Algebraic-Volume Meshfree Method for Application in Finite Volume Solver

Yuewen Jiang

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

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

Abstract

Nowadays the finite volume (FV) method remains the most popular choice for Computational Fluid Dynamics (CFD). But few meshfree methods have been developed based on the framework of existing FV codes which can boost the development of the meshfree method. An algebraic volume (AV) concept for the meshfree method is proposed to mimic the geometric control volume (CV) of the FV method via a meshfree point cloud. Similar to the CV, each AV is closed by algebraic faces which are constructed via the weighted least-squares method. Only the bias fluxes are solved at the midpoint of each edge between the current point and its neighbour. The other flow variables of Navier-Stokes (NS) equations are solved in the same way as the traditional FV method. So it automatically solves the compressible flow with shock wave without additional treatment. The AV meshfree method only requires to modify the calculation method of volume and face area vector from a geometric way to an algebraic solution in pre-processing, therefore it is straightforward to implement in a traditional FV solver. The numerical methods of upwind schemes for convective flux, corrected central difference for viscous flux and implicit temporal discretization are identical for both the mesh-based FV method and AV meshfree method. The AV meshfree method has been integrated into an FV code and applied to high Reynolds number, transonic, viscous flow simulations. The accuracy and convergence of both the FV method and AV meshfree method are verified via 2D low-speed flow and transonic 3D flow. The results of AV meshfree method agree quite well with those of the FV method and experimental data. And the comparison shows good point size convergence. Furthermore, a demo of hybrid FV method and AV meshfree method is presented to investigate the potential

April 6, 2019

of this new meshfree method. The trajectory results obtained by CFD and experimental data are in good agreement.

Keywords: meshfree, finite volume (FV), algebraic volume (AV), control volume (CV), Navier-Stokes (NS) equations, Computational Fluid Dynamics (CFD)

Highlights:

- An algebraic volume (AV) meshfree method is proposed to mimic the control volume (CV) of finite volume (AV) via a point cloud;
- The AV meshfree method is merged into the framework of mesh-based FV method which is the most popular CFD method;
- The existing FV code only needs minor modifications in pre-processor to apply the AV meshfree method;
- The AV meshfree method is applied to simulate the high Reynolds viscous flows and obtains good agreement with both FV methods and experiments.

1. Introduction

Meshing is the foundation of the numerical discretization of a partial differential equation (PDE). A numerical method usually relies on the mesh type. So researchers have developed different types of numerical schemes which are usually constrained to the mesh types. For instance, the weighted essentially non-oscillatory (WENO) scheme is relatively straightforward to implement in a structured mesh but considerable effort is required to implement in an unstructured solver. Nowadays, the popular mesh types include three categories: structured mesh, unstructured mesh, and meshfree method. Compared to the former two, the meshfree method doesn't need the connection between nodes. It provides tremendous freedom to the meshing of complex geometry or domain (even with moving boundaries). But its numerical method is not yet mature in CFD application.

The structured mesh is one of the most important and popular meshing methods. It is widely used by researchers. Thompson proposed a mesh generation method which uses the solution of partial differential equation [1]. It can generate a good quality structured mesh. Eiseman reviewed the structured mesh generation methods in [2]. Nowadays, many technologies have been developed to keep this method active in CFD, such as multi-blocks method [3, 4], chimera method [5], etc. But it takes considerable effort to generate structured meshes for complex geometries because of the strict constraint of the mesh topology.

The unstructured mesh (including hybrid unstructured mesh) provides more freedom in the meshing and allows users to utilize a variety of element types to discretize the geometry. Mavriplis reviewed the unstructured mesh generation methods in [6]. Some of the latest meshing and numerical methods are discussed in [7–9]. The Cartesian mesh [10, 11] and arbitrary polyhedral (polygonal in 2D) mesh [12, 13] are the particular types of unstructured mesh. But it still struggles to generate a high-quality unstructured mesh on a complex geometry for viscous flow simulation. Meshing a narrow gap or conjunction configuration, etc. is still challenging, especially that with a moving boundary, such as might be encountered in insect flight.

The unstructured mesh breaks the shackles of the structured topology, and makes the mesh generation and its processing more flexible and convenient. Then the researchers [14, 15] developed the meshfree method which further breaks the mesh topological constraints. It doesn't need the connection between nodes to form control volumes. It exhibits greater geometric flexibility while dealing with complex geometry and moving boundaries. This advantage attracted many CFD researchers to contribute their effort on the development of the meshfree method.

Oate [16] proposed a stabilized finite point method for incompressible flow simulation. The shape functions were applied for approximation. The semi-implicit method based on an enhanced fractional step procedure was used. Zhang [17] presented a least-squares meshfree method for the two-dimensional (2D) flow. Gauss quadrature was used in the background cells constructed by the quadtree algorithm and the boundary conditions were enforced by the penalty method. Kim et al [18] developed a collocation method for 2D incompressible flow. The derivatives of governing equations were obtained by the linear combination of the shape functions which were constructed by the moving least square approximation. Sanyasiraju and Chandhini [19] proposed a local radial basis function (RBF) based mesh-

free scheme for unsteady incompressible flow. The velocity and pressure were decoupled. Borthakur and Biswas [20] presented a Hermite Taylor least square finite difference method for the meshfree method. An adaptive upwind scheme was utilized to handle the convective flux by modifying the stencil in the upstream direction. Dehghan and Abbaszadeh [21] proposed a variational multiscale element free Galerkin meshfree method which was based on the shape functions of moving Kriging interpolation. Proper orthogonal decomposition method was used to improve efficiency. This method was further developed in [22–24]. The 2D incompressible flow was solved in a primitive variables approach. Foy et al [25] developed a meshfree finite volume method and applied to multi-phase porous media models. The conservative, strong-form partial differential equation was used. The spline interpolation was employed to approximate the fluxes on the edges of nodal control volumes. One key feature of incompressible flow is the continuous flowfield which is convenient to reconstruct the flow solution.

Hietel et al [26] derived a finite volume particle method to model the moving particles. The smoothed particle hydrodynamics (SPH) method was used to solve the Euler equation. Sridar and Balakrishnan [27] presented an upwind finite difference scheme for the meshfree method on 2D inviscid flow. The results were compared to those obtained by the finite volume method. It was found that the computational effort of the meshfree method was comparable with that of cell vertex finite volume method. Wang [28] studied the point clouds generation and meshfree numerical method for unsteady flow simulation. The spatial derivatives used to estimate the inviscid flux were directly approximated by using the local least-squares method. Harish [29] used the meshfree Euler solver to study the store separation. The entropy variables based least-squares kinetic upwind method was used. The meshfree point clouds were generated separately for the wing and store via chimera cloud of points. Katz and Jameson [30] proposed an edge-based meshfree method for compressible inviscid flow simulations. The mesh-based reconstruction, diffusion, and convergence acceleration schemes were applied to the meshfree method. Wang and Periaux [31] developed a meshfree method with artificial dissipation. It was used on a 2D Euler equations which were solved by the explicit five-stage Runge-Kutta scheme. Ma et al [32] implemented a meshfree dynamic cloud method on the graphic processing units (GPUs) to achieve high performance. The spatial derivatives of Euler equation were discretised by the moving-least square scheme on each cloud of points. The progress of meshfree method for inviscid flow simulation was reviewed in [33].

The isotropic point distribution is usually employed in the decomposition of the inviscid flow domain. For one point cloud, the distance between points is similar at different directions. It is relatively easy to design numerical schemes.

Gunther et al [34] presented a multi-scale meshfree method based on the Petrov-Galerkin formulation with streamline upwind Petrov-Galerkin (SUPG) and shock capturing for viscous, compressible flows. The multiresolution analysis was used to determine the location of shocks. Munikrishna and Balakrishnan [35] proposed a meshfree method of which discretization strategy was based on positivity of a discrete Laplacian. The method was applied to simulate the viscous flow around 2D airfoils. Kennett et al [36] developed an implicit meshfree method to solve 2D Euler and Navier-Stokes equations. The spatial derivatives were approximated by a least squares method on point cloud. Namvar and Jahangirian [37] investigated the meshfree method for compressible turbulent flows. The capabilities of the Taylor least square method for calculation of spatial derivatives were evaluated. Ortega et al [38] applied the finite point method to high-Reynolds number compressible flow problems. In the boundary layer, the construction of numerical approximation in the highly stretched clouds of points was studied.

On the one hand, the meshfree method has the advantages on the meshing of complex computational domains, especially those with moving boundaries. On the other hand, the drawback of the meshfree method is obvious. It is neither well developed nor widely applied. Since the meshfree method is quite different from the mesh-based method, these meshfree methods [16–38] have been developed either specifically for meshfree applications, or incompatibly with traditional FV flow solver. They are not easy to implement in the FV framework. However, the majority of the current CFD codes utilize the FV method and are already mature, especially in industry. It has successfully solved the high Reynolds number, transonic flow which is challenging for the meshfree method because of the high aspect-ratio point distribution in the boundary layer and the discontinuousness of flowfield. If the mature finite volume code is abandoned, it will waste a lot of research resources and take a long time to develop a new, mature meshfree code.

The best way is to find an efficient numerical scheme for the meshfree method to maximize the use of existing FV code. However, the FV discretization is based on the control volume (CV) method. Since the mesh is abandoned in the meshfree method, it is not easy to construct the geometric control volume. In order to make great use of the existing FV codes,

an algebraic volume (AV) concept is proposed for the meshfree method. It employs a similar idea as FV method of arbitrary polyhedral mesh to derive discretization scheme for the meshfree method. The algebraic volume is constructed by the weighted least-squares method. It can be implemented in mesh-based FV solver with minor changes to the code.

