathbf{Q}_m) = -[\mathbf{R}(\mathbf{Q}_m) - \mathbf{R}^{vis}(\mathbf{Q}_m)] \tag{30}$$

where $\bar{\mathbf{R}}$ is the residual of NS equations. $\mathbf{R}$ and $\mathbf{R}^{vis}$ represent the inviscid and viscous terms, respectively. Implicit discretization of the equation yields

$$\mathcal{V}_m\frac{\Delta\mathbf{Q}^{k+1}_m}{\Delta t} = -\bar{\mathbf{R}}(\mathbf{Q}^{k+1}_m) \tag{31}$$

where k is the time step. Δt is the time step size. And $\Delta \mathbf{Q}_m^{k+1} = \mathbf{Q}_m^{k+1} - \mathbf{Q}_m^k$. Since the residual value is implicit, it is linearized by

$$\bar{\mathbf{R}}(\mathbf{Q}_m^{k+1}) \approx \bar{\mathbf{R}}(\mathbf{Q}_m^k) + \mathbf{A}_m^k \Delta \mathbf{Q}_m^{k+1} \tag{32}$$

where $\mathbf{A}$ is the Jacobian matrix, $\mathbf{A} = \partial \bar{\mathbf{R}} / \partial \mathbf{Q}$.

$$\begin{cases} \mathbf{A}_m^k = \mathbf{A}_m^{inv,k} + \mathbf{A}_m^{vis,k} \\ \mathbf{A}_m^{inv,k} = \sum_{p \in \Xi(m)} \frac{\partial \mathbf{F}_{m,p}^k}{\partial \mathbf{Q}} \cdot \Delta \mathcal{S}_{m,p} \\ \mathbf{A}_m^{vis,k} = \sum_{p \in \Xi(m)} \frac{\partial \mathbf{F}_{m,p}^{vis,k}}{\partial \mathbf{Q}} \cdot \Delta \mathcal{S}_{m,p} \end{cases} \tag{33}$$

Thus Equation (31) is rewritten as follows:

$$\frac{\mathcal{V}_m}{\Delta t} \Delta \mathbf{Q}_m^{k+1} + \sum_{p \in \Xi(m)} \mathbf{A}_{m,p}^k \Delta \mathbf{Q}_{m,p}^{k+1} = -\bar{\mathbf{R}}(\mathbf{Q}_m^k) \tag{34}$$

After temporal linearization, the previous implicit right-hand residual term becomes explicitly solved now.

4.1. Flux Splitting of the Inviscid Term

The maximum eigenvalue splitting scheme [42] is utilized for the Jacobian matrix $\mathbf{A}_m^{inv}$

$$\begin{cases} \mathbf{A}_m^{inv,+} = \frac{\mathbf{A}_m^{inv} + \mathbf{I} \beta \lambda_{m,p}^{inv}}{2} \\ \mathbf{A}_m^{inv,-} = \frac{\mathbf{A}_m^{inv} - \mathbf{I} \beta \lambda_{m,p}^{inv}}{2} \end{cases} \tag{35}$$

where $\mathbf{I}$ is a 5×5 identity matrix (4×4 for 2D). β stands for a relaxation factor. $\lambda_{m,p}^{inv}$ represents the largest eigenvalue of Jacobian matrix

$$\lambda_{m,p}^{inv} = |\mathbf{v}_{m,p} \cdot \Delta \mathcal{S}_{m,p}| + c_{m,p} \|\Delta \mathcal{S}_{m,p}\| \tag{36}$$

where $c_{m,p}$ denotes the speed of sound on the face. Thus we get the splitting scheme as follows:

$$\mathbf{A}_{m,p}^{inv,k} \Delta \mathbf{Q}_{m,p}^{k+1} \approx \mathbf{A}_{m,p}^{inv,k,+} \Delta \mathbf{Q}_m^{k+1} + \mathbf{A}_{m,p}^{inv,k,-} \Delta \mathbf{Q}_n^{k+1} \tag{37}$$

Substituting into Equation (34), we obtain

$$\frac{\mathcal{V}_m}{\Delta t} \Delta \mathbf{Q}_m^{k+1} + \sum_{p \in \Xi(m)} \left(\mathbf{A}_{m,p}^{inv,k,+} \Delta \mathbf{Q}_m^{k+1} + \mathbf{A}_{m,p}^{inv,k,-} \Delta \mathbf{Q}_n^{k+1} \right) = -\bar{\mathbf{R}}(\mathbf{Q}_m^k) \tag{38}$$

or

$$\left(\frac{\mathcal{V}_m}{\Delta t}\mathbf{I}+\sum_{p\in\Xi(m)}\mathbf{A}_{m,p}^{inv,k,+}\right)\Delta\mathbf{Q}_m^{k+1}+\sum_{p\in\Xi(m)}\mathbf{A}_{m,p}^{inv,k,-}\Delta\mathbf{Q}_n^{k+1}=-\bar{\mathbf{R}}(\mathbf{Q}_m^k) \quad (39)$$

If the Jacobian matrix $\mathbf{A}_{m,p}^{inv}$ is calculated by the first-order precision, we have

$$\left(\frac{\mathcal{V}_m}{\Delta t}+\frac{\beta}{2}\sum_{p\in\Xi(m)}\lambda_{m,p}^{inv}\right)\Delta\mathbf{Q}_m^{k+1}+\sum_{p\in\Xi(m)}\mathbf{A}_{m,p}^{inv,k,-}\Delta\mathbf{Q}_n^{k+1}=-\bar{\mathbf{R}}\left(\mathbf{Q}_m^k\right) \quad (40)$$

This equation is solved by multiple symmetric Gauss-Seidel iterations. Each iteration is operated as a pair of sweeps: one forward and one backward.

4.2. Treatment of the Viscous Term

The Jacobian matrix of the viscous term is relatively complex and difficult to solve accurately. In general, the viscous term has less influence on the convergence and stability of the simulation than the convection term. So only the effect of viscosity on the maximum eigenvalue is considered here

$$\lambda_m^{vis}=\left[max\left(\frac{4}{3\rho_m},\frac{\gamma}{\rho_m}\right)\right]\left(\frac{\mu_m}{P_r}+\frac{\mu_{t,m}}{P_{rt}}\right)\frac{\left(\Delta S_m^x\right)^2+\left(\Delta S_m^y\right)^2+\left(\Delta S_m^z\right)^2}{\mathcal{V}_m} \quad (41)$$

where

$$\begin{cases}\Delta S_m^x=\frac{1}{2}\sum_{p\in\Xi(m)}\left|\Delta\mathcal{S}_{x,m,p}\right|\\ \Delta S_m^y=\frac{1}{2}\sum_{p\in\Xi(m)}\left|\Delta\mathcal{S}_{y,m,p}\right|\\ \Delta S_m^z=\frac{1}{2}\sum_{p\in\Xi(m)}\left|\Delta\mathcal{S}_{z,m,p}\right|\end{cases} \quad (42)$$

5. Results and Discussion

The general mesh method is implemented in the in-house code $\mathbf{GC_{fd}}$ [25] that originates from the cell-centred finite volume method and then is extended to the gridfree method. First of all, it is verified and validated by the low-speed and transonic flows. The interface-free mixing of mesh and points is emphasized. And then, it is applied to improve the convergence of simulation. The low-quality elements are converted into points to improve the quality.

5.1. Verification and Validation

The general mesh method is investigated by different mixing ideas of mesh nodes and meshfree points. A wing with a cross section of NACA0012 airfoil and one-chord span is employed to comprehensively verify the proposed method. Two typical high-Reynolds number flows, namely, low speed and transonic, are simulated. The flow conditions of low speed are at Mach number (Ma) of 0.15, angle of attack (α) of 0°, and Reynolds number (Re) of 6×10^6. For the transonic flow, Mach number (Ma) is 0.775; the angle of attack (α) is 2.05°; and Reynolds number (Re) is 10^7. The Roe's scheme [36] is adopted for the simulation of low-speed flow, whereas AUSM [35] for the transonic flow solution. The Venkatakrishnan limiter [38] is applied in the simulations.

The basic grid is 225×65 from NASA website [43]. It is extruded to generate the 3D grid with 4 layers. The different general meshes are generated by converting part of mesh nodes into meshfree points for consistency comparisons. Two types of mesh/meshfree unifying idea, namely, zonal meshing and fusion meshing, are utilized to obtain the general meshes. The former includes mesh zone and point zone such that the same type of mesh can connect well in the respective zones. The interface of two zones is emphasized in the investigation. The latter converts a certain percentage of mesh nodes into meshfree points. The mesh nodes and meshfree points are sufficiently mixed. The interface appears everywhere in order to further investigate the interface-free feature of general volume method. Each meshfree point cloud collects all the nodes which share the same mesh element with the particular point.

5.1.1. Compared to Traditional Schemes

The general volume scheme is verified in this section. Its results should be close to those of finite volume scheme when only mesh is used. Similarly, the results should be close to those of gridfree method when only points are utilized. The traditional grid is shown in Figure 2 (a) and gridfree points are exhibited in Figure 2 (b). The latter is converted from the grid nodes in order to keep consistent. The solutions obtained by the grid-based finite volume method are compared with those by general mesh method with 100% mesh and 0% points. Similarly, the solutions of gridfree method are compared with those of general mesh method with pure meshfree points.

Figures 3 and 4 show the convergence history of maximum and average residuals for the low speed and transonic flow simulations, respectively. The

residual of the energy equation is compared. The finite volume results are based on the traditional vertex-based finite volume method. AV gridfree method represents the algebraic-volume gridfree method [25]. The corresponding general mesh method with pure mesh or points utilizes dashed curves. The flowfield is uniformly initialized by the farfield settings for all simulations. All methods converge well for both low speed and transonic high-Reynolds number flows. For both flow simulations, the convergence history of general volume scheme with pure mesh is quite close to that of finite volume scheme. And it is close to that of AV gridfree method when only points are employed by the general mesh method. The slight difference of residuals is reasonable because of a slight difference in boundary treatment.

Figures 5 and 6 exhibit the non-dimensional pressure solved by different methods for low-speed and transonic flows, respectively. The same scales of contour are applied in all the images of which use the same flow condition. The pressure contour shows almost the same results for both finite volume method and general mesh method with 100% mesh in the respective low-speed and transonic flows. The contour lines and shock wave width look the same between two numerical methods. Meanwhile, the pressure contour is also nearly the same for AV gridfree method and general mesh method with pure points in the respective flows.

A detailed comparison is demonstrated in Figure 7. The surface pressure coefficients obtained by different numerical methods are compared. The results of general mesh with pure mesh (dashed curve) and finite volume method (solid curve) almost coincide for each flow condition. Those of general mesh with pure points (dashed curve) and gridfree method (solid curve) perform the excellent agreement as well.

The results evidently verify that the proposed general mesh method achieves good consistency with the respective finite volume and gridfree methods. The numerical solutions are in good agreement between two relevant counterparts.

5.1.2. Zonal General Meshing

The zonal general mesh is demonstrated in Figures 8. The inner zone is defined by a rectangle of $[x \in (-0.15, 1.15), y \in (-0.15, 0.15)]$. The mesh nodes outside the rectangle are converted into points for the general mesh of inner mesh zone and outer point zone shown in Figure 8 (a). Similarly, the mesh nodes inside the rectangle are converted into points for the general mesh of outer mesh zone and inner point zone exhibited in Figure 8 (b). There exists a clear interface between two zones. Therefore, this section is

employed to investigate the feature of solved flow across the interface.

Figures 9 and 10 show the convergence history of maximum and average residuals for the low speed and transonic flows, respectively. In the figure, "inner mesh & outer point" denotes that the inner zone is decomposed by mesh and the outer by meshfree points. "inner point & outer mesh" indicates that the inner zone is decomposed by meshfree points and the outer by mesh. The flowfield is uniformly initialized by the farfield settings. All methods show good convergence for both low-speed and transonic flows. The convergence history of general volume scheme is quite close to each other for both zonal general meshes. It is similar to that of the finite volume scheme and slightly better than that of the gridfree method. The results indicate that the discontinuous shock wave does not affect the convergence of zonal general mesh even if there exists an interface between two types of mesh.

Figures 11 and 12 present the solved non-dimensional pressure of the low speed and transonic flows, respectively. The results obtained by two general meshes look quite close and are very similar to those of finite volume and gridfree methods. Meanwhile, it is found that the pressure contour lines (depicted by black colour) across the irregular interface (shown in the red curve) between mesh zone and point zone are continuous. The close-up of the shock wave and relevant nodes/points is exhibited in the right top box of each sub-figure for transonic flow shown in Figure 12. The shock waves of both zonal meshing strategies are solved incredibly well across the interface of two zones by the interface-free general volume scheme. No oscillation is found on the contour lines.

The comparison of surface pressure coefficients obtained by different methods is demonstrated in Figure 13. The results are in good agreement with the measurement for the low-speed flow exhibited in Figure 13 (a). The computational surface pressure data coincide with each other. In order to show the resolution of computational results, the meshfree points that also represent the distribution of mesh nodes for the respective methods are depicted as purple dots along the curve of gridfree method in Figure 13 (b). The comparisons indicate that the shock wave is accurately captured within two nodes or points without any oscillation. The solutions obtained by both zonal meshes agree well with experimental data. They are very close to those solved by the finite volume method and better than those by the gridfree method.

The comparisons indicate that the general mesh method with zonal mixing of mesh and points achieves the comparable accuracy and convergence speed with the finite volume method and performs better than the gridfree

method. Moreover, it does not require any treatment for the interface of two zones.

5.1.3. Fusion General Meshing

The zonal meshing treats the flow domain as two separate mesh/points zones. The results indicate that the general volume method successfully solves the interface between mesh zone and point zone without any oscillations. Thus the test cases go further in this section. The general mesh is generated by the fusion mixing of mesh nodes and meshfree points such that they are sufficiently mixed together. The fusion general mesh is shown in Figure 14. The mesh nodes are converted into meshfree points through the value of node ID. For example, in the fusion meshing of 50% mesh nodes and 50% meshfree points shown in Figure 14 (b), the mesh nodes with odd IDs remain mesh connectivity, whereas those with even IDs are converted into meshfree points. The interface between mesh and point is not regular and appears everywhere. Therefore, the mesh nodes and meshfree points are arbitrarily mixed in this fusion meshing strategy.