2. Finite Volume Method based on Arbitrary Polyhedron

The integral form of Reynolds Averaged Navier-Stokes (RANS) 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 volume and boundary of discrete control volume, respectively. $\mathbf{n}$ is the unit outward-normal vector of the control volume face $\partial\Omega$. V is the volume (area in 2D) of Ω. S is the area (length in 2D) of $\partial\Omega$. The conservative flow variable vector $\mathbf{Q}$, inviscid flux vector $\mathbf{F}(\mathbf{Q})$, and viscous flux vector $\mathbf{F}^{vis}(\mathbf{Q})$ are

$$\mathbf{Q}=\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ \rho e_0\end{bmatrix} \tag{2}$$

$$\mathbf{F}(\mathbf{Q})\cdot\mathbf{n}=\begin{bmatrix}\rho\mathbf{v}\\ \rho u\mathbf{v}\cdot\mathbf{n}+Pn_x\\ \rho v\mathbf{v}\cdot\mathbf{n}+Pn_y\\ \rho w\mathbf{v}\cdot\mathbf{n}+Pn_z\\ \rho e_0\mathbf{v}\cdot\mathbf{n}+P(\mathbf{v}\cdot\mathbf{n})\end{bmatrix} \tag{3}$$

$$\mathbf{F}^{vis}(\mathbf{Q})\cdot\mathbf{n}=\begin{bmatrix}0\\ \tau_{xx}n_x+\tau_{xy}n_y+\tau_{xz}n_z\\ \tau_{yx}n_x+\tau_{yy}n_y+\tau_{yz}n_z\\ \tau_{zx}n_x+\tau_{zy}n_y+\tau_{zz}n_z\\ \sigma_x n_x+\sigma_y n_y+\sigma_z n_z\end{bmatrix} \tag{4}$$

respectively. ρ, P and e_0 denote the density, static pressure, and specific total energy per unit mass, respectively. $\mathbf{v}$ is the vector of velocity. u, v, w

are the velocity components of vector $\mathbf{v}$. n_x, n_y, n_z are the components of normal vector $\mathbf{n}$. The Reynolds stress tensor reads:

$$\begin{cases} \tau_{xx} = 2(\mu+\mu_t)\frac{\partial u}{\partial x} - \frac{2}{3}(\mu+\mu_t)\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \tau_{yy} = 2(\mu+\mu_t)\frac{\partial v}{\partial y} - \frac{2}{3}(\mu+\mu_t)\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \tau_{zz} = 2(\mu+\mu_t)\frac{\partial w}{\partial z} - \frac{2}{3}(\mu+\mu_t)\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \tau_{xy} = \tau_{yx} = (\mu+\mu_t)\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \\ \tau_{xz} = \tau_{zx} = (\mu+\mu_t)\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ \tau_{yz} = \tau_{zy} = (\mu+\mu_t)\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \end{cases} \tag{5}$$

And $\sigma_x, \sigma_y, \sigma_z$ are:

$$\begin{cases} \sigma_x = u\tau_{xx} + v\tau_{xy} + w\tau_{xz} + \kappa\frac{\partial T}{\partial x} \\ \sigma_y = u\tau_{yx} + v\tau_{yy} + w\tau_{yz} + \kappa\frac{\partial T}{\partial y} \\ \sigma_z = u\tau_{zx} + v\tau_{zy} + w\tau_{zz} + \kappa\frac{\partial T}{\partial z} \end{cases} \tag{6}$$

where κ is the thermal conductivity coefficient

$$\kappa = \frac{\gamma R}{\gamma - 1}\left(\frac{\mu}{P_r} + \frac{\mu_t}{P_{rt}}\right) \tag{7}$$

where P_r and P_{rt} are the laminar and turbulent Prandtl numbers and their values are 0.72 and 0.9 respectively. T is the static temperature. γ is the ratio of specific heat coefficient. R is the specific gas constant. μ and μ_t represent the molecular viscosity and turbulent viscosity, respectively. The former is calculated by Sutherland's law. The extra turbulent model is required to obtain the turbulent eddy viscosity μ_t, such as Spalart-Allmaras (SA) [39], $k-\omega$ SST [40], etc.

The finite volume method for spatial discretization allows one to handle the arbitrary polyhedral control volume. It is assumed that a mesh for the computational domain has been generated. Figure 1 shows a 2D example of control volumes used for meshing. The flow variables are stored in the center of control volumes. For arbitrary polygonal control volume m formed by the blue lines, the discretization of each term in RANS Equation (1) can be expressed in second-order spatial discretization scheme as follows:

$$\begin{cases} \frac{\partial}{\partial t}\iiint_{\Omega,m}\mathbf{Q}dV \approx \frac{\mathrm{d}(\mathbf{Q}_m V_m)}{\mathrm{d}t} \\ \mathbf{R}(\mathbf{Q}_m) = \iint_{\partial\Omega}\mathbf{F}(\mathbf{Q})\cdot\mathbf{n}dS \approx \sum_{p\in\Xi(m)}\mathbf{F}(\mathbf{Q}_{m,p})\cdot\Delta\mathbf{S}_{m,p} \\ \mathbf{R}^{vis}(\mathbf{Q}_m) = \iint_{\partial\Omega}\mathbf{F}^{vis}(\mathbf{Q})\cdot\mathbf{n}dS \approx \sum_{p\in\Xi(m)}\mathbf{F}^{vis}(\mathbf{Q}_{m,p})\cdot\Delta\mathbf{S}_{m,p} \end{cases} \tag{8}$$

where $\mathbf{R}(\mathbf{Q}_m)$ and $\mathbf{R}^{vis}(\mathbf{Q}_m)$ denote the inviscid and viscous flux vectors of control volume m, respectively. $\Xi(m)$ is the set of the faces of m. V_m is the volume of the control volume. $\Delta\mathbf{S}_{m,p}$ is the area vector of p-th face, $\Delta\mathbf{S}_{m,p} = \mathbf{n}_{m,p}\Delta S_{m,p}$. $\Delta S_{m,p}$ is the area of face. $\mathbf{Q}_m$ is the vector of conservative flow variables at the centroid of control volume, representing the averaged value of control volume m. $\mathbf{Q}_{m,p}$ is the vector of reconstructed flow variables at the centroid of its p-th face, representing the averaged value of face. In order to achieve second-order precision, the reconstruction scheme must be at least first order. The piecewise linear reconstruction is employed frequently.

After spatial discretization (Equation (8)), the governing partial differential Equation (1) is reduced to an ordinary differential equation. The semi-discrete form for control volume m can be written as follows:

$$\frac{\mathrm{d}(\mathbf{Q}_m V_m)}{\mathrm{d}t} + \sum_{p\in\Xi(m)}\mathbf{F}(\mathbf{Q}_{m,p})\cdot\Delta\mathbf{S}_{m,p} = \sum_{p\in\Xi(m)}\mathbf{F}^{vis}(\mathbf{Q}_{m,p})\cdot\Delta\mathbf{S}_{m,p} \tag{9}$$

or

$$\frac{\mathrm{d}(\mathbf{Q}_m V_m)}{\mathrm{d}t} = -[\mathbf{R}(\mathbf{Q}_m) - \mathbf{R}^{vis}(\mathbf{Q}_m)] \tag{10}$$

For steady simulation or pseudo time marching, the volume V_m remains the same. So we can convert Equation (9) to

$$\frac{\mathrm{d}\mathbf{Q}_m}{\mathrm{d}t} + \sum_{p\in\Xi(m)}\mathbf{F}(\mathbf{Q}_{m,p})\cdot\Delta\mathbf{S}'_{m,p} = \sum_{p\in\Xi(m)}\mathbf{F}^{vis}(\mathbf{Q}_{m,p})\cdot\Delta\mathbf{S}'_{m,p} \tag{11}$$

where $\mathbf{S}'_{m,p} = \mathbf{S}_{m,p}/V_m$.

3. Construction of Algebraic Volume via Point Cloud

The differential form of RANS equations can be written as follows:

$$\frac{\partial \mathbf{Q}}{\partial t} + \left(\frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{F}}{\partial z} \right) = \left(\frac{\partial \mathbf{F}^{vis}}{\partial x} + \frac{\partial \mathbf{F}^{vis}}{\partial y} + \frac{\partial \mathbf{F}^{vis}}{\partial z} \right) \tag{12}$$

Compared to Equation (11), we can find that the face area vectors of the control volume are equivalent to a transformer that converts the face fluxes into a derivative (differential form). The face area vector is a key role for the finite volume method. For any control volume, we also have the geometric property

$$\sum_{p \in \Xi(m)} \Delta \mathbf{S}'_{m,p} = \mathbf{0} \tag{13}$$

This is a fundamental geometric identity defining the geometric control volume as a closed set. Violation of Equation (13) signals problems in the construction of the control volume.

In order to solve the RANS equations, the meshfree point set that contributes to the current point is called a point cloud, shown in Figure 2. m is the current meshfree point. Assuming that the point cloud including all its contributors is represented by set $\tilde{\Xi}(m)$, which owns points $n, n_1, n_2, n_3, n_4, n_5, n_6, n_7$ in this case. It is similar to the neighbours in the arbitrary polyhedral control volume method. They are surrounding the point m. Point n is its p-th neighbor. A simple connection has been established by the point cloud. It should be noted that this so-called connection is quite weak compared to the control volume method. A node is added to the point cloud set $\tilde{\Xi}(m)$, usually because of the angle and distance.

As shown in Equation (11), the key characteristics of the finite volume method is the control volume which has closed geometric faces. On this control volume, the volume integral of the differential equation is converted into area integral by the Green-Gauss theorem, which is the bridge to connect the derivative values and face fluxes. However, we only have points rather than mesh cells in space when the meshfree method is used to decompose the flow domain. So we don't have the mesh to construct the control volume. In order to use the same code as mesh-based finite volume solver, the meshfree method needs to solve a classic question:

How to construct a so-called control volume, so-called face area vector, and so-called face flux?

The only information we have is the set of point cloud. Suppose that the flow domain is decomposed by meshfree points. For a meshfree point m, its point cloud set is $\tilde{\Xi}(m)$. The total number of neighbours is N_m. And $N_m > d+1$, where d is the number of dimensions. The coordinate of point m is (x_m, y_m, z_m). The midpoint of edge between m and its p-th neighbor n is $(x_{m,p}, y_{m,p}, z_{m,p})$ or vector $\mathbf{x}_{m,p}$

$$\mathbf{x}_{m,p} = \frac{1}{2}(\mathbf{x}_m + \mathbf{x}_n) \tag{14}$$

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

$$\begin{aligned}\mathbf{F}_{m,p} = \mathbf{F}_m + \frac{\partial \mathbf{F}_m}{\partial x}(x_{m,p} - x_m) + \frac{\partial \mathbf{F}_m}{\partial y}(y_{m,p} - y_m) + \frac{\partial \mathbf{F}_m}{\partial z}(z_{m,p} - z_m) \\ + o(\Delta x, \Delta y, \Delta z)^2, \quad p \in \tilde{\Xi}(m)\end{aligned} \tag{15}$$

For a viscous flow simulation, the high aspect ratio point distribution is usually used near the viscous wall. A weight ω is added to both sides of the equation to enhance the numerical accuracy in the solving of the inverse matrix. Neglecting the high order part $o(\Delta x, \Delta y, \Delta z)^2$, yields

$$\begin{aligned}&\omega_{m,p}(x_{m,p} - x_m)\frac{\partial \mathbf{F}_m}{\partial x} + \omega_{m,p}(y_{m,p} - y_m)\frac{\partial \mathbf{F}_m}{\partial y} + \omega_{m,p}(z_{m,p} - z_m)\frac{\partial \mathbf{F}_m}{\partial z} + \omega_{m,p}\mathbf{F}_m \\ &= \omega_{m,p}\mathbf{F}_{m,p}, \quad p \in \tilde{\Xi}(m)\end{aligned} \tag{16}$$

The inverse of distance is utilized as the weight ω in the paper. For N_m neighbors of the meshfree point m , we get the linear equation as follows:

$$\mathbf{A}_m \mathbf{B}_m = \mathbf{C}_m \tag{17}$$

where

$$\mathbf{A}_m = \begin{bmatrix} \omega_{m,1}(x_{m,1} - x_m) & \omega_{m,1}(y_{m,1} - y_m) & \omega_{m,1}(z_{m,1} - z_m) & \omega_{m,1} \\ \vdots & \vdots & \vdots & \vdots \\ \omega_{m,p}(x_{m,p} - x_m) & \omega_{m,p}(y_{m,p} - y_m) & \omega_{m,p}(z_{m,p} - z_m) & \omega_{m,p} \\ \vdots & \vdots & \vdots & \vdots \\ \omega_{m,N_m}(x_{m,N_m} - x_m) & \omega_{m,N_m}(y_{m,N_m} - y_m) & \omega_{m,N_m}(z_{m,N_m} - z_m) & \omega_{m,N_m} \end{bmatrix} \tag{18}$$

$$\mathbf{B}_m = \begin{bmatrix} \frac{\partial \mathbf{F}_m}{\partial x} & \frac{\partial \mathbf{F}_m}{\partial y} & \frac{\partial \mathbf{F}_m}{\partial z} & \mathbf{F}_m \end{bmatrix}^T \tag{19}$$

$$\mathbf{C}_m = \begin{bmatrix} \omega_{m,1}\mathbf{F}_{m,1} & \dots & \omega_{m,p}\mathbf{F}_{m,p} & \dots & \omega_{m,N_m}\mathbf{F}_{m,N_m} \end{bmatrix}^T \tag{20}$$

The superscript T represents the transpose.

The equation is solved by

$$\mathbf{A}_m^T \mathbf{A}_m \mathbf{B}_m = \mathbf{A}_m^T \mathbf{C}_m \tag{21}$$

It is the least squares solution of Equation (17). If matrix $\mathbf{A}_m^T \mathbf{A}_m$ is singular, it can be avoided via moving point m. Otherwise, we obtain the solution as follows:

$$\mathbf{B}_m = [(\mathbf{A}_m^T \mathbf{A}_m)^{-1} \mathbf{A}_m^T] \mathbf{C}_m \tag{22}$$

or

$$\mathbf{B}_m^i = \sum_{p \in \tilde{\Xi}(m)} C_{m,p} \Delta \tilde{S}_{m,p}^i \tag{23}$$

where superscript i represents the i-th row of the vector or matrix. $\Delta \tilde{S}_{m,p}^i$ is the p-th element of the i-th row of the matrix $[(\mathbf{A}_m^T \mathbf{A}_m)^{-1} \mathbf{A}_m^T]$. So we obtain the derivatives of flux as follows:

$$\begin{cases} \frac{\partial \mathbf{F}_m}{\partial x} = \sum_{p \in \tilde{\Xi}(m)} \omega_{m,p} \mathbf{F}_{m,p} \Delta \tilde{S}_{m,p}^1 \\ \frac{\partial \mathbf{F}_m}{\partial y} = \sum_{p \in \tilde{\Xi}(m)} \omega_{m,p} \mathbf{F}_{m,p} \Delta \tilde{S}_{m,p}^2 \\ \frac{\partial \mathbf{F}_m}{\partial z} = \sum_{p \in \tilde{\Xi}(m)} \omega_{m,p} \mathbf{F}_{m,p} \Delta \tilde{S}_{m,p}^3 \end{cases} \tag{24}$$

It should be noted that the fourth row of vector $\mathbf{B}_m$, namely, the solution of $\mathbf{F}_m$, is neglected. It is not used in the discretization. Therefore, we only need to save three elements which relate to the geometric information of point cloud. Let

$$\Delta\ \tilde{\mathbf{S}}_{m,p} = \omega_{m,p} \begin{bmatrix} \Delta \tilde{S}_{m,p}^1 & \Delta \tilde{S}_{m,p}^2 & \Delta \tilde{S}_{m,p}^3 \end{bmatrix}^T \tag{25}$$

We can obtain the summation of derivatives of inviscid flux as follows:

$$\frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{F}}{\partial y} + \frac{\partial \mathbf{F}}{\partial z} = \sum_{p \in \tilde{\Xi}(m)} \mathbf{F}_{m,p} \cdot \Delta \tilde{\mathbf{S}}_{m,p} \tag{26}$$

Moreover, supposing that the flow is uniform, yields

$$\sum_{p\in\tilde{\Xi}(m)} \Delta\tilde{\mathbf{S}}_{m,p} = \mathbf{0} \tag{27}$$

Namely, we obtain the similar conservation property as the finite volume method (Equation (13)). The summation of the vectors is equal to 0.

Therefore, we can call the vector $\Delta\tilde{\mathbf{S}}_{m,p}$ as algebraic area vector, as shown in the black dashed arrow of Figure 2. It is solved by the algebraic equations rather than the geometric outward-normal face vector that is shown in the black solid arrow of Figure 1. Moreover, the summation of algebraic area vectors is also 0. Therefore, the algebraic volume (AV) is also closed by algebraic faces, shown in the blue dashed curve of Figure 2. It is different from the geometric control volume which is shown in the blue solid line segments of Figure 1. Its volume is constant 1.

Similarly, we can get the summation of derivatives of viscous flux. Therefore, the discretization of the differential form of the RANS Equation (12) by algebraic-volume meshfree method yields:

$$\frac{\mathrm{d}\mathbf{Q}_m}{\mathrm{d}t} + \sum_{p\in\tilde{\Xi}(m)} \mathbf{F}(\mathbf{Q}_{m,p})\cdot\Delta\tilde{\mathbf{S}}_{m,p} = \sum_{p\in\tilde{\Xi}(m)} \mathbf{F}^{vis}(\mathbf{Q}_{m,p})\cdot\Delta\tilde{\mathbf{S}}_{m,p} \tag{28}$$

It can be seen that there are no essential differences between the discretization form of finite volume (Equation (11)) and that of algebraic-volume meshfree (Equation (28)) methods. The only minor differences lie in the following: Firstly, the volume of the meshfree algebraic volume is constant 1 while the volume of the control volume of FV method is calculated by the geometric formula; secondly, the calculation method of the face area vector is different. The meshfree method calculates the algebraic area vector via the weighted least squares matrix operation rather than the geometric area vector in finite volume method. Otherwise, the spatial discretization, time integration, and turbulence modelling of the arbitrary topology mesh method and code do not need to be modified. They are suitable for the meshfree method as well. Therefore, this considerably reduces the effort for researchers to implement the AV meshfree method in the numerous popular FV codes.

Therefore, a special control volume is defined for meshfree method via the point cloud: each meshfree point m corresponds to an algebraic volume $\tilde{m}$, and the meshfree point m is located in the center of $\tilde{m}$; the set of point cloud $\tilde{\Xi}(m)$ records the neighbours of $\tilde{m}$. The algebraic volume is formed

by algebraic faces. The face center is the midpoint of an edge between the current point and its neighbour. Each algebraic face has an area vector, which is calculated by the matrix $[(\mathbf{A}_m^T\mathbf{A}_m)^{-1}\mathbf{A}_m^T]$ rather than the geometric area vector $\Delta\mathbf{S}_{m,p}$ in Figure 1. The volume of each algebraic volume is always 1.

From now on, the meshfree method can use the same discretization scheme as the arbitrary topology mesh. They can share the same flow solver. For the sake of narrative simplicity, the following variable description only adopts the symbology used in the mesh method. The algebraic volume is the same as control volume if it is not specially mentioned, so as to the algebraic face.