For the critical fusion of 50% mixing depicted in Figure 14 (b), the mesh nodes and meshfree points are connected with each other everywhere. Each node is surrounded by points and each point is surrounded by nodes (shown in the close-up of Figure 14 (b)). This particular meshing idea is employed to comprehensively demonstrate the extreme situation of general mesh and to intensively investigate the ability of the general volume scheme which solves the mesh node and meshfree point without any interface treatment. It should be noted that the number of points is only a small proportion of general mesh in most of the applications. Namely, situations such as Figures 14 (c) and (d) are more commonplace. On the contrary, the mixing of 1% mesh nodes and 99% meshfree points shown in Figure 14 (a) is only used to investigate the consistency of general volume method.

The comparison of convergence histories of different methods is shown in Figures 15 and 16 for the low speed and transonic flows, respectively. Both the maximum and average residuals converge well for all four general meshes. They exhibit similar convergence speed as the traditional finite volume method. The convergence is slightly better than that of the algebraic-volume gridfree method if only a small proportion of mesh nodes are converted into points. Figures 17 and 18 show the comparisons of non-dimensional pressure contour. Each tiny red dot represents a meshfree point that is converted from a mesh node. The solved flowfields are close to those

of the gridfree method shown before when the percentage of meshfree points is high. The flowfields are close to those of the finite volume method shown before when the percentage of meshfree points becomes low. The black contour lines are remarkably continuous even for the discontinuous flow such as the shock wave. The critical investigation — 50% fusion mixing — impressively demonstrates the interface-free feature of general mesh method which naturally solves two types of mesh together via the general volume method.

Figure 19 compares the surface pressure coefficients obtained by different general meshes. The results of general meshes agree well with those of experiment, finite volume method, and gridfree method. For the low-speed flow shown in Figure 19 (a), the computational data agree remarkably well with each other and with experimental result as well. For the transonic flow exhibited in Figure 19 (b), the purple dots represent the meshfree points or mesh nodes on the airfoil surface. The result of 1% nodes and 99% points is close to that of the gridfree method. It becomes closer to that of the finite volume method when the proportion of points decreases.

The comparisons indicate that the gridfree method is less accurate than the grid-based finite volume method. The results obtained by the general mesh method show good consistency. Moreover, it automatically solves the interface of mesh node and meshfree point. The obtained flow is accurate and oscillation is not observed.

5.2. Application to Improve the Simulation

It is difficult to generate a mesh of which each element can be guaranteed to achieve high quality. Those low-quality elements directly affect the convergence and accuracy of simulations. After a mesh is obtained, its quality is assessed and the nodes of low-quality elements are converted into meshfree points. Here is an example which utilizes the general mesh method to improve the convergence of simulation.

The configuration of HIgh REynolds Number AeroStructural Dynamics (HIRENASD) Project [44] is investigated. The sweptback angle of wing is 34°, and the BAC3-11 airfoil is utilized in any sections. The wingspan is $1.28857m$, reference length $0.3445m$, and reference area $0.3926m^2$. The farfield of computational domain is one hundred times of reference length. The unstructured grid with high-aspect ratio prism is provided by the official website.

The mesh is exhibited in Figure 20. The total number of grid elements is 8 million, and the number of grid nodes 3 million. As shown in Figures 20 (c)

and (d), the grid is refined at the leading edge, trailing edge, and wing tip, etc. The results of this original grid solved by the traditional finite volume method are compared to those obtained by the general mesh method which converts the low-quality elements into meshfree points (shown in Figure 20 (f)). The total number of meshfree points is around 3000 which is 0.1% of the entire general mesh. The image of converted points indicates that the mesh quality is difficult to improve at the leading edge, trailing edge, wing/fuselage junction, and wing tip even if it is a relatively simple configuration. It usually requires mesh refinement in these high-curvature regions to better represent the geometry. Consequently, it becomes a challenge for the transition of volume mesh between these refined regions and the others. This fact also indicates that the general mesh has great potential to improve the simulations.

The comparison of convergence histories of both methods is shown in Figure 21. The abbreviation "Max." and "Av." stand for the maximum and average residuals of the energy equation, respectively. "Cl" and "Cd" represent the lift and drag coefficients, respectively. The freestream Mach number (Ma) is 0.8 and the reference length based Reynolds number (Re) is 7×10^6. The angle of attack (α) is 1.5°. Courant-Friedrichs-Lewy (CFL) number is 1 for both methods. The flow is uniformly initialized by the freestream condition. SA turbulence model is utilized in simulation. In the beginning, the residual convergence histories of both finite volume and general mesh methods are similar. Henceforth their difference becomes evident. Both the maximum and average residuals of the traditional finite volume method based on the original unstructured grid do not converge well due to the influence of low-quality elements. On the contrary, those of the general mesh method converge much better. From the comparison of Figure 21 (b), the convergence of aerodynamic force shows similar performance for both methods.

The flowfield solved by the general mesh method is exhibited in Figure 22. The pressure contour on the boundary is compared in Figure 22 (a) where "P" represents the non-dimensional pressure. The weak shock wave can be found on the top of supercritical wing. A slice of flowfield is shown in Figure 22 (b) where "Ma" stands for the Mach number. The shock wave can be found near the upper surface. The boundary layer is also clearly demonstrated. A close-up shown on the right top side of Figure 22 (b) compares the obtained shock wave with the scale of mesh cells. The width of shock is sharply captured within two cells.

Figure 23 shows the comparison of surface pressure coefficients on different spanwise stations. "C" represents the local chord. The results obtained by the finite volume and general mesh methods are in quite good agreement. They almost coincide on all the spanwise stations. The comparison indicates that the mesh quality is not too bad to affect the surface pressure obtained by the finite volume method because the geometry is relatively easy for mesh generation.

6. Conclusions

The general mesh method is proposed. It includes all three existing types of mesh and the arbitrary mixing of mesh and points. The mesh nodes and meshfree points are naturally solved by a unified numerical method — general volume scheme. The general mesh method has the respective advantages of physical accuracy of mesh and geometric flexibility of points. The numerical method is as unified and compact as the unstructured grid, but the meshing published separately becomes much more flexible. The meshing is as automated and flexible as the gridfree method, whereas the numerical solution becomes more accurate and robust. The general mesh method is well validated and the results show good agreement with experimental data. The impressive results obtained by the fusion mixing of mesh and points demonstrate the great potential of this interface-free scheme.

Both low-speed and transonic high-Reynolds number flows of a wing with a cross section of NACA0012 airfoil are employed to investigate the general volume method. Both zonal and fusion general meshing strategies are utilized to convert part of mesh nodes into meshfree points. All the simulations show quite good convergence and accuracy. The interface-free benefit of general volume scheme is well demonstrated. Moreover, the comparisons indicate that both convergence and accuracy of the general mesh method are comparable to those of the traditional finite volume method and perform better than those of the gridfree method if only a small percentage of mesh nodes are converted into meshfree points.

The low mesh quality elements significantly affect the convergence of simulations. And it is difficult to generate a mesh of which each element achieves high quality. The general mesh method demonstrates the potential to improve the convergence of simulation in the case of HIRENASD which has a relatively simple geometry. The general mesh is generated by converting the nodes of low-quality elements into meshfree points. The finite volume

method based on the unstructured grid can not converge well, whereas the proposed general mesh method exhibits much better performance.

References

[1] S. K. Lele, Compact finite difference schemes with spectral-like resolution, Journal of Computational Physics 103 (1992) 16–42.

[2] D. Fauconnier, C. D. Langhe, E. Dick, A family of dynamic finite difference schemes for large-eddy simulation, Journal of Computational Physics 228 (2009) 1830–1861.

[3] Z. F. Tian, P. Yu, An efficient compact difference scheme for solving the streamfunction formulation of the incompressible Navier–Stokes equations, Journal of Computational Physics 230 (2011) 6404–6419.

[4] J. Fernández-Fidalgo, S. Clain, L. Ramírez, I. Colominas, X. Nogueira, Very high-order method on immersed curved domains for finite difference schemes with regular Cartesian grids, Computer Methods in Applied Mechanics and Engineering 360 (2020) 112782.

[5] O. Zienkiewicz, R. Taylor, P. Nithiarasu, The Finite Element Method for Fluid Dynamics, Elsevier, 2014.

[6] B. Cockburn, C.-W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems, Journal of Computational Physics 141 (1998) 199–224.

[7] F. Bassi, L. Botti, A. Colombo, D. D. Pietro, P. Tesini, On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations, Journal of Computational Physics 231 (2012) 45–65.

[8] C. Lehrenfeld, J. Schöberl, High order exactly divergence-free hybrid discontinuous Galerkin methods for unsteady incompressible flows, Computer Methods in Applied Mechanics and Engineering 307 (2016) 339–361.

[9] J. Juno, A. Hakim, J. TenBarge, E. Shi, W. Dorland, Discontinuous Galerkin algorithms for fully kinetic plasmas, Journal of Computational Physics 353 (2018) 110–147.

[10] A. Jameson, W. Schmidt, E. Turkel, Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes, in: 14th Fluid and Plasma Dynamics Conference, American Institute of Aeronautics and Astronautics, 1981.

[11] J. Blazek, Computational Fluid Dynamics: Principles and Applications, Elsevier, 2015.

[12] Y. Ito, A. Shih, R. Koomullil, N. Kasmai, M. Jankun-Kelly, D. Thompson, Solution adaptive mesh generation using feature-aligned embedded surface meshes, AIAA Journal 47 (2009) 1879–1888.

[13] Y. Jiang, Z. Ye, A cell-centered finite volume method for arbitrary grid type, Chinese Journal of Theoretical and Applied Mechanics 42 (2010) 830–837.

[14] G. Wang, H. H. Mian, Y. Liu, Z. Ye, A hybrid implicit scheme for solving Navier-Stokes equations, International Journal for Numerical Methods in Fluids 78 (2015) 319–334.

[15] S. Lorenzi, A. Cammi, L. Luzzi, G. Rozza, POD-Galerkin method for finite volume approximation of Navier–Stokes and RANS equations, Computer Methods in Applied Mechanics and Engineering 311 (2016) 151–179.

[16] L. M. Vieira, M. Giacomini, R. Sevilla, A. Huerta, A second-order face-centred finite volume method for elliptic problems, Computer Methods in Applied Mechanics and Engineering 358 (2020) 112655.

[17] X. K. Zhang, K.-C. Kwon, S.-K. Youn, The least-squares meshfree method for the steady incompressible viscous flow, Journal of Computational Physics 206 (2005) 182–207.

[18] Y. Sanyasiraju, G. Chandhini, Local radial basis function based gridfree scheme for unsteady incompressible viscous flows, Journal of Computational Physics 227 (2008) 8922–8948.

[19] A. Katz, A. Jameson, Edge-based meshless methods for compressible flow simulations, in: 46th AIAA Aerospace Sciences Meeting and Exhibit, American Institute of Aeronautics and Astronautics, 2008.

[20] H. Wang, J. Periaux, A fast meshless method coupled with artificial dissipation for solving 2d euler equations, Computers & Fluids 71 (2013) 83–90.

[21] N. Munikrishna, N. Balakrishnan, Turbulent flow computations on a hybrid Cartesian point distribution using meshless solver LSFD-u, Computers & Fluids 40 (2011) 118–138.

[22] M. Namvar, A. Jahangirian, An investigation of mesh-less calculation for compressible turbulent flows, Computers & Fluids 86 (2013) 483–489.

[23] M. Dehghan, M. Abbaszadeh, The use of proper orthogonal decomposition (POD) meshless RBF-FD technique to simulate the shallow water equations, Journal of Computational Physics 351 (2017) 478–510.

[24] Y. Jiang, Numerical solution of Navier–Stokes equations on generalized mesh and its applications, Ph.D. thesis, Northwestern Polytechnical University, 2013.

[25] Y. Jiang, Algebraic-volume meshfree method for application in finite volume solver, Computer Methods in Applied Mechanics and Engineering 355 (2019) 44–66.

[26] H. Luo, J. D. Baum, R. Löhner, A hybrid Cartesian grid and gridless method for compressible flows, Journal of Computational Physics 214 (2006) 618–632.

[27] T. P. Fries, H. G. Matthies, A stabilized and coupled meshfree/meshbased method for the incompressible Navier-Stokes equations—part I: Stabilization, Computer Methods in Applied Mechanics and Engineering 195 (2006) 6205–6224.

[28] T. P. Fries, H. G. Matthies, A stabilized and coupled meshfree/meshbased method for the incompressible Navier-Stokes equations—part II: Coupling, Computer Methods in Applied Mechanics and Engineering 195 (2006) 6191–6204.

[29] G. Wang, Z. Ye, C. Li, C. Jiang, A unified gridless/FVM solution method for simulating high reynolds number viscous flow, Proceedings of ICCES'07 (2007) 519–528.

[30] X. Su, D. Sasaki, K. Nakahashi, Cartesian mesh with a novel hybrid WENO/meshless method for turbulent flow calculations, Computers & Fluids 84 (2013) 69–86.

[31] A. Javed, K. Djijdeli, J. T. Xing, A coupled meshfree-mesh-based solution scheme on hybrid grid for flow-induced vibrations, Acta Mechanica 227 (2016) 2245–2274.

[32] P. Spalart, S. Allmaras, A one-equation turbulence model for aerodynamic flows, in: 30th Aerospace Sciences Meeting and Exhibit, American Institute of Aeronautics and Astronautics, 1992.

[33] F. R. Menter, Two-equation eddy-viscosity turbulence models for engineering applications, AIAA Journal 32 (1994) 1598–1605.

[34] B. van Leer, Towards the ultimate conservative difference scheme. V. a second-order sequel to Godunovs method, Journal of Computational Physics 32 (1979) 101–136.

[35] M. S. Liou, A sequel to AUSM: AUSM+, Journal of Computational Physics 129 (1996) 364–382.

[36] P. L. Roe, Characteristic-based schemes for the Euler equations, Annual Review of Fluid Mechanics 18 (1986) 337–365.

[37] T. Barth, D. Jespersen, The design and application of upwind schemes on unstructured meshes, in: 27th Aerospace Sciences Meeting, American Institute of Aeronautics and Astronautics, 1989.