4. Flux Calculation and Boundary Conditions

The fluxes are evaluated at the face center. The inviscid flux is solved by

$$\mathbf{F}(\mathbf{Q}_{m,p}) \cdot \Delta\mathbf{S}_{m,p} = \mathbf{F}^c(\mathbf{Q}_{m,p}^L, \mathbf{Q}_{m,p}^R) \cdot \Delta\mathbf{S}_{m,p} \tag{29}$$

where $\mathbf{F}^c$ is the scheme for convective flux. The flux-vector splitting (AUSM [41]) and flux-difference splitting (Roe's [42]) are used in the paper.$\mathbf{Q}_{m,p}^L$ and $\mathbf{Q}_{m,p}^R$ are the values of left and right hand sides of the face, respectively. It is assumed that the solution is piecewise linearly distributed in each control volume. So the values of the left- and right-hand sides of the face are interpolated from the neighbours

$$\begin{cases} q_{m,p}^L = q_m + \phi_m[\nabla q_m \cdot (\mathbf{x}_{m,p} - \mathbf{x}_m)] \\ q_{m,p}^R = q_n + \phi_n[\nabla q_n \cdot (\mathbf{x}_{m,p} - \mathbf{x}_n)] \end{cases} \tag{30}$$

where ∇q is the flow gradient, obtained by Green-Gauss method. $\mathbf{x}_{m,p}$ is the coordinate vector of face center. $\mathbf{x}_m$ is the coordinate vector of the centroid of control volume m. Subscript n is the p-th neighbor of control volume m. ϕ is the limiter function. Its value should tend to be 1 in the smooth region and 0 in the discontinuous region. The principle of the limiter is that the constructed interpolation polynomial (30) can't bring new extremum. Two popular limiters are adopted: Barth and Jespersen [43] and Venkatakrishnan [44] limiters.

The viscous flux is solved by the central difference method. The spatial derivative of the flow velocity and temperature on the face is calculated by the averaging method. Since the flow gradient at the center of the control

volume has been obtained at the calculation of inviscid flux, it is natural to calculate the gradient at the face center by averaging the gradient values.

$$\nabla q_{m,p} = \frac{1}{2}(\nabla q_m + \nabla q_n) \tag{31}$$

As mentioned in [45], this stencil may introduce the decoupling of the solution on quadrilateral or hexahedral cells. So the weighted averaged method [46] is used.

For the inviscid simulation, the wall uses no penetration boundary condition and the pressure is obtained using the extrapolation method. For the viscous wall, it applies a no-slip boundary condition and the pressure gradient is 0 in the normal direction. The temperature at the boundary generally needs to be provided by the user. Or it is decided by temperature gradient at normal direction $\nabla T \cdot \mathbf{n} = 0$ for adiabatic wall boundary. The density is obtained by solving the equation of state.

For the symmetry boundary, the velocity normal to the symmetry plane is 0, and the other quantities are the same as those at the center of the control volume.

In the far field, the Riemann non-reflecting boundary is adopted.

5. Implicit Time-Marching Method

The semi-discrete form of RANS Equation (10) can be written as follows:

$$\frac{\mathrm{d}(\mathbf{Q}_m V_m)}{\mathrm{d}t} = -\bar{\mathbf{R}}(\mathbf{Q}_m) = -[\mathbf{R}(\mathbf{Q}_m) - \mathbf{R}^{vis}(\mathbf{Q}_m)] \tag{32}$$

where $\bar{\mathbf{R}}$ is the residual of RANS equation. $\mathbf{R}$ and $\mathbf{R}^{vis}$ represent the inviscid and viscous flux terms, respectively. Implicit discretization of the residual term of above equation yields:

$$V_m \frac{\Delta \mathbf{Q}_m^{k+1}}{\Delta t} = -\bar{\mathbf{R}}(\mathbf{Q}_m^{k+1}) \tag{33}$$

where k is the number of the time step. Δt is the size of the time step and

$$\Delta \mathbf{Q}_m^{k+1} = \mathbf{Q}_m^{k+1} - \mathbf{Q}_m^k \tag{34}$$

Since the residual value at the next time step $(k+1)$ cannot be obtained directly, the implicitly expressed residual term $\bar{\mathbf{R}}$ is time linearized by

$$\bar{\mathbf{R}}(\mathbf{Q}_m^{k+1}) \approx \bar{\mathbf{R}}(\mathbf{Q}_m^k) + \mathbf{A}_m^k \Delta \mathbf{Q}_m^{k+1} \tag{35}$$

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

We know that the residual value is the summation of face fluxes

$$\bar{\mathbf{R}}(\mathbf{Q}_m^{k+1}) \approx \bar{\mathbf{R}}(\mathbf{Q}_m^k) + \sum_{p\in\Xi(m)} \left(\frac{\partial \mathbf{F}_{m,p}^k}{\partial \mathbf{Q}} - \frac{\partial \mathbf{F}_{m,p}^{vis,k}}{\partial \mathbf{Q}} \right) \cdot \Delta \mathbf{S}_{m,p} \Delta \mathbf{Q}_{m,p}^{k+1} \tag{36}$$

Compared to Equation (35), we obtain the Jacobian matrix as follows:

$$\begin{cases} \mathbf{A}_m^k = \mathbf{A}_m^{inv,k} + \mathbf{A}_m^{vis,k} \\ \mathbf{A}_m^{inv,k} = \sum_{p\in\Xi(m)} \frac{\partial \mathbf{F}_{m,p}^k}{\partial \mathbf{Q}} \cdot \Delta \mathbf{S}_{m,p} \\ \mathbf{A}_m^{vis,k} = \sum_{p\in\Xi(m)} \frac{\partial \mathbf{F}_{m,p}^{vis,k}}{\partial \mathbf{Q}} \cdot \Delta \mathbf{S}_{m,p} \end{cases} \tag{37}$$

Substituting Equation (35) and Equation (37) formulas into Equation (33), yields

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

After temporal linearization, the previous implicit right-hand residual term is explicitly solved now. Next, the Jacobian matrix needs to be split. Due to the complexity of the spatial discretization, it is difficult to accurately solve the Jacobian matrix. The flux-splitting of the inviscid flux Jacobian matrix is given first. Then the treatment of viscous one is discussed.

5.1. Flux Splitting Method

The maximum eigenvalue splitting [47] of the Jacobian matrix of inviscid flux $\mathbf{A}_m^{inv}$ is

$$\begin{cases} \mathbf{A}_m^{inv,+} = \frac{\mathbf{A}_m^{inv} + \mathbf{I}\beta\lambda_{m,p}^{inv}}{2} \\ \mathbf{A}_m^{inv,-} = \frac{\mathbf{A}_m^{inv} - \mathbf{I}\beta\lambda_{m,p}^{inv}}{2} \end{cases} \tag{39}$$

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

$$\lambda_{m,p}^{inv} = |\mathbf{v}_{m,p} \cdot \Delta \mathbf{S}_{m,p}| + c_{m,p} \|\Delta \mathbf{S}_{m,p}\| \tag{40}$$

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

$$\mathbf{A}_{m,p}^{inv,k}\Delta\mathbf{Q}_{m,p}^{k+1} \approx \mathbf{A}_{m,p}^{inv,k,+}\Delta\mathbf{Q}_{m}^{k+1} + \mathbf{A}_{m,p}^{inv,k,-}\Delta\mathbf{Q}_{n}^{k+1} \tag{41}$$

Substituting into Equation (38), we obtain

$$\frac{V_m}{\Delta t}\Delta\mathbf{Q}_m^{k+1} + \sum_{p\in\Xi(m)}\left(\mathbf{A}_{m,p}^{inv,k,+}\Delta\mathbf{Q}_{m}^{k+1} + \mathbf{A}_{m,p}^{inv,k,-}\Delta\mathbf{Q}_{n}^{k+1}\right) = -\bar{\mathbf{R}}(\mathbf{Q}_m^k) \tag{42}$$

or

$$\left(\frac{V_m}{\Delta t}\mathbf{I} + \sum_{p\in\Xi(m)}\mathbf{A}_{m,p}^{inv,k,+}\right)\Delta\mathbf{Q}_m^{k+1} + \sum_{p\in\Xi(m)}\mathbf{A}_{m,p}^{inv,k,-}\Delta\mathbf{Q}_{n}^{k+1} = -\bar{\mathbf{R}}(\mathbf{Q}_m^k) \tag{43}$$

It should be noted that the viscous part is not considered yet. If the Jacobian matrix $\mathbf{A}_{m,p}^{inv}$ is calculated by the first-order precision, we have

$$\sum_{p\in\Xi(m)}\mathbf{A}_{m,p}^{inv,+} = \frac{\beta\mathbf{I}}{2}\sum_{p\in\Xi(m)}\lambda_{m,p}^{inv} \tag{44}$$

So the Equation (43) is converted into

$$\left(\frac{V_m}{\Delta t} + \frac{\beta}{2}\sum_{p\in\Xi(m)}\lambda_{m,p}^{inv}\right)\Delta\mathbf{Q}_m^{k+1} + \sum_{p\in\Xi(m)}\mathbf{A}_{m,p}^{inv,k,-}\Delta\mathbf{Q}_{n}^{k+1} = -\bar{\mathbf{R}}\left(\mathbf{Q}_m^k\right) \tag{45}$$

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

5.2. Treatment of 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

$$\lambda_m^{vis} = \left[max\left(\frac{4}{3\rho_m}, \frac{\gamma}{\rho_m}\right)\right]\left(\frac{\mu_m}{P_r} + \frac{\mu_{t,m}}{P_{r,t}}\right)\frac{(\Delta S_m^x)^2 + (\Delta S_m^y)^2 + (\Delta S_m^z)^2}{V_m} \tag{46}$$

where

$$\begin{cases} \Delta S_m^x = \frac{1}{2}\sum_{p\in\Xi(m)} |\Delta \mathbf{S}_{x,m,p}| \\ \Delta S_m^y = \frac{1}{2}\sum_{p\in\Xi(m)} |\Delta \mathbf{S}_{y,m,p}| \\ \Delta S_m^z = \frac{1}{2}\sum_{p\in\Xi(m)} |\Delta \mathbf{S}_{z,m,p}| \end{cases} \tag{47}$$

6. Results and Discussion

The mesh element is directly used as the control volume in the finite volume code. In other words, it is the so-called cell-centered scheme. The meshfree point is the mesh node of the corresponding finite volume method in order to keep consistent in comparisons. Each point cloud collects all the nodes which share the same mesh element with the current point. Three cases are chosen. The first is from NASA Turbulence Modeling Resource in order to consistently verify the FV method and AV meshfree method with experiment and the third-party codes. The second one is a 3D transonic simulation to demonstrate the robustness of the method for solving flow discontinuity. The last case presents a demo of hybrid FV method and AV meshfree method. Both methods are simultaneously used in different domains to solve the motion of a store separation. In order to explain simply and clearly, the code is named as $\mathbf{GC_{fd}}$.