[38] V. Venkatakrishnan, Convergence to steady state solutions of the Euler equations on unstructured grids with limiters, Journal of Computational Physics 118 (1995) 120–130.

[39] D. Mavriplis, Revisiting the least-squares procedure for gradient reconstruction on unstructured meshes, in: 16th AIAA Computational Fluid Dynamics Conference, American Institute of Aeronautics and Astronautics, 2003.

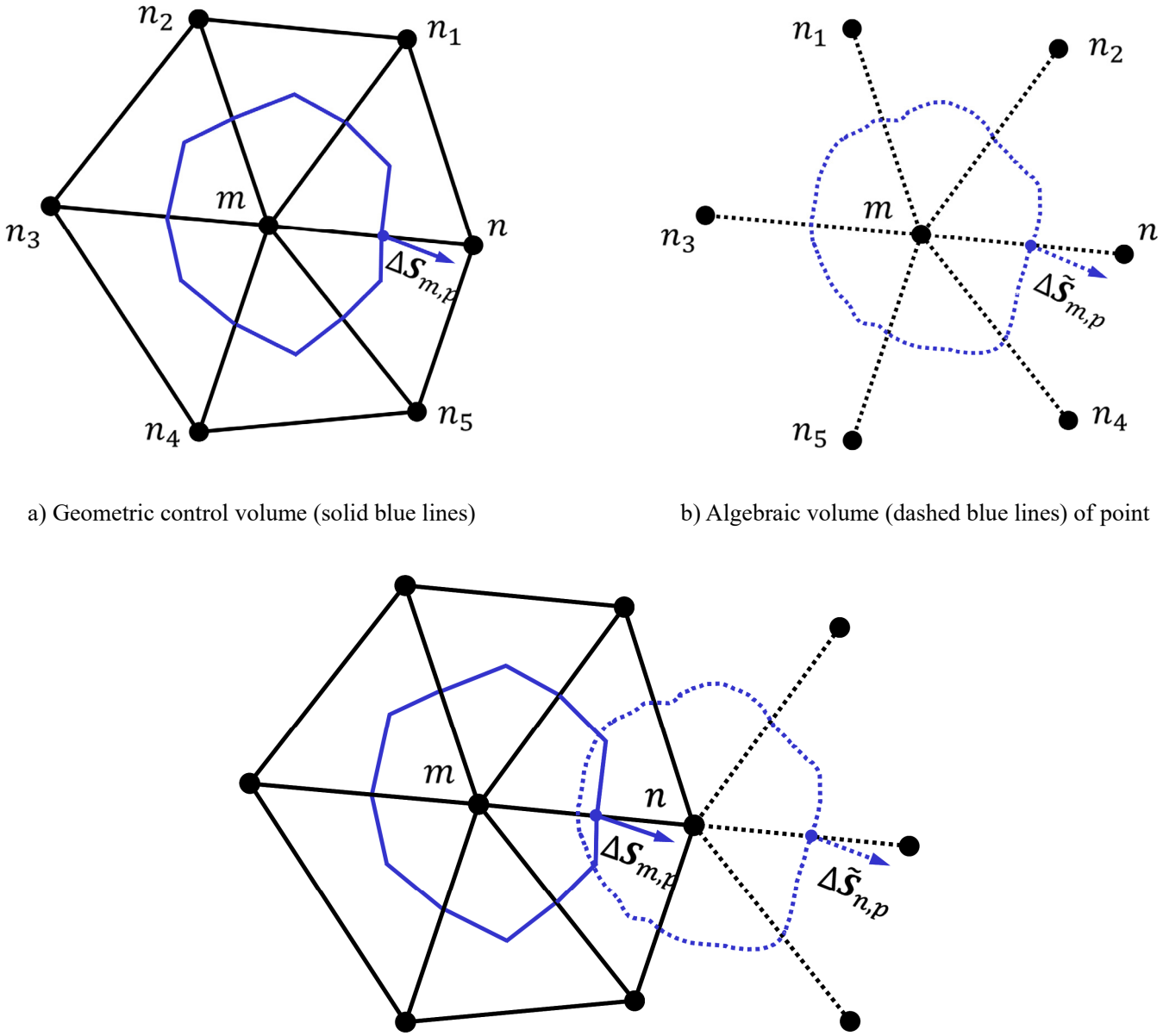
[40] A. Haselbacher, J. Blazek, Accurate and efficient discretization of Navier-Stokes equations on mixed grids, AIAA Journal 38 (2000) 2094–2102.

[41] A. Cary, A. Dorgan, M. Mani, Towards accurate flow predictions using unstructured meshes, in: 19th AIAA Computational Fluid Dynamics, American Institute of Aeronautics and Astronautics, 2009.

[42] G. Wang, Y. Jiang, Z. Ye, An improved LU-SGS implicit scheme for high Reynolds number flow computations on hybrid unstructured mesh, Chinese Journal of Aeronautics 25 (2012) 33–41.

[43] NASA, https://turbmodels.larc.nasa.gov/naca0012_val.html, Turbulence Modeling Resource (2019).

[44] NASA, https://c3.nasa.gov/dashlink/static/media/other/aepw.htm, 1st AIAA Aeroelastic Prediction Workshop (AePW-1) (2012).



a) Geometric control volume (solid blue lines)

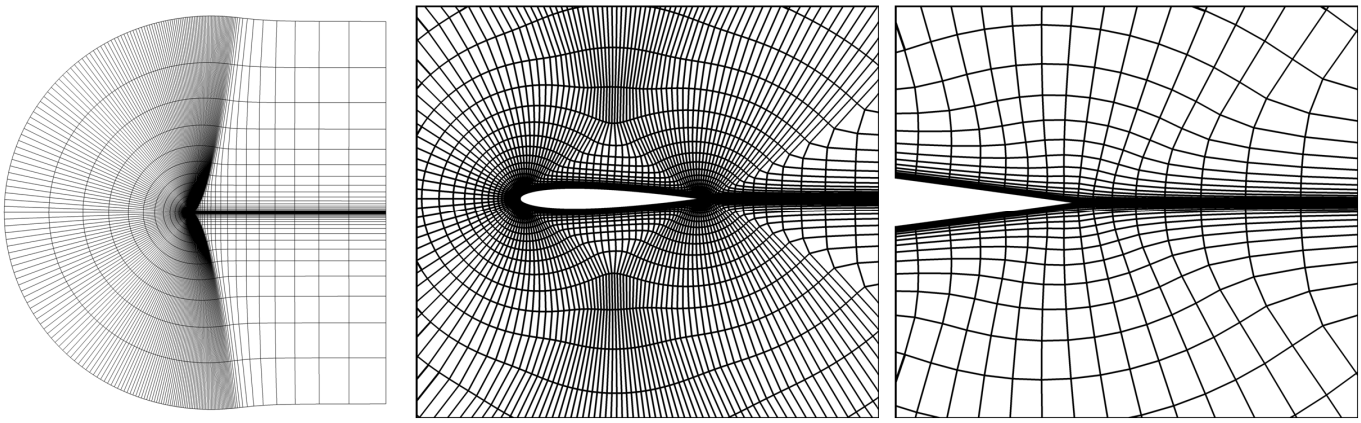
b) Algebraic volume (dashed blue lines) of point

c) Schematic of the interface-free general volume method with control volume and algebraic volume

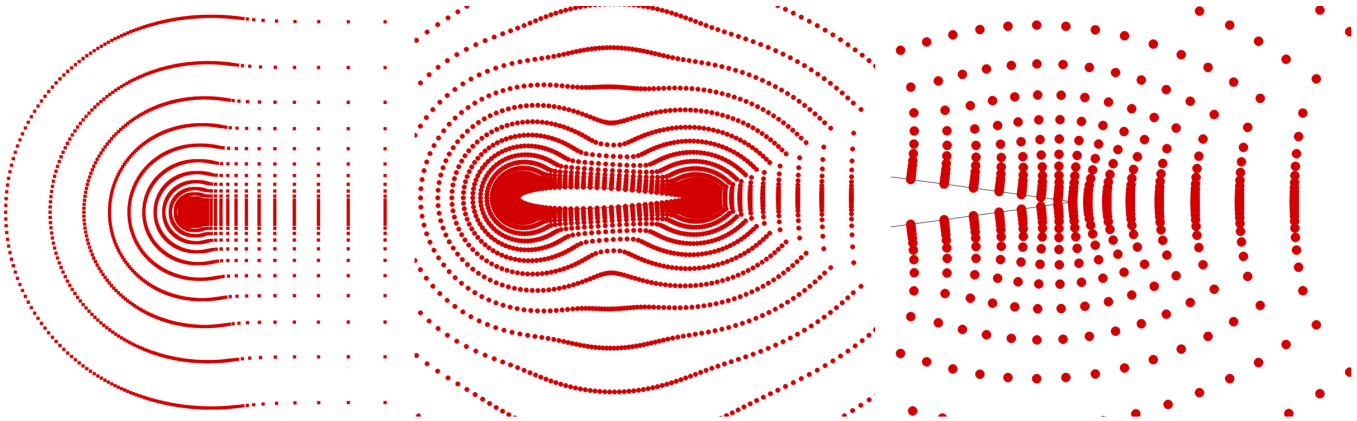
Figure 1 2D examples of general volume for mesh and point

Table 1 Values of averaged part and corrected part

Averaged part	Corrected part and its direction
$\frac{\nabla q_m + \nabla q_n}{2}$	$\frac{\vec{t}}{\|\vec{t}\|^2}(q_n - q_m);\ \vec{t} = \vec{x}_n - \vec{x}_m$
$\frac{V_m \nabla q_m + V_n \nabla q_n}{V_m + V_n}$	$\frac{\vec{t}}{\vec{t}\cdot(\vec{x}_n - \vec{x}_m)}(q_n - q_m);\ \vec{t} = \vec{n}_{m,p}$
$\frac{\|\vec{x}_{m,p} - \vec{x}_m\|\nabla q_m + \|\vec{x}_{m,p} - \vec{x}_n\|\nabla q_n}{\|\vec{x}_{m,p} - \vec{x}_m\| + \|\vec{x}_{m,p} - \vec{x}_n\|}$	$\frac{\vec{t}}{\vec{t}\cdot(\vec{x}_n - \vec{x}_m)}[q_n - q_m + (\nabla q_n \cdot \vec{t}_\perp)(\vec{x}_{m,p} - \vec{x}_n)\cdot\vec{t}_\perp - (\nabla q_m \cdot \vec{t}_\perp)(\vec{x}_{m,p} - \vec{x}_m)\cdot\vec{t}_\perp];\ \vec{t} = \vec{n}_{m,p}$

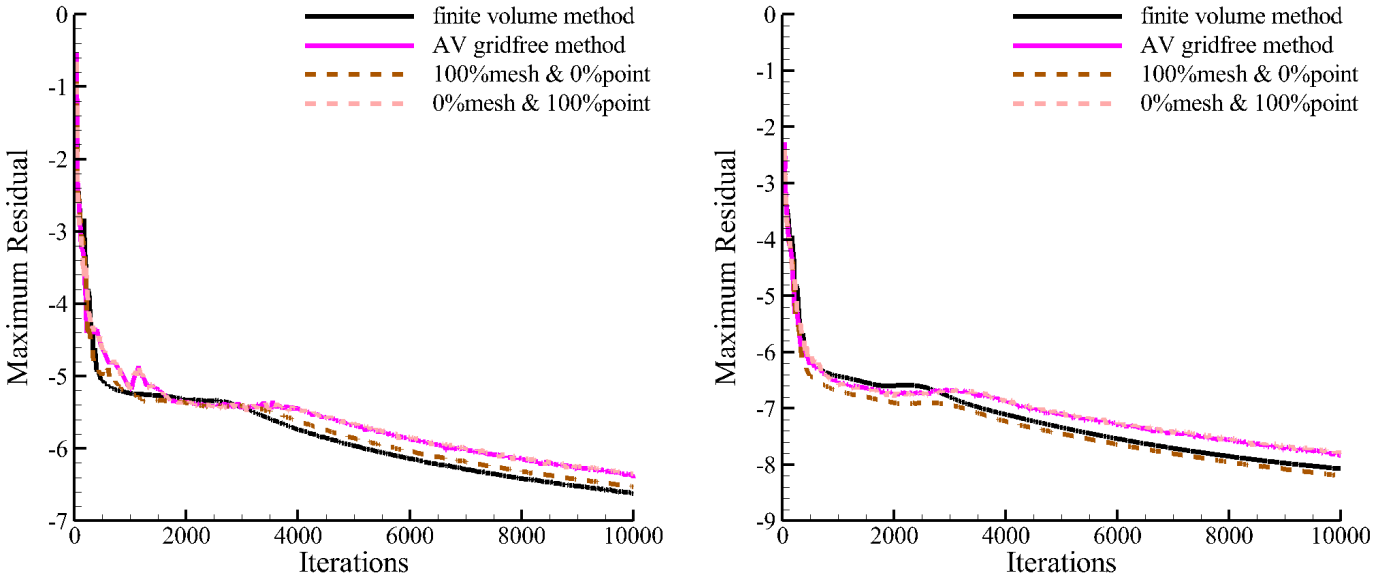


a) Traditional grid with geometric elements



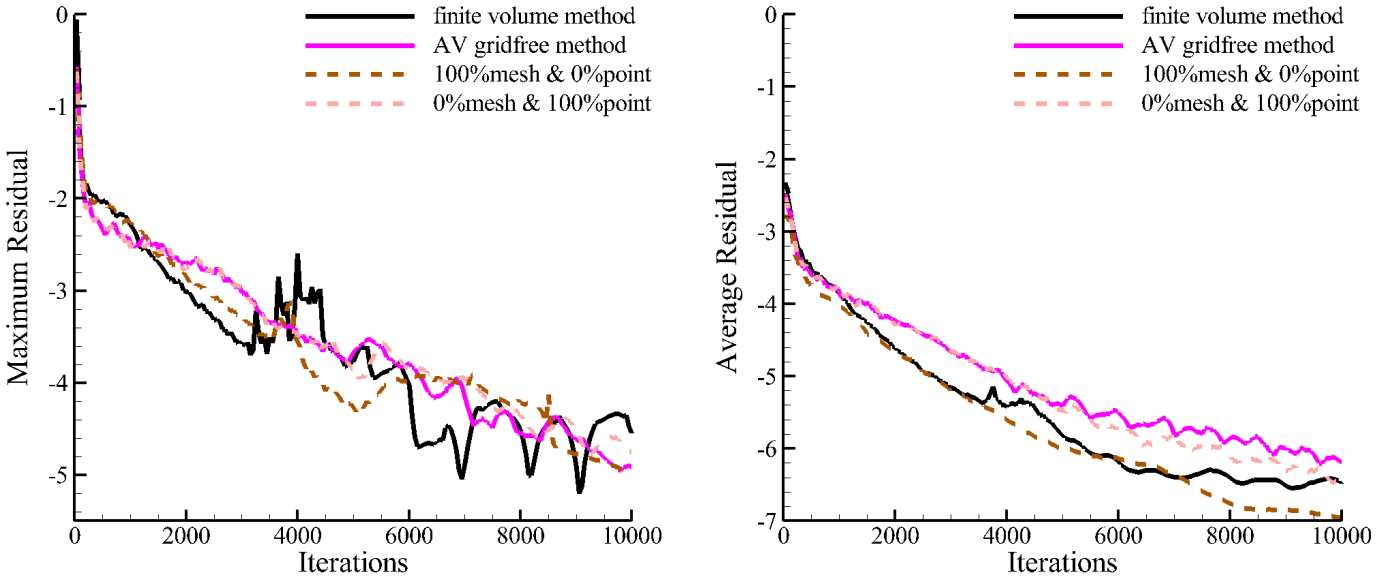
b) Gridfree points without geometric connectivity