6.1. Study of Mesh/Point Sensitivity

The refining rule indeed affects the conclusion. In order to better reflect the regularity of mesh refining, a set of meshes recommended by NASA [48] for mesh dependency study is chosen. The structured mesh can provide better controllability of the node distribution. So the mesh refinement is easy to keep similarity. The airfoil is NACA0012. The farfield is 500 times of the chord. The mesh size includes five levels: extra coarse, coarse, medium, fine, and extra fine. The number of elements will be increased to 4 times after each refinement. The number of extra coarse mesh elements is 3584 while that of the extra fine is 917504. The medium mesh is shown in Figure 3. SA turbulence model is used. Mach number is 0.15 and Reynolds number

is 6×10^6. Roe's scheme is used for the inviscid flux. No limiter is used in solution reconstruction. The convergence of both the FV method and AV meshfree method is investigated and compared.

Figure 4 shows the convergence history of maximum and average residual. In the figure, "Max" represents the maximum residual of the energy equation. "Ave" is the averaged one. Time shown is based on the workstation of which CPU is Intel Xeon E5-2630 @ 2.30GHz and 24 threads are employed for the simulations. The flowfield is uniformly initialized by the farfield settings for both methods. The Courant, Fredricks, and Levy (CFL) criterion is set to 20. Both the FV method and AV meshfree method converge well. The convergence speed of the AV meshfree method is quite similar to that of the FV method when we focus on the time steps. For CPU time, the convergence of meshfree method becomes much slower because the number of neighbours (point cloud set $\tilde{\Xi}(m)$) of each algebraic volume (AV) is two times of that of each control volume (CV). Figure 5 compares the convergence history of the aerodynamic force. The results indicate that the force of the FV method converges more quickly.

Figure 6 presents the solved non-dimensional pressure and turbulence viscosity, respectively. The results of two methods look very similar. Figure 7 compares the pressure and friction coefficients with experiment between $\mathbf{GC_{fd}}$ and CFL3D. The latter is the code of NASA Langley. In the figure, C represents the chord. The result of fine mesh is shown because CFL3D only submitted that of this mesh size. All the data agree quite well with each other for both angle of attack 0° and 10°. The friction comparisons indicate that the AV meshfree method resolves the boundary layer as accurately as the finite volume on this mesh size. The convergence of mesh and point size is presented in Figure 8. The force coefficients calculated by both the FV method and AV meshfree method can gradually converge as the mesh (or point cloud) is refined. From the comparison, the convergence speed of mesh-based FV method is faster than that of the meshfree method. The accuracy of the FV method is also slightly higher.

Figure 9 shows the curve of the aerodynamic force coefficient with the angle of attack. The results of the FV method and AV meshfree method are quite close, both of which are in good agreement with the experimental values. The calculated maximum lift coefficient and stall angle of attack are all in good agreement with the measurement. The drag coefficient agrees very well with the experimental value when the angle of attack is small. The calculation error of both methods increases with the increase of the angle of

attack. The error of drag coefficient becomes bigger because the RANS is not suitable for separation flow. The error is also related to the transition technique used in the experiment while all the CFD simulations are fully turbulent. Moreover, the results of the meshfree method are quite close to those of the FV method even if the separation occurs.

Table 1 compares the calculation results with the experimental values and those of the third-party CFD codes [48]. They are in good agreement. For the lift coefficient, when the angle of attack is 0°, the lift coefficients calculated by both the FV method and AV meshfree method are all very close to 0. When the angle of attack is increased to 10°, the results of $\mathbf{GC_{fd}}$ are consistent with the third-party CFD codes. The lift coefficient obtained by experiment is about 1.4% higher. For the drag coefficient, when the angle of attack is small, the calculated value is quite close to the experimental one. It is more accurate than the results of the third-party codes. When the angle of attack increases to 10°, the results are consistent with those of the third-party codes, and only slightly differ from the experimental value by about 5 counts. The difference between the calculated value and the experimental result may be due to the influence of turbulence simulation. The forced transition was used in the experiment, while the CFD simulations are fully turbulent.

6.2. Application to 3D Transonic Flow

The medium mesh from the last section is used in the 3D validation. The 2D mesh is extruded at the spanwise direction for 1 chord length where 20 layers are generated. The 3D mesh is shown in Figure 10. Mach number is 0.775 and the angle of attack is 2.05°. Reynolds number is 10^7. The convergence of the residual and aerodynamic force is compared in Figure 11. The initial flow field is uniform. CFL number is 2.5. The Venkatakrishnan limiter is applied in the simulation. The result indicates that both methods can converge very well for the transonic flow. Similar to the results of the last section, the convergence of the force coefficient by the meshfree method is still slower than that of FV. Figure 12 compares the obtained flowfield. It is found that both methods can capture the shock wave within one to two mesh elements (or points for meshfree method). The resolution of the shock wave is high for both. It should be noted that the colour shown at the leading edge looks different from each other because the node display of meshfree method couldn't show the gradual change of colour at a single point. The pressure and friction coefficients are compared in Figure 13. Each tiny spot on the

curve represents one mesh or point. It shows the shock wave in the boundary layer is captured at around 2 elements (or points). Both the FV method and AV meshfree method obtain very good agreement with experimental data. The friction agrees well with each other as well. The comparison indicates that the position of shock wave affects the friction significantly downstream.

6.3. Demo of Hybrid FV and Meshfree Method

Fries and Matthies [49, 50] developed a coupled meshfree/mesh-based method for the incompressible Navier-Stokes equations. The meshfree Galerkin method was used for the approximation of the governing equation. Different from these works, a hybrid FV method and AV meshfree method is proposed here. From the verifications of the last two sections, the accuracy and convergence of AV meshfree method are competitive to traditional FV method. It has the potential to solve the high Reynolds number, transonic flow. Furthermore, because the AV meshfree method shares the Navier-Stokes solver with traditional FV method, we can naturally combine both FV method and AV meshfree method to solve the complex geometry or moving boundary.

Here is an application of store separation simulation [51]. The store is released from a wing. The meshing should be dynamic to solve the motion of store. A strategy is presented here: the traditional mesh-based FV method is employed for both store domain (moving) and wing domain (stationary); the chimera mesh method [52] is used for hole cutting between two domains; the meshfree method is applied at the interface for information communication. A 2D example is used to explain the strategy, as depicted in Figure 14. Blue lines represent the Domain 1 after hole cutting by chimera method while red ones are Domain 2. Two layers mesh is used for the interface. Their nodes marked as black are directly used for meshfree points. And the gray points are the boundary of the meshfree method which flow variables are interpolated from the mesh based FV method. The boundary information of FV is interpolated from meshfree points.

The mesh of wing and store domains is shown in Figure 15. The number of elements around the wing is 2.6 million, nodes 0.7 million. For the store domain, the number of elements 1.8 million, that of nodes is 0.5 million. Mach number of freestream is 0.95, the angle of attack is 0°, and the height of atmosphere for air condition is 8000m. AUSM scheme is used for the inviscid flux solution. The CFL number is 10. The motion of store is solved by the 6-DoF (degree of freedom) equations in which aerodynamic force is calculated by CFD. The meshfree method is used for the communication

between the wing domain and store domain. Figure 16 shows a slice before the release of the store. The real-time step size is 0.005s (second). The convergence criterion of pseudo-time iteration is that the maximum residual of the energy equation is less than 10^{-5}.

Figure 17 shows the slices of Mach number of two time snapshots where the wall surface displays the pressure. The region between two black lines is solved by the meshfree method. The contour shows very good continuity between different zones. Figure 18 presents the trajectory of the store. The comparison between the calculated results and the experimental values is shown in Figure 19. Fx, Fy, and Fz are the total force components at x, y, and z directions, respectively. Mx, My, and Mz are the total moment components at x, y, and z directions, respectively. Xc, Yc, and Zc represent the displacements of the centroid. ϕ, θ, and Ψ are the roll angle, pitch angle, and yaw angle, respectively. Uc, Vc, and Wc represent the centroid velocity, respectively. ω_x, ω_y, and ω_z are the roll angular velocity, pitch angular velocity, and yaw angular velocity, respectively. The obtained results are all in good agreement with the experimental values.

7. Conclusions

An algebraic volume (AV) concept is proposed to allow the finite volume (FV) solver to implement the meshfree method conveniently. The algebraic face area vector is constructed via the weighted least squares method. The traditional FV code only needs minor changes in the calculation method of geometric control volume to algebraic matrix operation in pre-processing. It is suitable to solve the high Reynolds, transonic viscous flow. The semi-discretized NS equations are implicitly solved by multiple symmetric Gauss-Seidel iterations.