Figure 2　Grid elements and gridfree points for a NACA0012 airfoil



a) Maximum residual　　b) Average residual

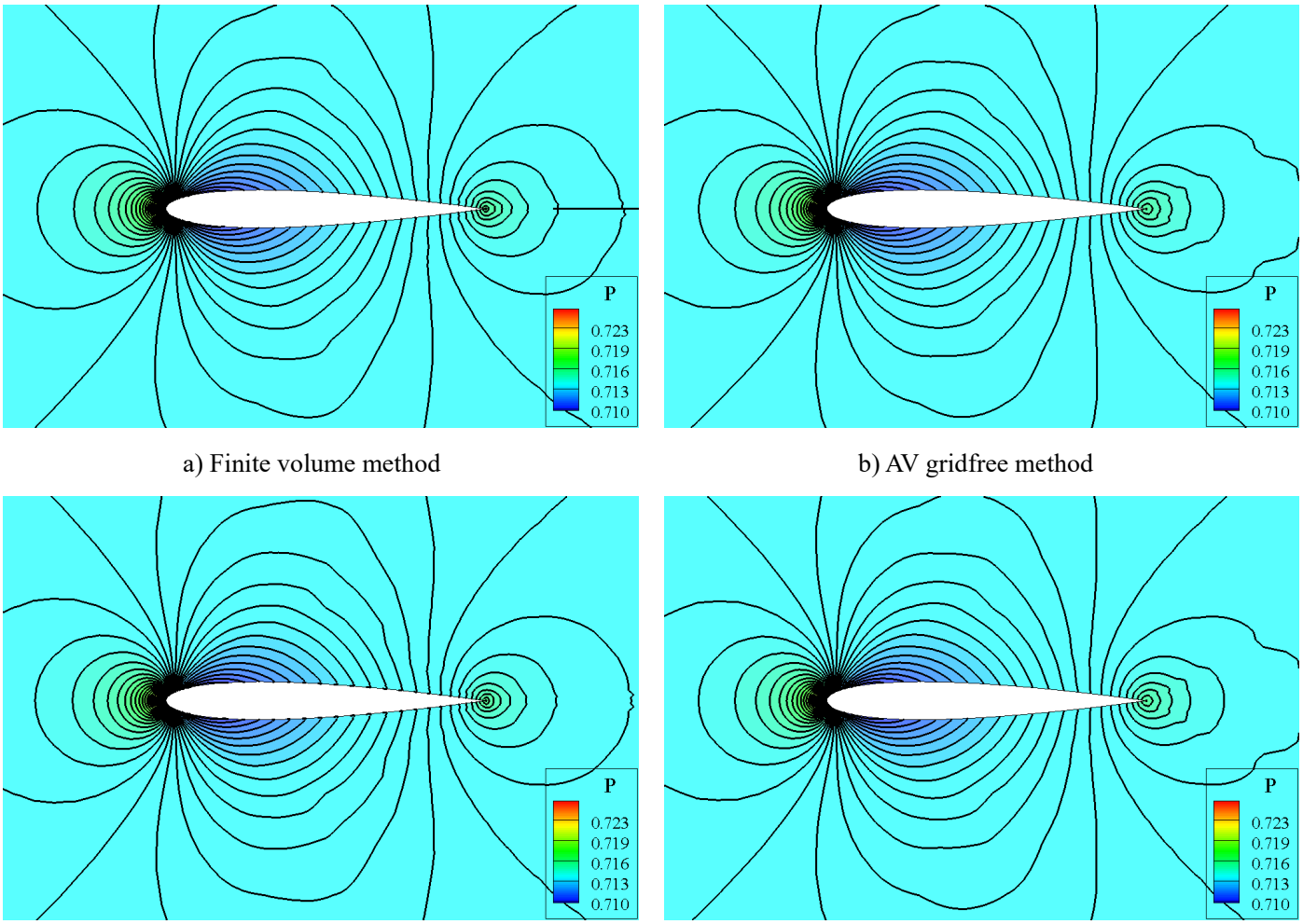
Figure 3　Convergence history of residuals, Ma=0.15, $\alpha=0^{o}$, $Re=6\times10^{6}$



a) Maximum residual

b) Average residual

Figure 4 Convergence history of residuals, Ma=0.775, α=2.05°, Re=10^7



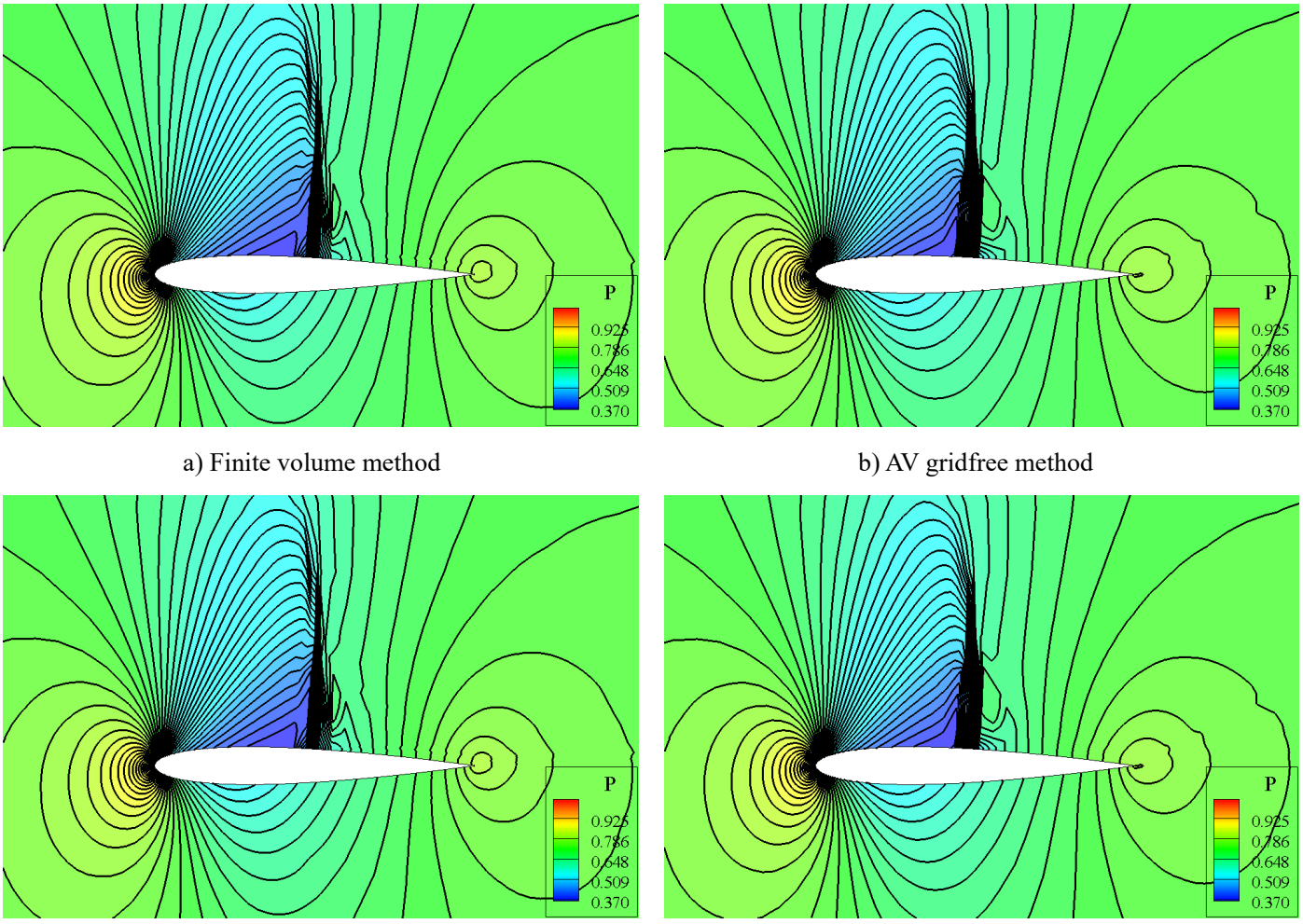
a) Finite volume method

b) AV gridfree method

c) General mesh method with 100% mesh & 0% point

d) General mesh method with 0% mesh & 100% point

Figure 5 Comparison of the obtained flowfields, Ma=0.15, α=0°, Re=6×10^6



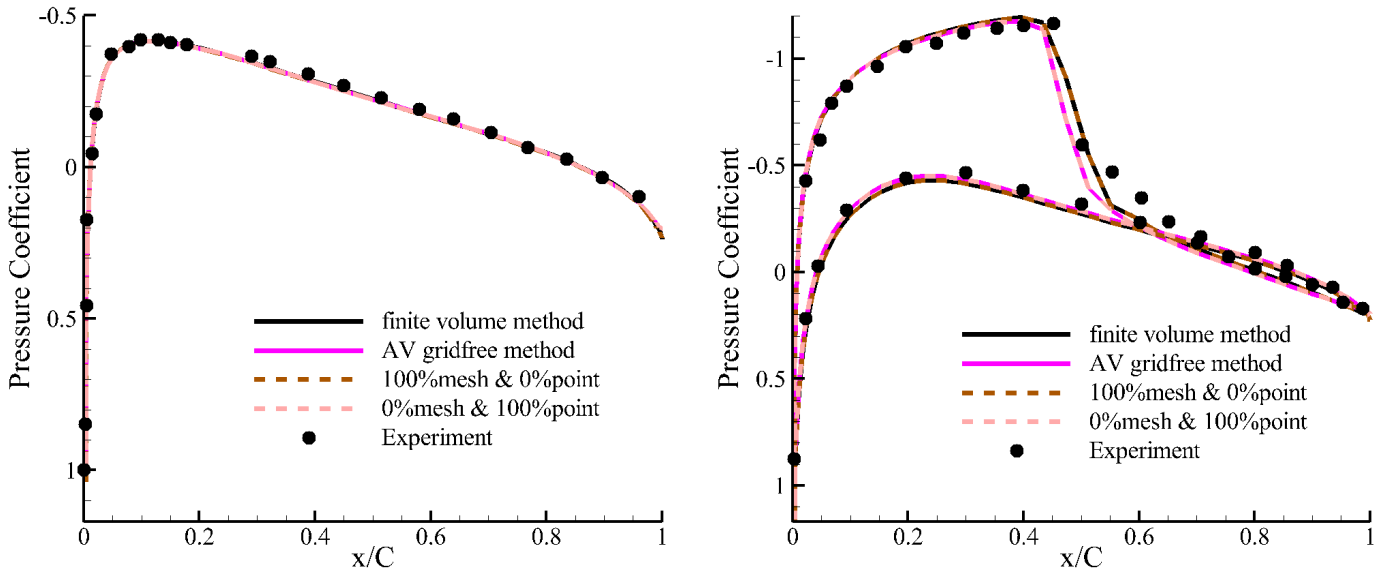
a) Finite volume method

b) AV gridfree method

c) General mesh method with 100% mesh & 0% point

d) General mesh method with 0% mesh & 100% point

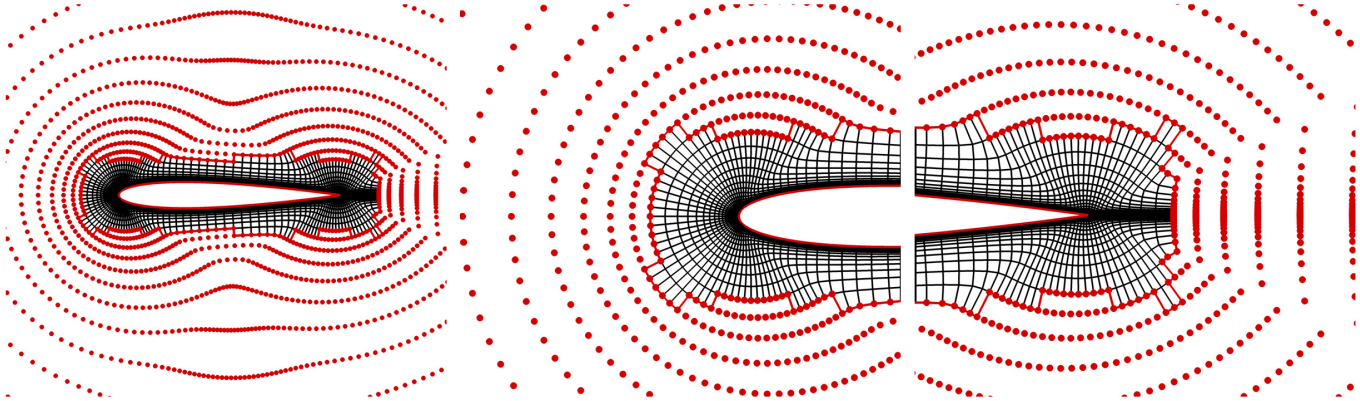
Figure 6 Comparison of the solved flowfields, Ma=0.775, α=2.05°, Re=10^7



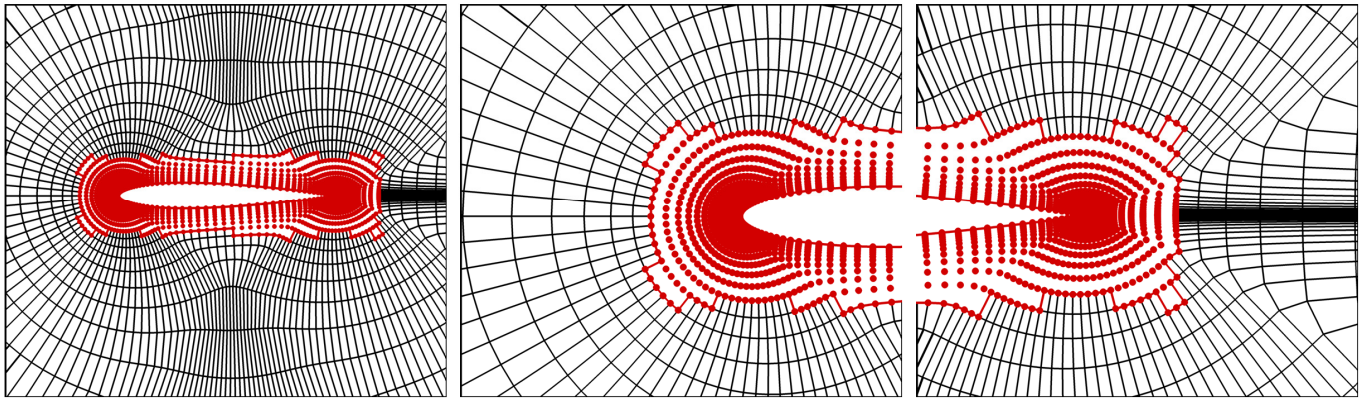
a) Ma=0.15, α=0°, Re=6×10^6

b) Ma=0.775, α=2.05°, Re=10^7

Figure 7 Comparison of surface pressure coefficients obtained by different methods

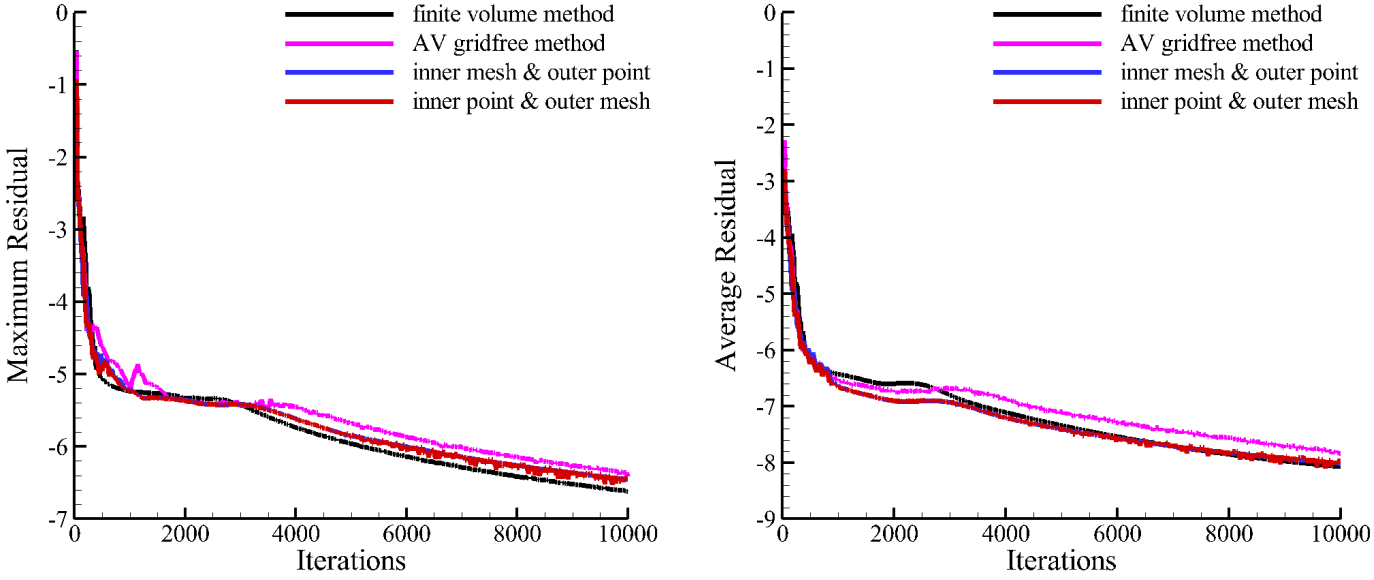


a) Zonal general mesh with inner mesh zone & outer point zone



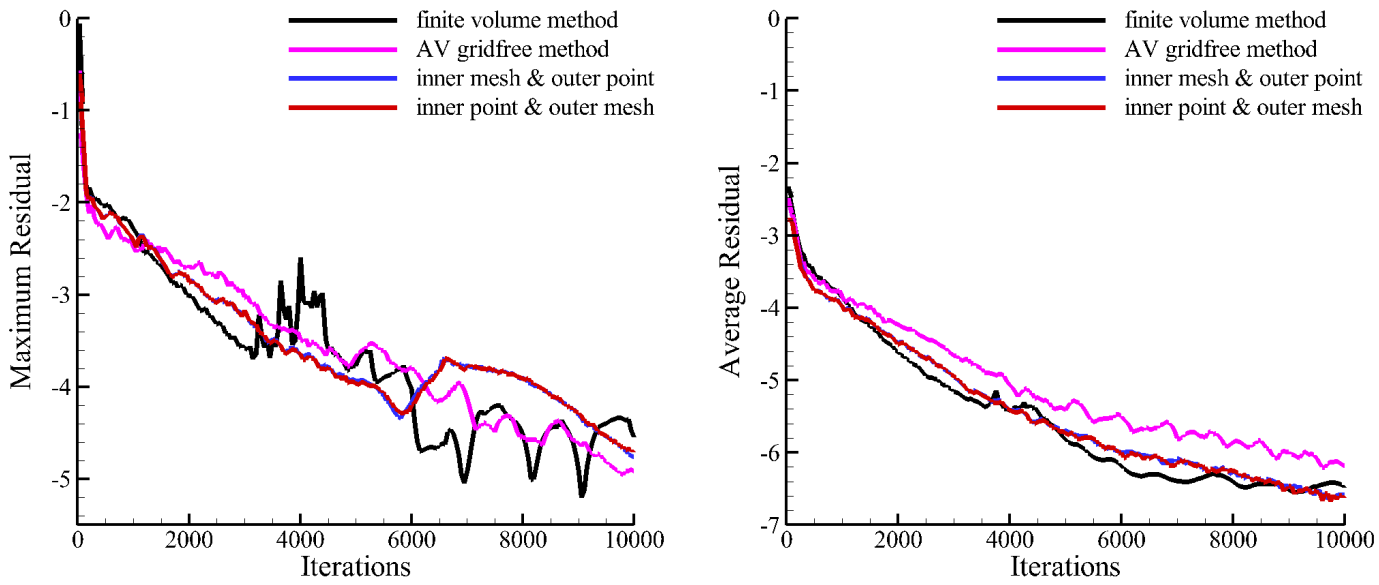
b) Zonal general mesh with inner point zone & outer mesh zone

Figure 8　Two zonal meshing methods for a NACA0012 airfoil



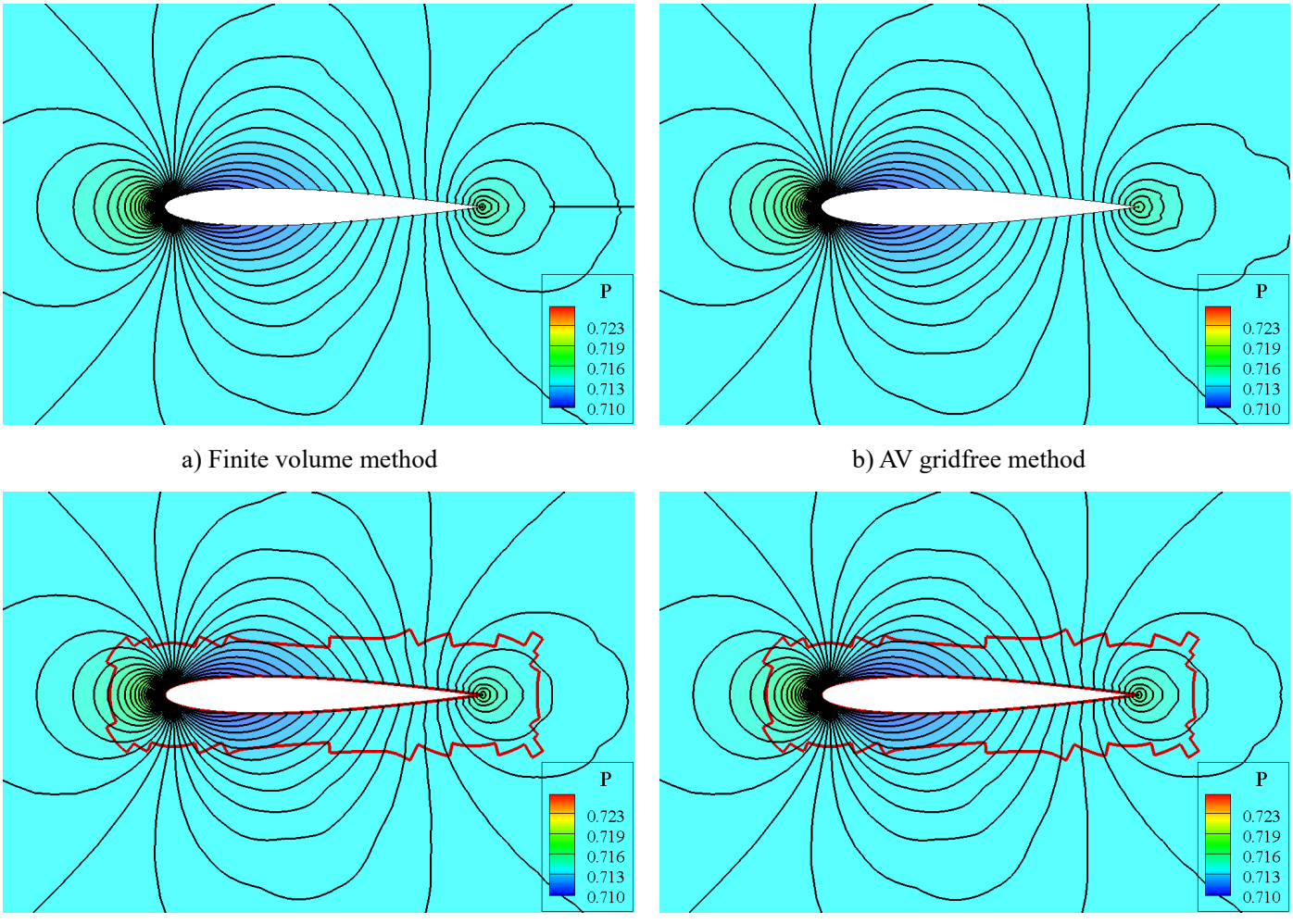
a) Maximum residual　　b) Average residual

Figure 9　Convergence history of residuals, Ma=0.15, $\alpha=0^{o}$, $Re=6\times10^{6}$



a) Maximum residual　　b) Average residual

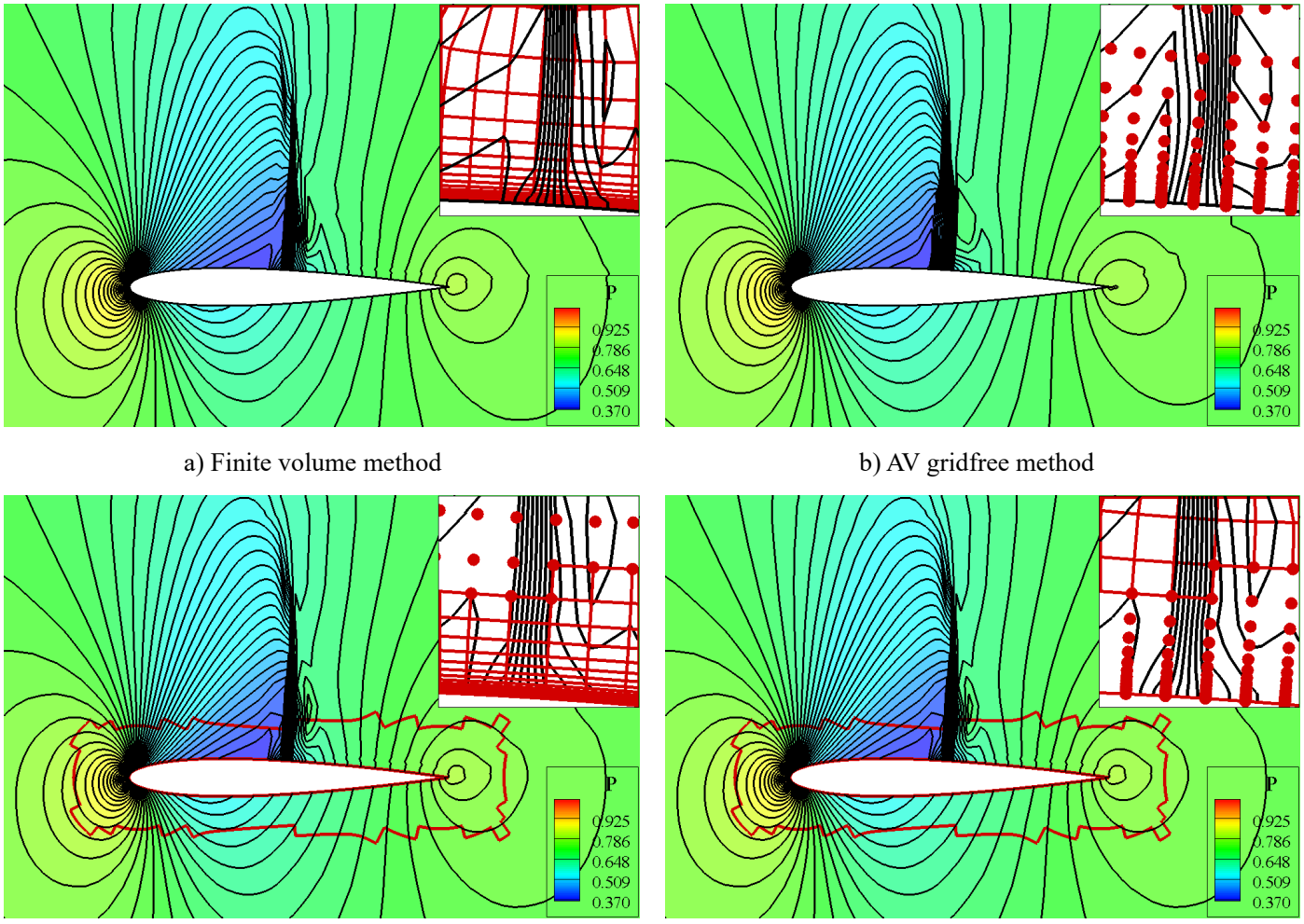
Figure 10　Convergence history of residuals, Ma=0.775, α=2.05°, Re=10^7



a) Finite volume method　　b) AV gridfree method

c) General mesh method with inner mesh & outer point　　d) General mesh method with inner point & outer mesh