The mesh (or point in meshfree method) convergence study shows that the results of both FV method and AV meshfree method are well convergent when the size of mesh (or point) increases. The pressure and friction coefficients of AV meshfree method agree quite well with those of the FV method. Both obtain almost the same distribution with experimental data and the submitted CFL3D solution. The aerodynamic forces also agree well with measurement and those of third-party CFD solvers. The transonic simulation indicates that the meshfree method converges quite well. The pressure and friction coefficients agree well with those of the FV method. The position of the shock is also captured remarkably well.

A demo of hybrid FV method and AV meshfree method is presented to simulate the store separation. The meshfree method is used to exchange the flow information between the wing and store domains which utilize the FV method. The solved flowfield is continuous well between FV and meshfree method zones even if the shock wave goes through the interface. The simulated trajectory of the released store is in good agreement with experimental data. It successfully demonstrates a complex application.

8. References

[1] J. F. Thompson, F. C. Thames, C. Mastin, Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies, Journal of Computational Physics 15 (1974) 299–319.

[2] P. R. Eiseman, Grid generation for fluid mechanics computations, Annual Review of Fluid Mechanics 17 (1985) 487–522.

[3] H. J. Fogg, C. G. Armstrong, T. T. Robinson, Automatic generation of multiblock decompositions of surfaces, International Journal for Numerical Methods in Engineering 101 (2015) 965–991.

[4] F. Wang, L. di Mare, Mesh generation for turbomachinery blade passages with three-dimensional endwall features, Journal of Propulsion and Power 33 (2017) 1459–1472.

[5] W. M. Chan, Overset grid technology development at NASA ames research center, Computers & Fluids 38 (2009) 496–503.

[6] D. J. Mavriplis, Unstructured grid techniques, Annual Review of Fluid Mechanics 29 (1997) 473–514.

[7] E. Marchandise, C. Piret, J.-F. Remacle, CAD and mesh repair with radial basis functions, Journal of Computational Physics 231 (2012) 2376–2387.

[8] Y. Zheng, Z. Xiao, J. Chen, J. Zhang, Novel methodology for viscous-layer meshing by the boundary element method, AIAA Journal 56 (2018) 209–221.

[9] Y. Liu, G. Wang, Z. Ye, Dynamic mode extrapolation to improve the efficiency of dual time stepping method, Journal of Computational Physics 352 (2018) 190–212.

[10] Z. Wang, A quadtree-based adaptive cartesian/quad grid flow solver for Navier-Stokes equations, Computers & Fluids 27 (1998) 529–549.

[11] W. Bennett, N. Nikiforakis, R. Klein, A moving boundary flux stabilization method for cartesian cut-cell grids using directional operator splitting, Journal of Computational Physics 368 (2018) 333–358.

[12] S. Muzaferija, D. Gosman, Finite-volume CFD procedure and adaptive error control strategy for grids of arbitrary topology, Journal of Computational Physics 138 (1997) 766–787.

[13] S. Chalasani, D. Thompson, Quality improvements in extruded meshes using topologically adaptive generalized elements, International Journal for Numerical Methods in Engineering 60 (2004) 1139–1159.

[14] R. A. Gingold, J. J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Monthly Notices of the Royal Astronomical Society 181 (1977) 375–389.

[15] G. Liu, Meshfree Methods, CRC Press, 2009.

[16] E. Oñate, C. Sacco, S. Idelsohn, A finite point method for incompressible flow problems, Computing and Visualization in Science 3 (2000) 67–75.

[17] X. K. Zhang, K.-C. Kwon, S.-K. Youn, Least-squares meshfree method for incompressible Navier-Stokes problems, International Journal for Numerical Methods in Fluids 46 (2004) 263–288.

[18] Y. Kim, D. W. Kim, S. Jun, J. H. Lee, Meshfree point collocation method for the stream-vorticity formulation of 2D incompressible Navier-Stokes equations, Computer Methods in Applied Mechanics and Engineering 196 (2007) 3095–3109.

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

[20] M. Borthakur, A. Biswas, A novel Hermite Taylor least square based meshfree framework with adaptive upwind scheme for two dimensional incompressible flows, Computers & Fluids 130 (2016) 37–48.

[21] M. Dehghan, M. Abbaszadeh, Proper orthogonal decomposition variational multiscale element free Galerkin (POD-VMEFG) meshless method for solving incompressible Navier–Stokes equation, Computer Methods in Applied Mechanics and Engineering 311 (2016) 856–888.

[22] 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.

[23] M. Dehghan, M. Abbaszadeh, An upwind local radial basis functions-differential quadrature (RBF-DQ) method with proper orthogonal decomposition (POD) approach for solving compressible euler equation, Engineering Analysis with Boundary Elements 92 (2018) 244–256.

[24] Z. Sedaghatjoo, M. Dehghan, H. Hosseinzadeh, Numerical solution of 2D Navier-Stokes equation discretized via boundary elements method and finite difference approximation, Engineering Analysis with Boundary Elements 96 (2018) 64–77.

[25] B. H. Foy, P. Perré, I. Turner, The meshfree finite volume method with application to multi-phase porous media models, Journal of Computational Physics 333 (2017) 369–386.

[26] D. Hietel, K. Steiner, J. Struckmeier, A finite-volume particle method for compressible flows, Mathematical Models and Methods in Applied Sciences 10 (2000) 1363–1382.

[27] D. Sridar, N. Balakrishnan, An upwind finite difference scheme for meshless solvers, Journal of Computational Physics 189 (2003) 1–29.

[28] G. Wang, Y. Sun, Z. Ye, Gridless solution method for two-dimensional unsteady flow, Chinese Journal of Aeronautics 18 (2005) 8–14.

[29] G. Harish, M. Pavanakumar, K. Anandhanarayanan, Store separation dynamics using grid-free Euler solver, in: 24th AIAA Applied Aerodynamics Conference, American Institute of Aeronautics and Astronautics, 2006.

[30] 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.

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

[32] Z. Ma, H. Wang, S. Pu, A parallel meshless dynamic cloud method on graphic processing units for unsteady compressible flows past moving boundaries, Computer Methods in Applied Mechanics and Engineering 285 (2015) 146–165.

[33] Z. Duan, Z. J. Wang, B. T. Vu, Recent progresses on a meshless Euler solver for compressible flows, in: 22nd AIAA Computational Fluid Dynamics Conference, American Institute of Aeronautics and Astronautics, 2015.

[34] F. Günther, W. K. Liu, D. Diachin, M. A. Christon, Multi-scale meshfree parallel computations for viscous, compressible flows, Computer Methods in Applied Mechanics and Engineering 190 (2000) 279–303.

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

[36] D. J. Kennett, S. Timme, J. Angulo, K. J. Badcock, An implicit meshless method for application in computational fluid dynamics, International Journal for Numerical Methods in Fluids 71 (2012) 1007–1028.

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

[38] E. Ortega, E. Oñate, S. Idelsohn, R. Flores, Application of the finite point method to high-Reynolds number compressible flow problems, International Journal for Numerical Methods in Fluids 74 (2014) 732–748.

[39] 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.

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

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

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

[43] 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.

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

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

[46] 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.

[47] 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.

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

[49] 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.

[50] 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.

[51] E. R. Heim, CFD wing-pylon-finned store mutual interference wind tunnel experiment, AEDC-TSR-91-P4 (1991).

[52] F. Togashi, Y. Ito, K. Nakahashi, S. Obayashi, Overset unstructured grids method for viscous flow computations, AIAA Journal 44 (2006) 1617–1623.

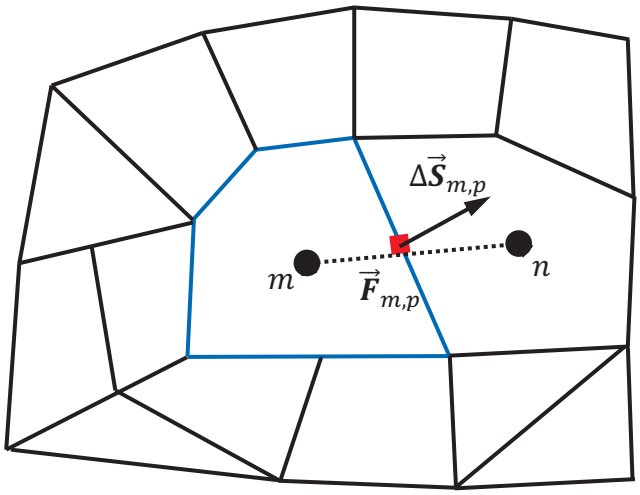


Figure 1 A sketch of arbitrary polygonal control volume m (blue lines)

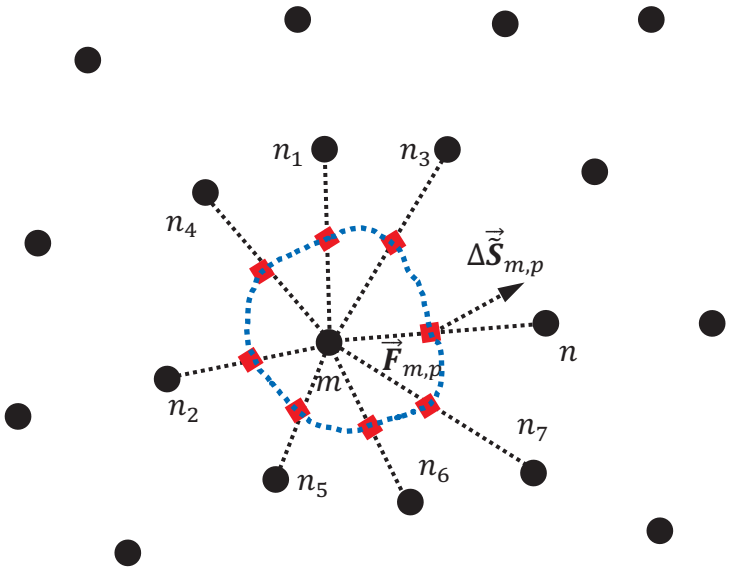


Figure 2 A sketch of construction of algebraic volume $\tilde{m}$ on point cloud

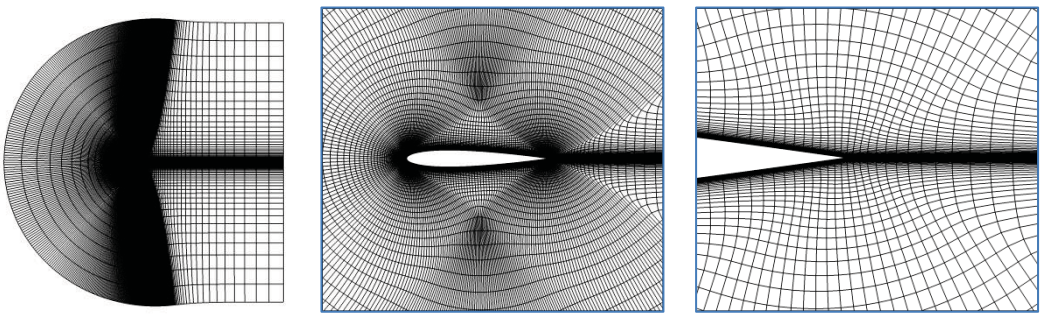


Figure3 Structured mesh around NACA0012 airfoil (medium mesh)

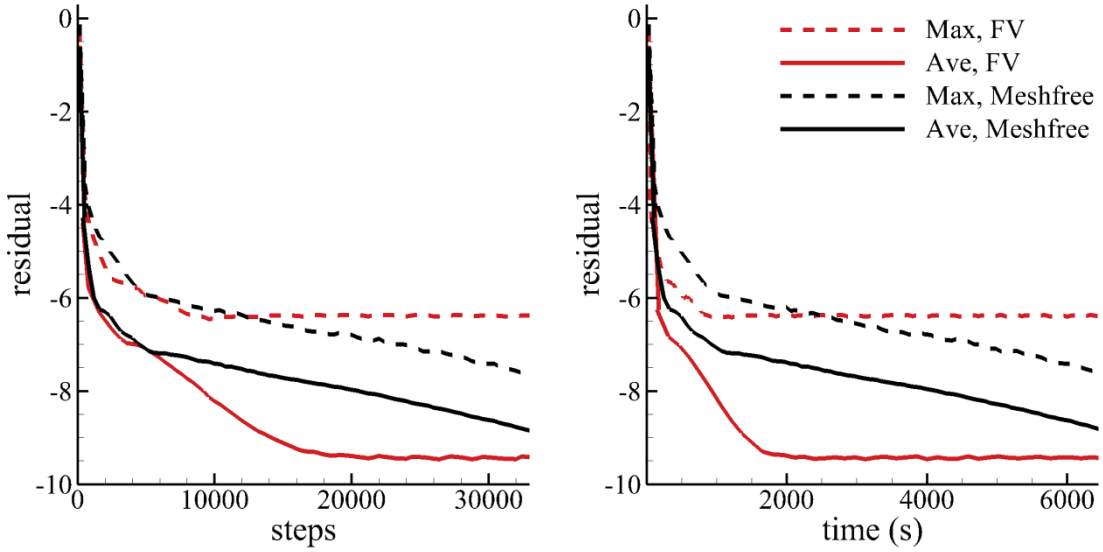


Figure 4 Convergence history of residual (medium), α= 10°

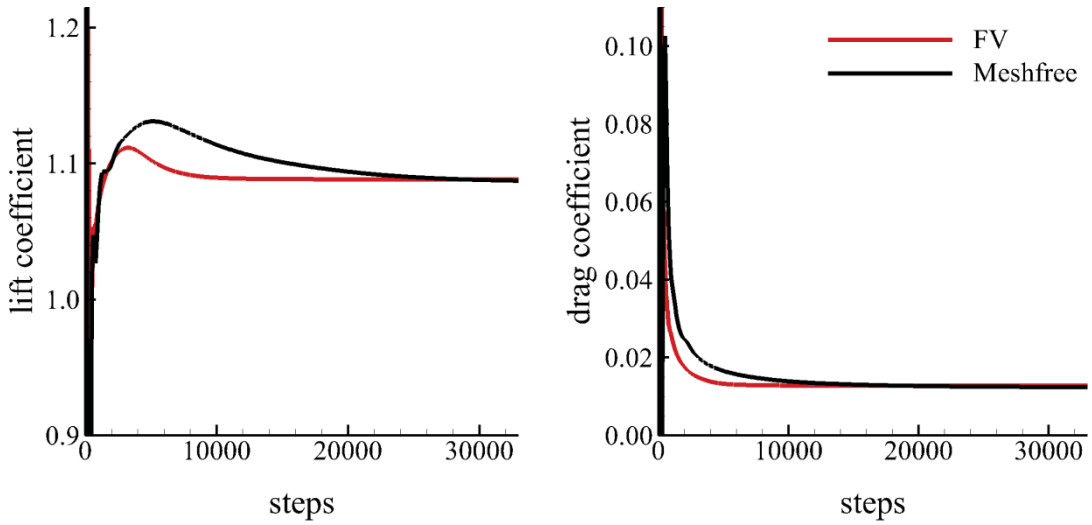
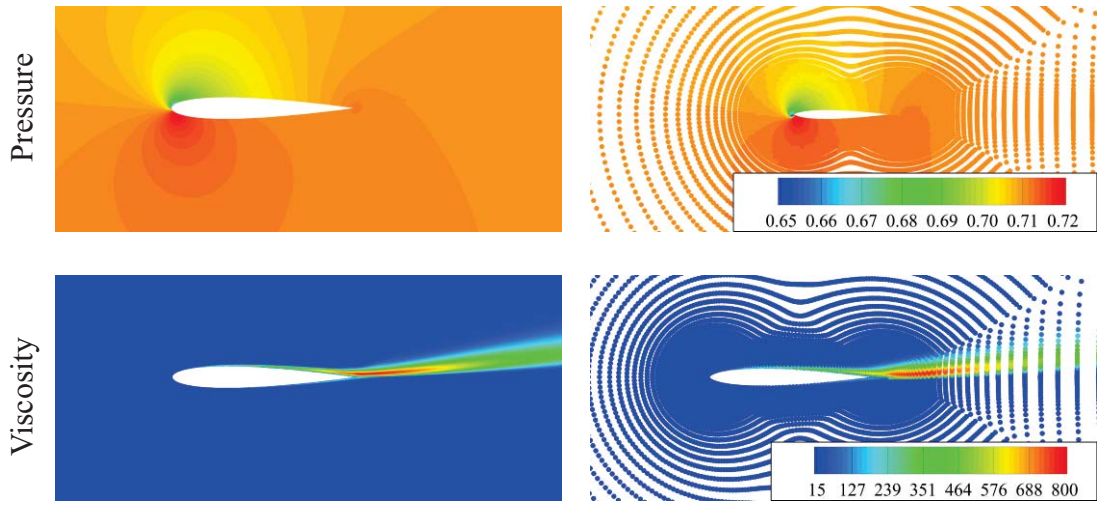
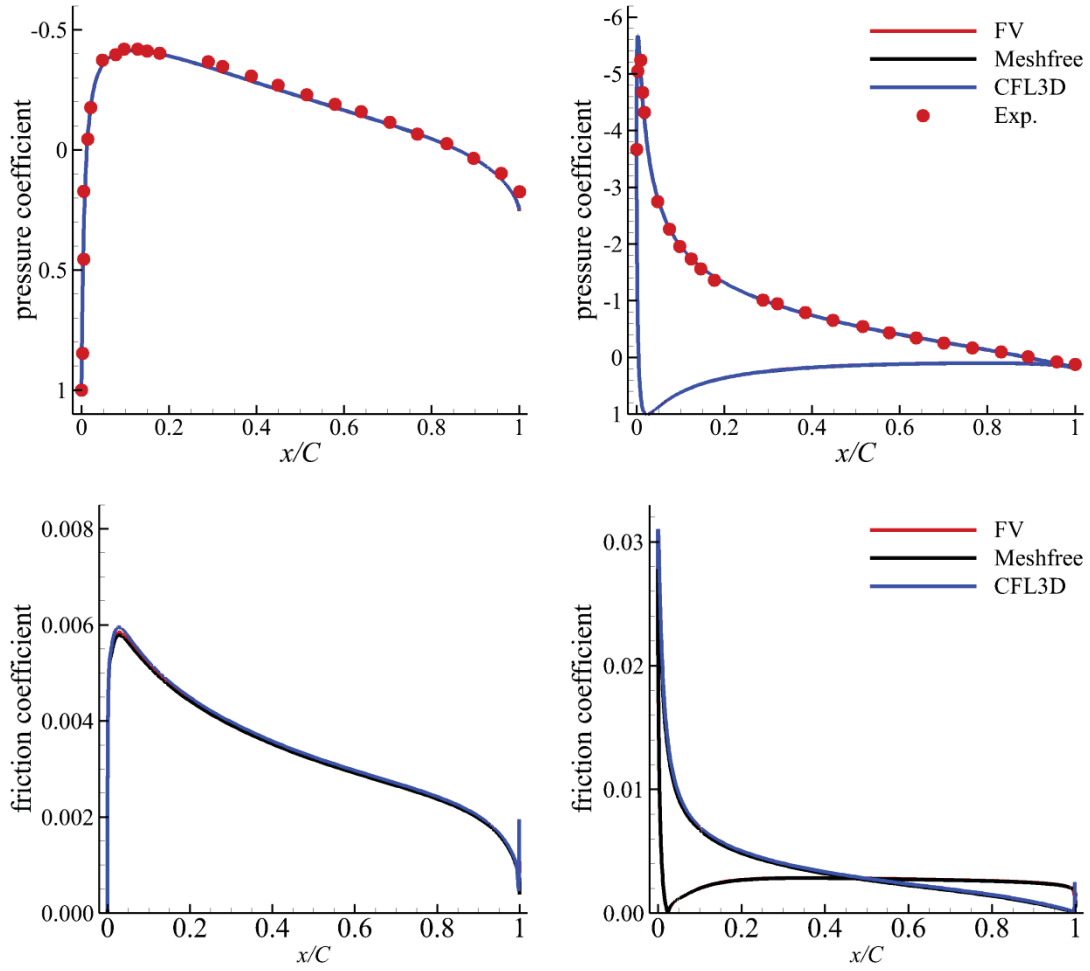