Figure 11　Comparison of the obtained flowfields, Ma=0.15, α=0°, Re=6×10^6



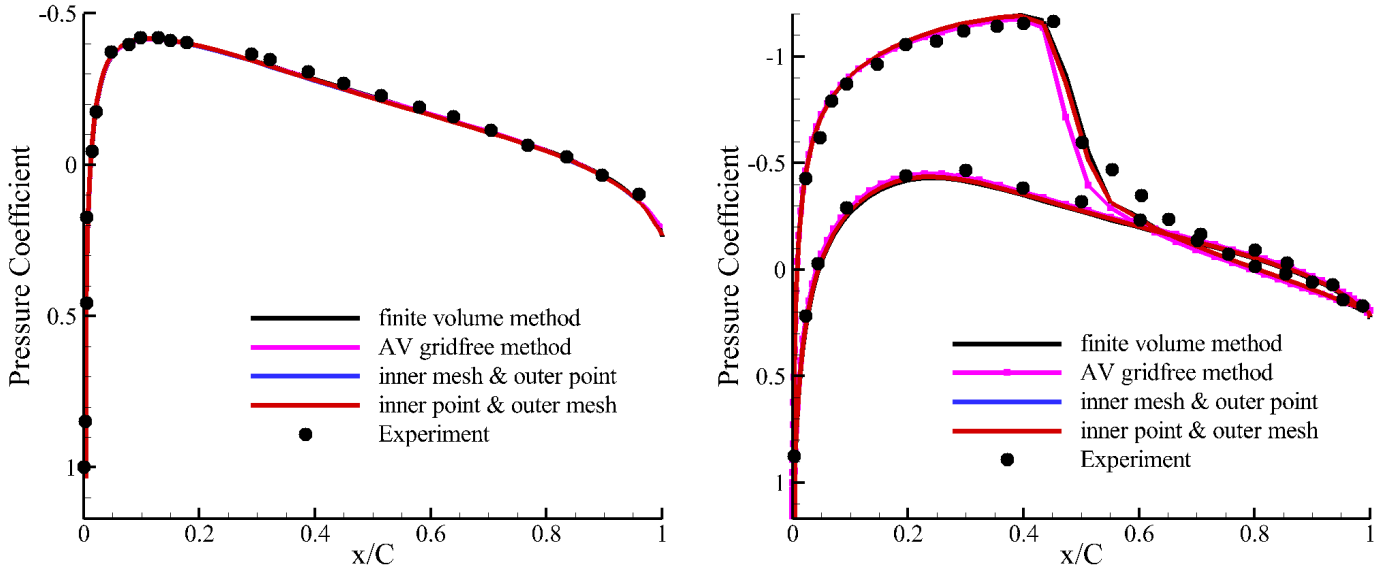
a) Finite volume method

b) AV gridfree method

c) General mesh method with inner mesh & outer point

d) General mesh method with inner point & outer mesh

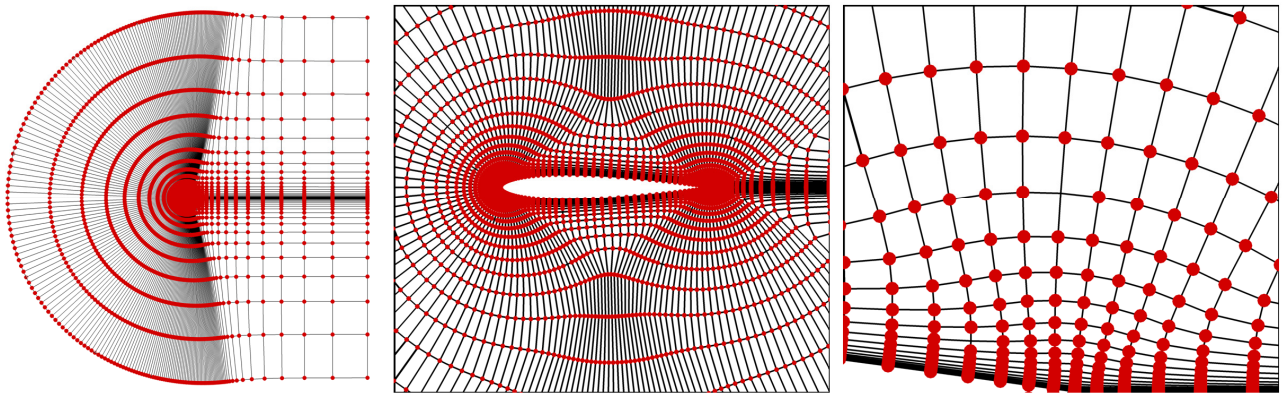
Figure 12 Comparison of the solved flowfields, Ma=0.775, α=2.05°, Re=10^7



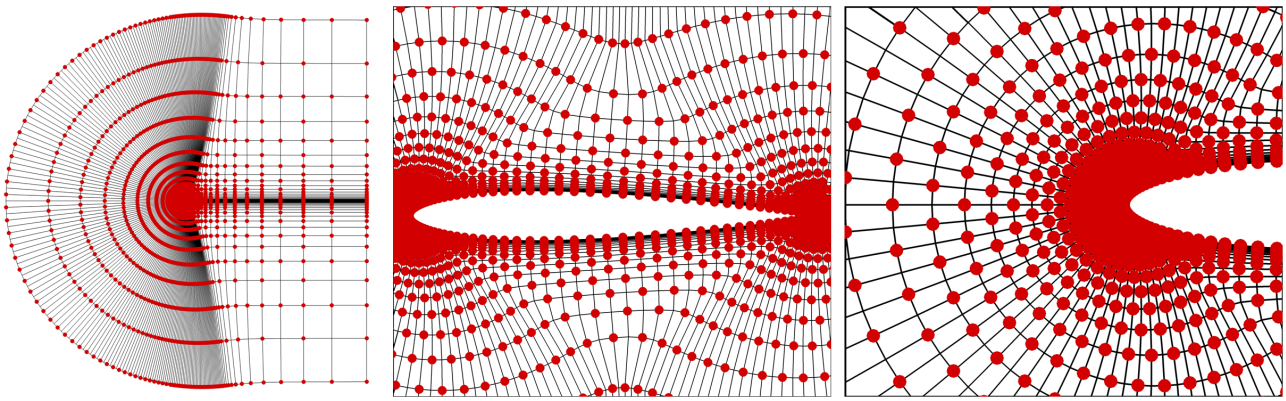
a) Ma=0.15, α=0°, Re=6×10^6

b) Ma=0.775, α=2.05°, Re=10^7

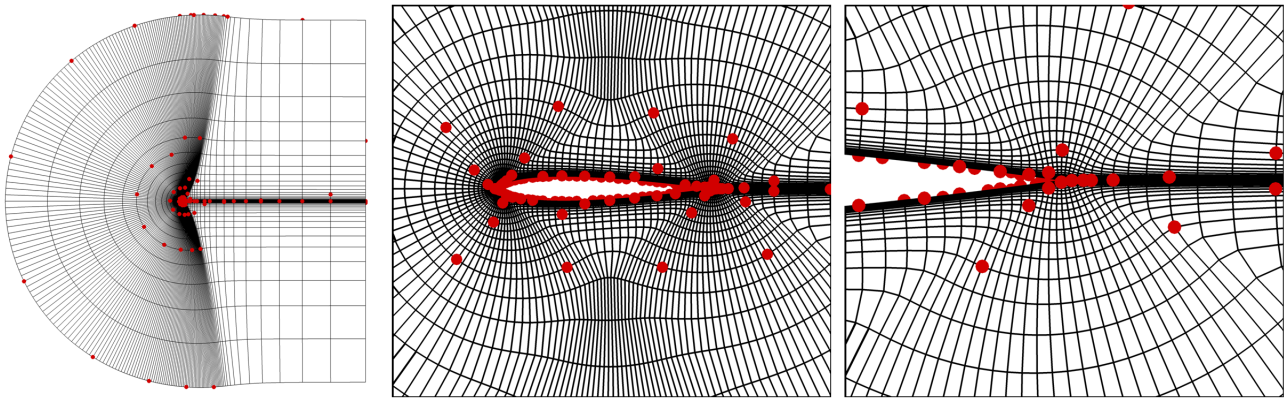
Figure 13 Comparison of surface pressure coefficients obtained by different methods



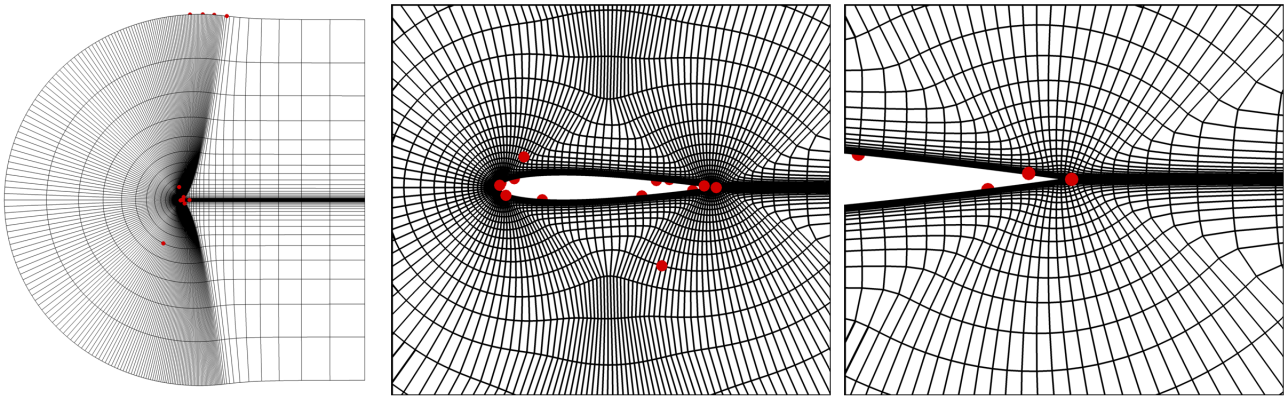
a) Fusion general mesh with 1% mesh & 99% point



b) Fusion general mesh with 50% mesh & 50% point

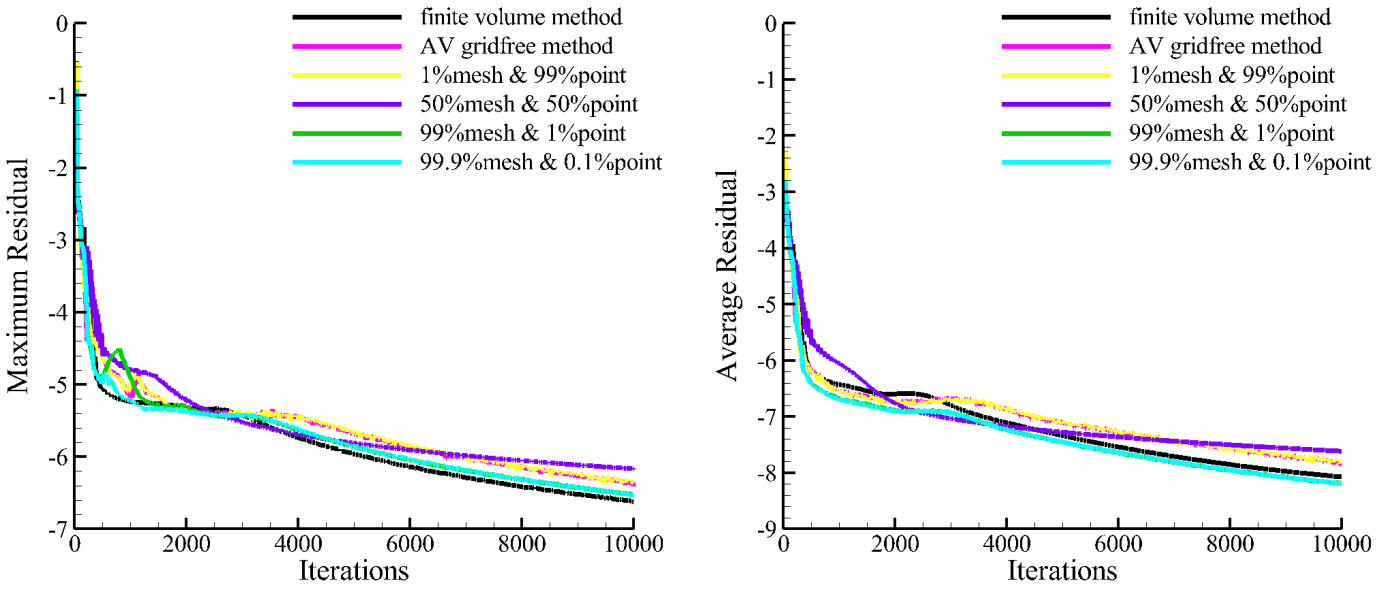


c) Fusion general mesh with 99% mesh & 1% point



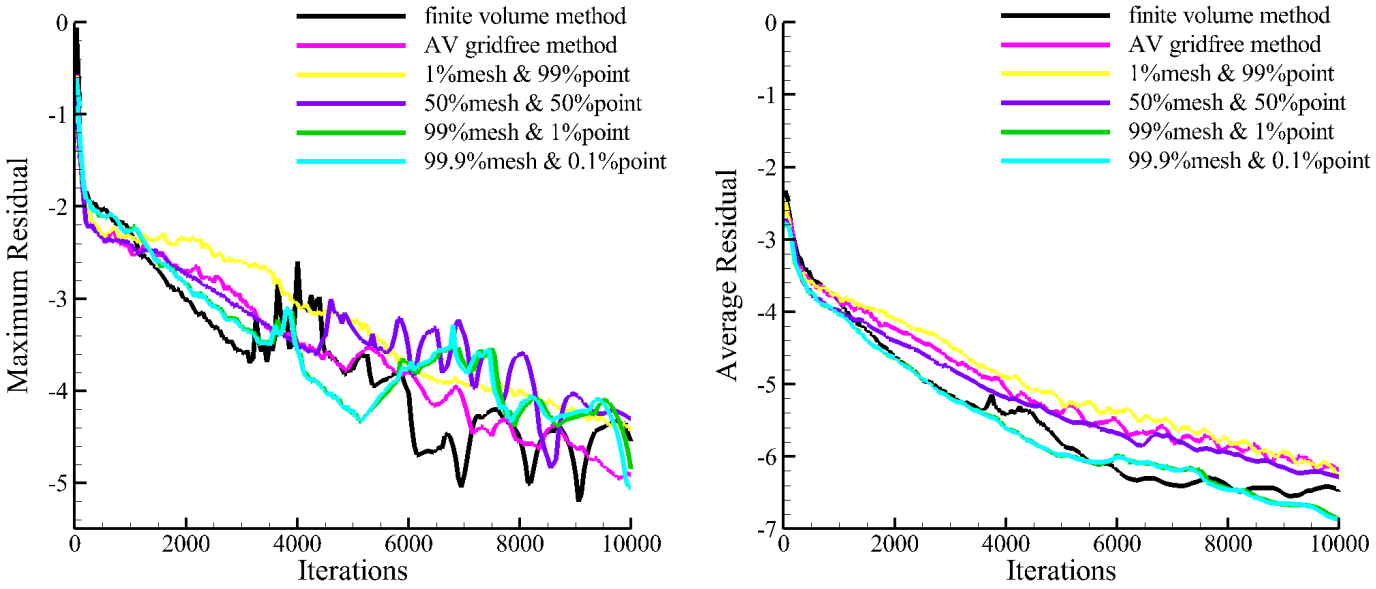
d) Fusion general mesh with 99.9% mesh & 0.1% point

Figure 14　Different fusion general meshing methods for a NACA0012 airfoil



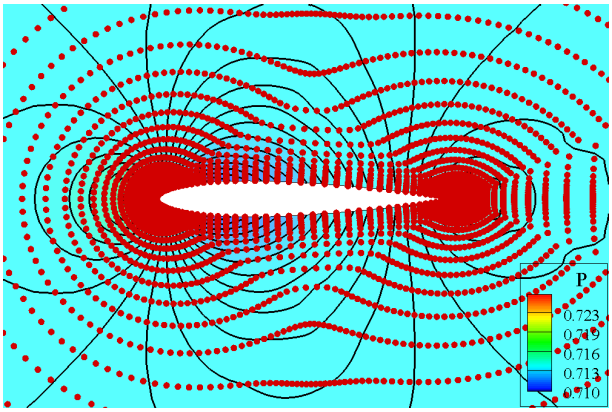
a) Maximum residual　　b) Average residual

Figure 15　Convergence history of residuals, Ma=0.15, α=0°, Re=6×10^6

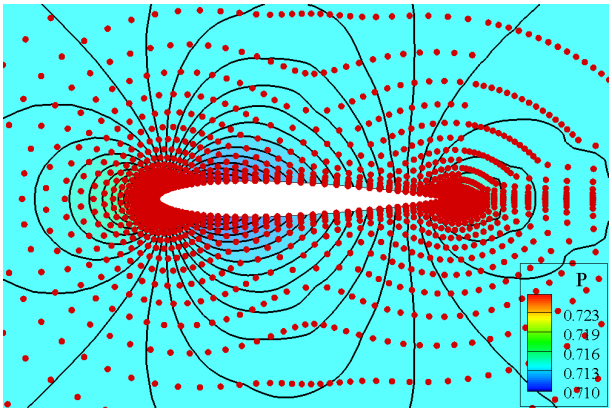


a) Maximum residual　　b) Average residual

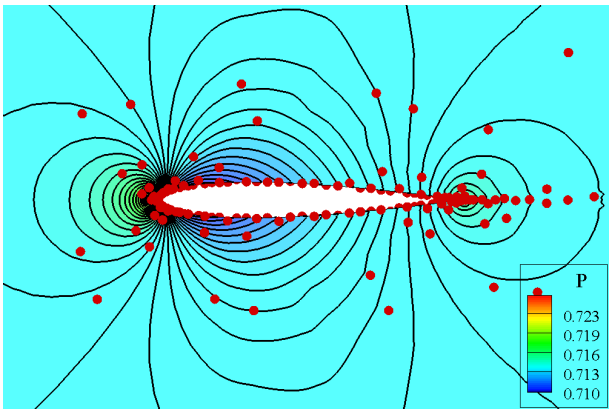
Figure 16　Convergence history of residuals, Ma=0.775, α=2.05°, Re=10^7



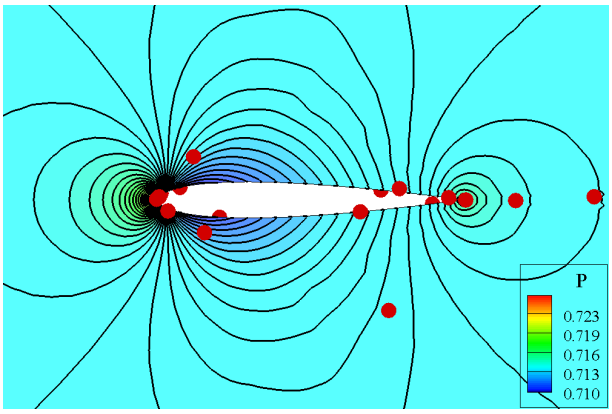
a) 1% mesh and 99% point



b) 50% mesh and 50% point

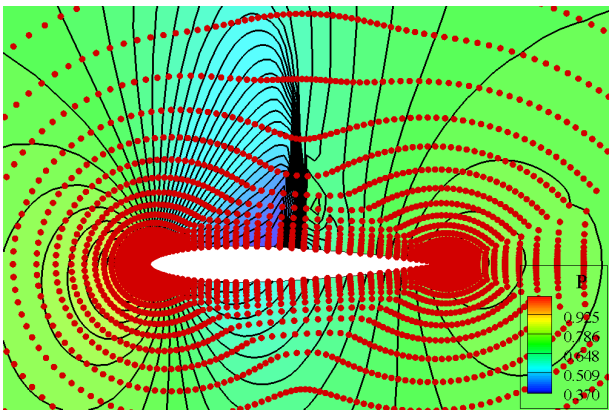


c) 99% mesh and 1% point

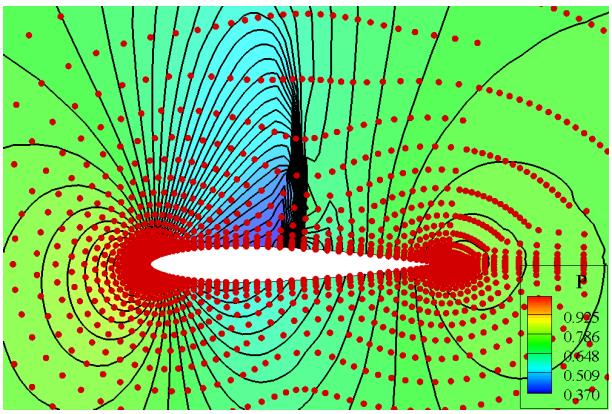


c) 99.9% mesh and 0.1% point

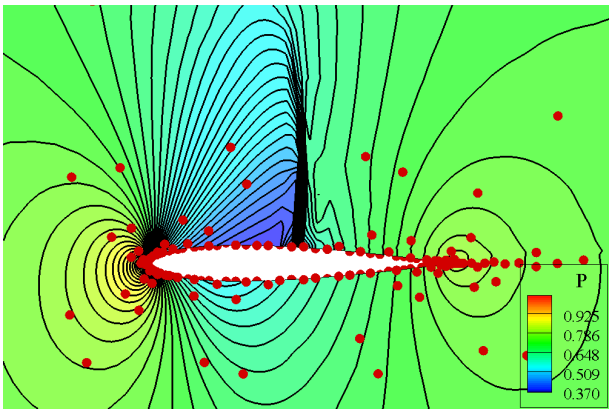
Figure 17 Comparison of flowfields obtained by general meshes, Ma=0.15, α=0°, Re=6×10^6



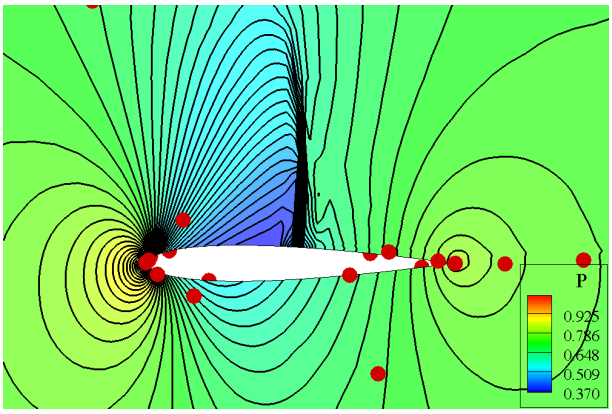
a) 1% mesh and 99% point



b) 50% mesh and 50% point

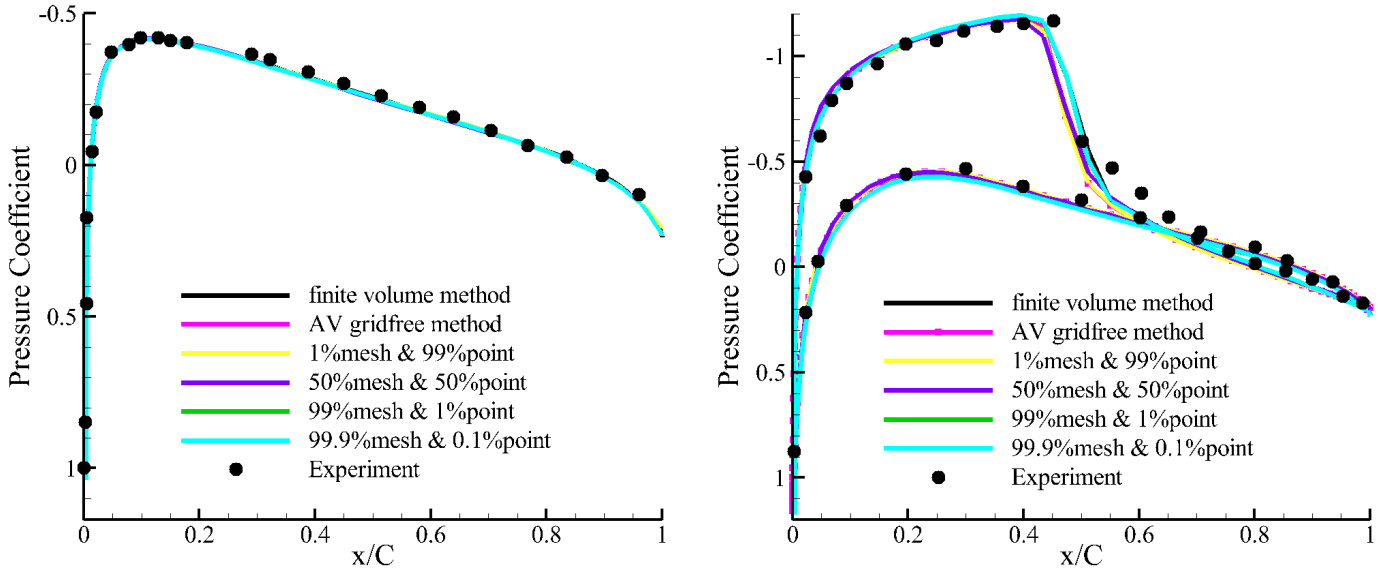


b) 99% mesh and 1% point



c) 99.9% mesh and 0.1% point

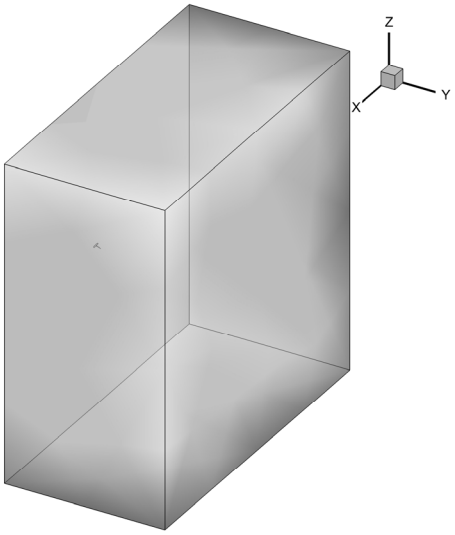
Figure 18 Comparison of flowfields obtained by general meshes, Ma=0.775, α=2.05°, Re=10^7



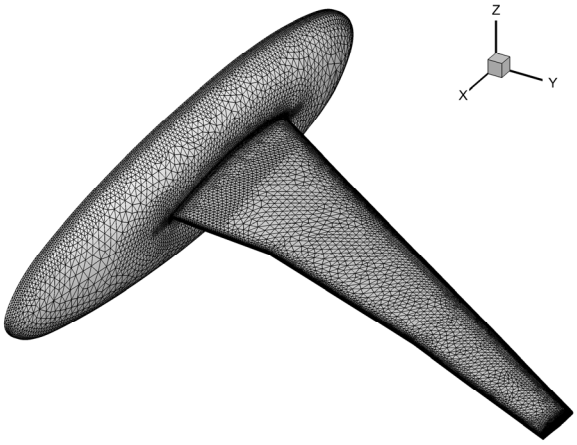
a) Ma=0.15, α=0°, Re=6×10^6

b) Ma=0.775, α=2.05°, Re=10^7

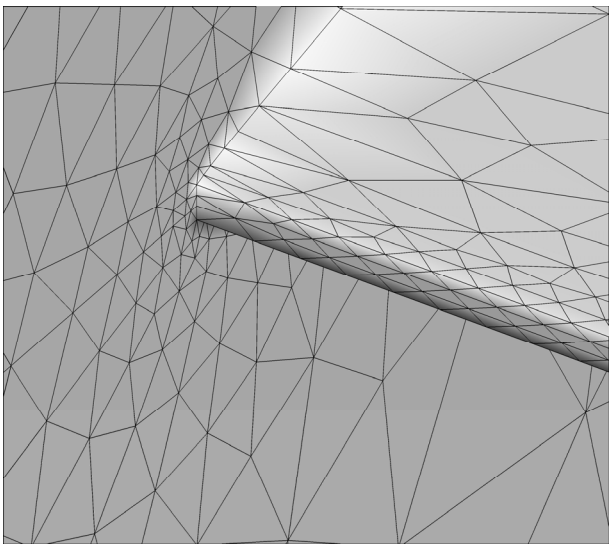
Figure 19 Comparison of surface pressure coefficients obtained by different methods



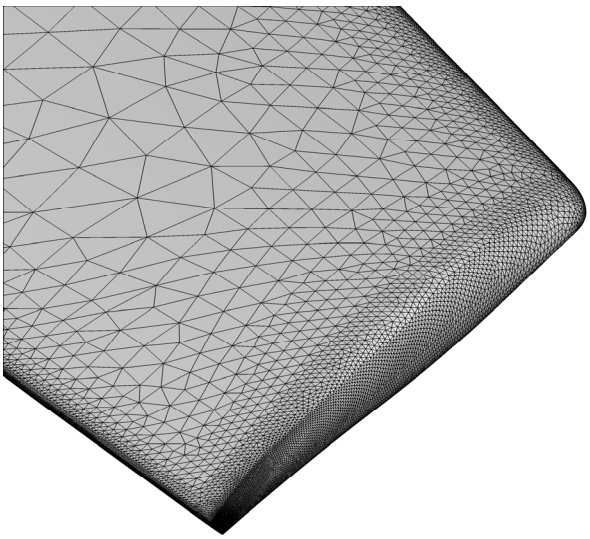
a) Computational domain



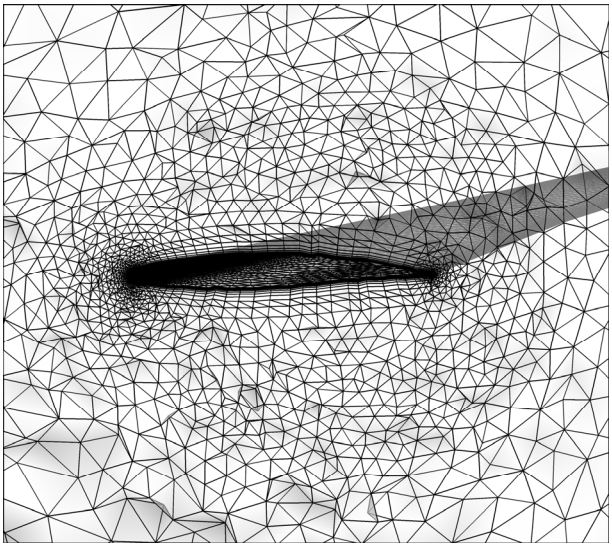
b) Surface grid



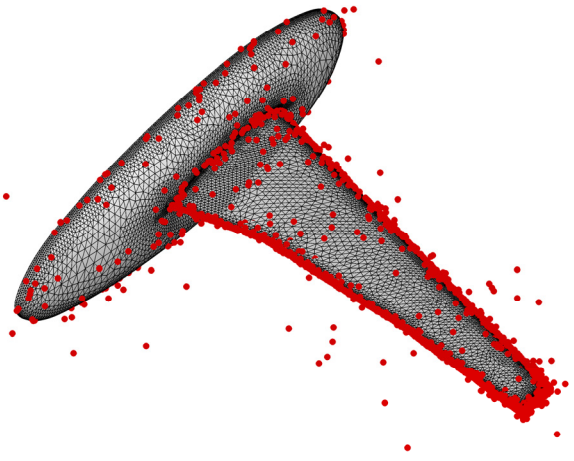
c) Close-up of wing/body junction



d) Close of wing tip grid

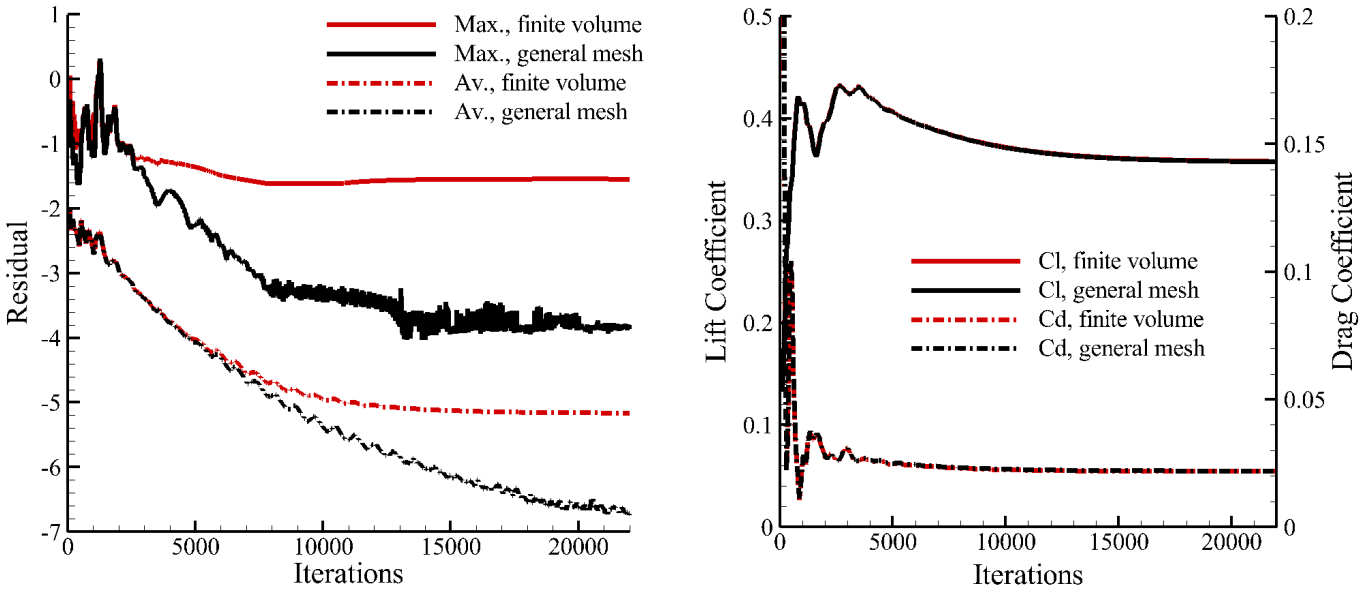


e) Volume mesh with prismatic boundary layer elements



f) Mesh nodes (red) converted into meshfree points

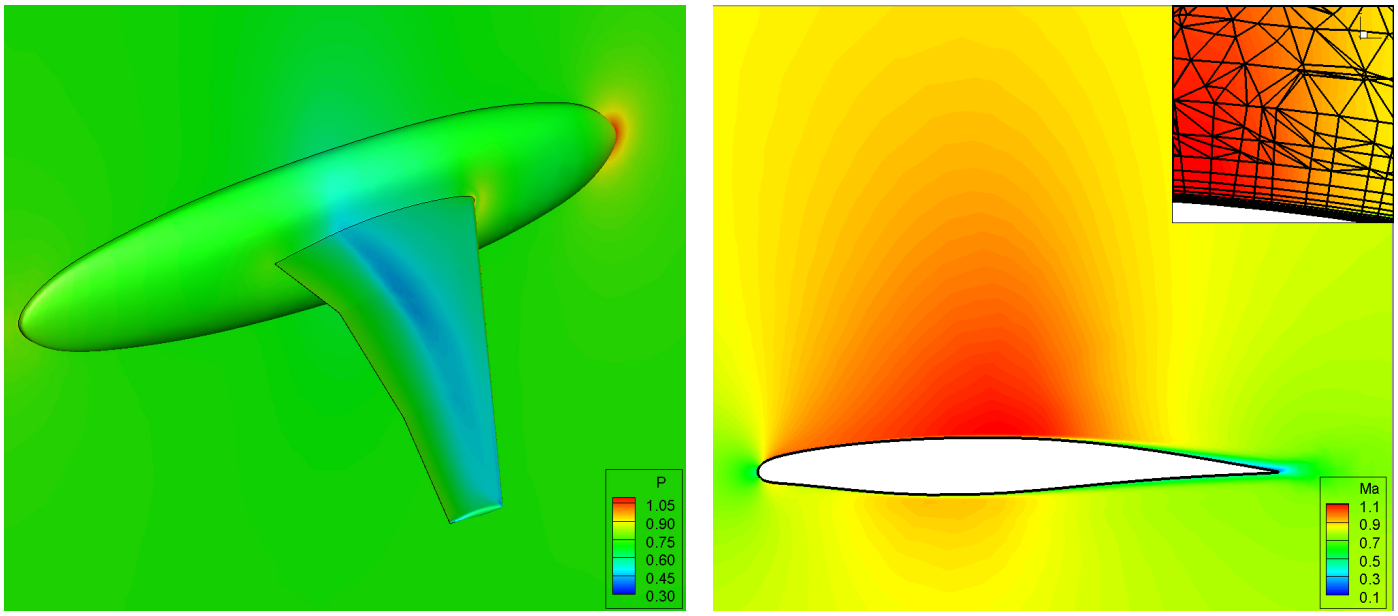
Figure 20 General mesh applied to improve the mesh quality for HIRENASD



a) Residual

b) Aerodynamic force

Figure 21 Comparison of convergence histories for transonic flow, Ma=0.8, α=1.5°, Re=7×10^6



a) Nondimensional surface pressure

b) Slice of flowfield (Mach number)

Figure 22 Flowfield with shock wave and boundary layer obtained by general mesh method, Ma=0.8, α=1.5°, Re=7×10^6

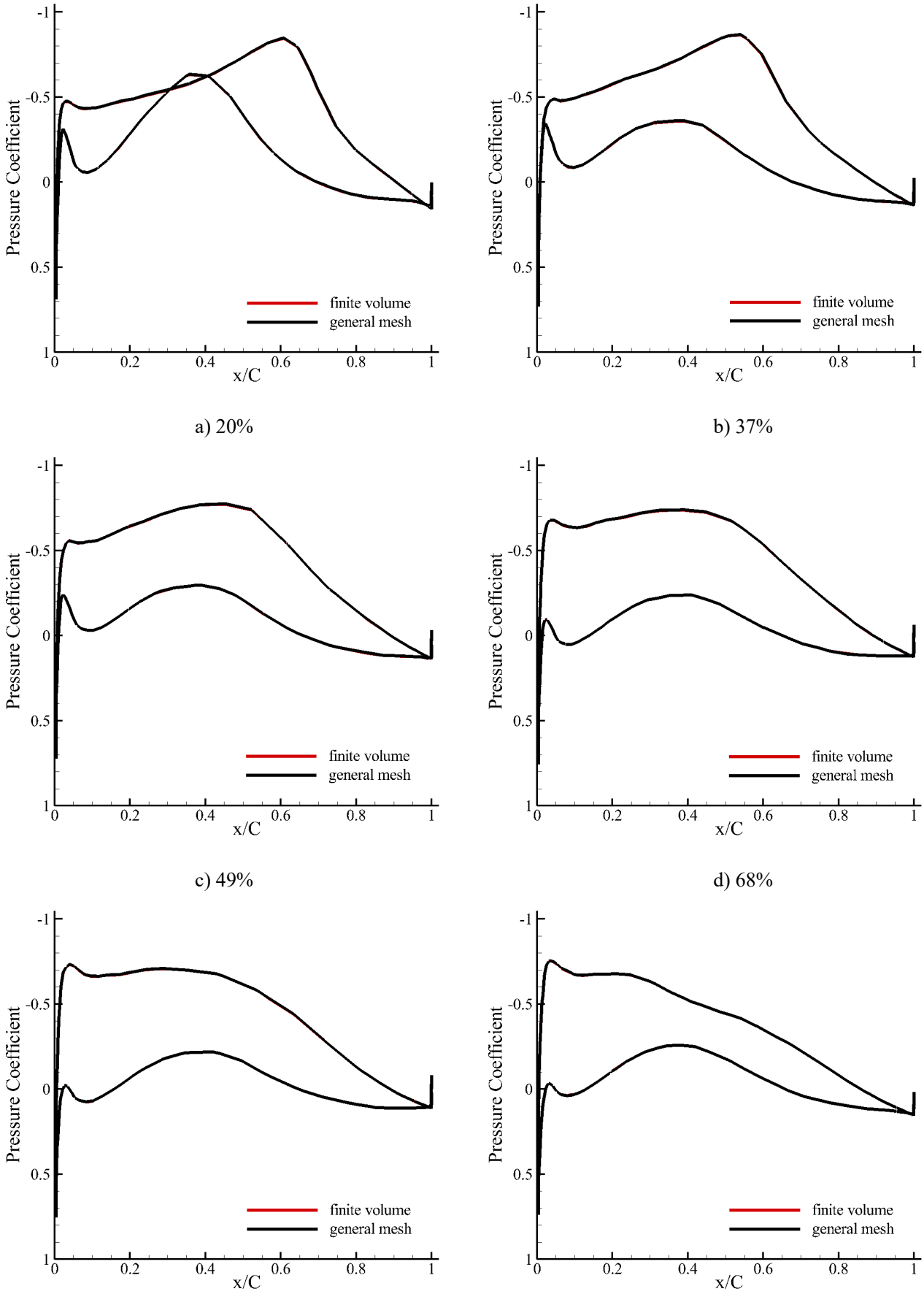


Figure 23 Comparison of pressure coefficient at different stations (normalized by local chord)