Figure 5 Convergence history of force (medium), α= 10°



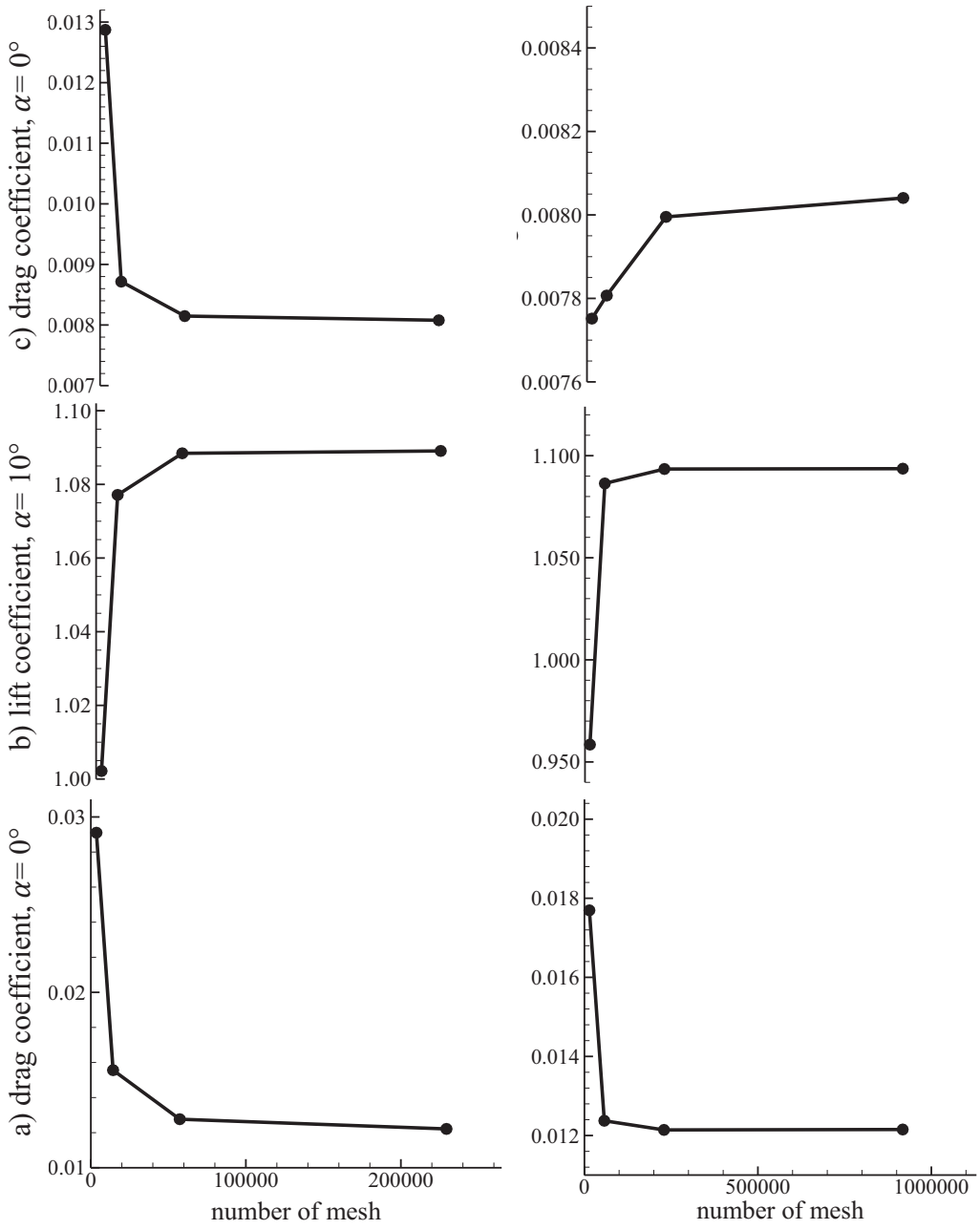
a) FV method b) Meshfree method

Figure 6 Contour of non-dimensional pressure (upper) and turbulence viscosity (lower) (medium), α= 10°



a) α= 0° b) α= 10°

Figure 7 Surface pressure (left) and friction (right) coefficients (fine)



a) FV method b) Meshfree method

Figure 8 Mesh dependency study of FV method and AV meshfree method

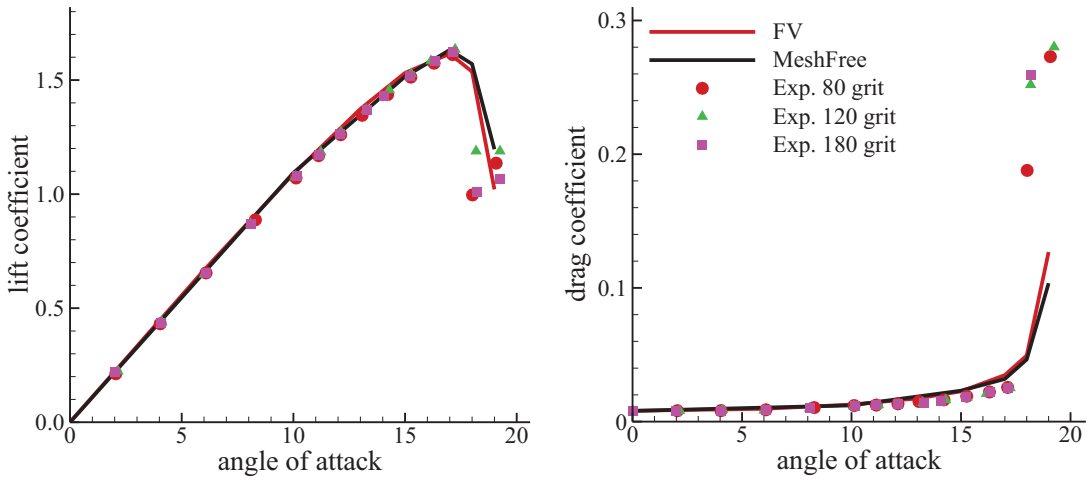


Figure 9 Comparison of lift (left) and drag (right) coefficients (medium mesh)

Table 1 Comparison between GC_{fd} and other methods (fine mesh)

Method		C_l (α= 0°)	C_d (α= 0°)	C_l (α= 10°)	C_d (α= 10°)
Experiment	80 grit	×	0.00809	1.0707	0.01201
	120 grit	×	0.00804	1.0775	0 .01175
	180 grit	×	0.00803	1.0809	0.01165
GC_{fd}	**FV**	**-0.0000033**	**0.00808**	**1.0891**	**0.01221**
	Meshfree	**-0.0000052**	**0.00800**	**1.0935**	**0.01214**
Other Solvers	CFL3D	×	0.00819	1.0909	0.01231
	FUN3D	×	0.00812	1.0983	0.01242
	NTS	×	0.00813	1.0891	0.01243
	JOE	×	0.00812	1.0918	0.01245
	SUMB	×	0.00813	1.0904	0.01233
	TURNS	×	0.00830	1.1000	0.01230
	GGNS	×	0.00817	1.0941	0.01225

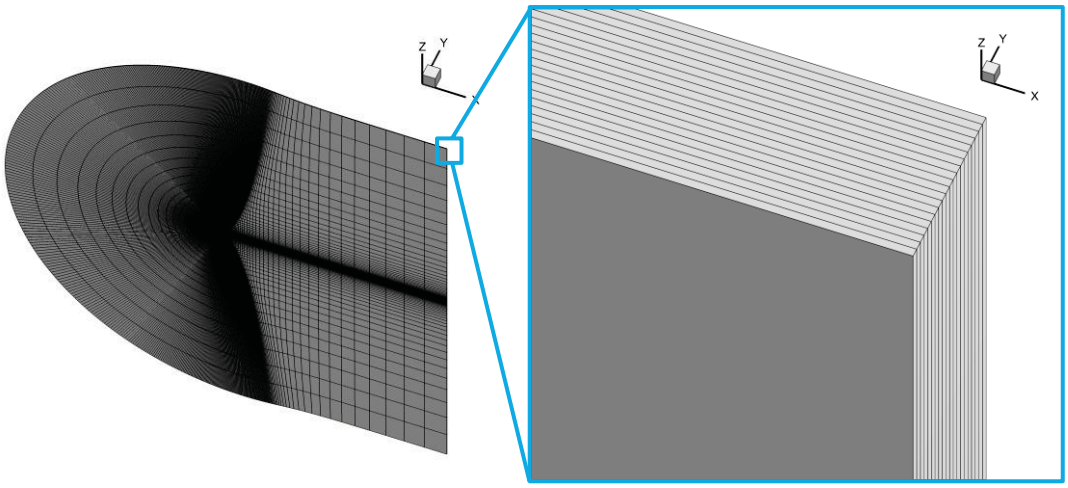


Figure 10 Extruded medium mesh for 3D verification

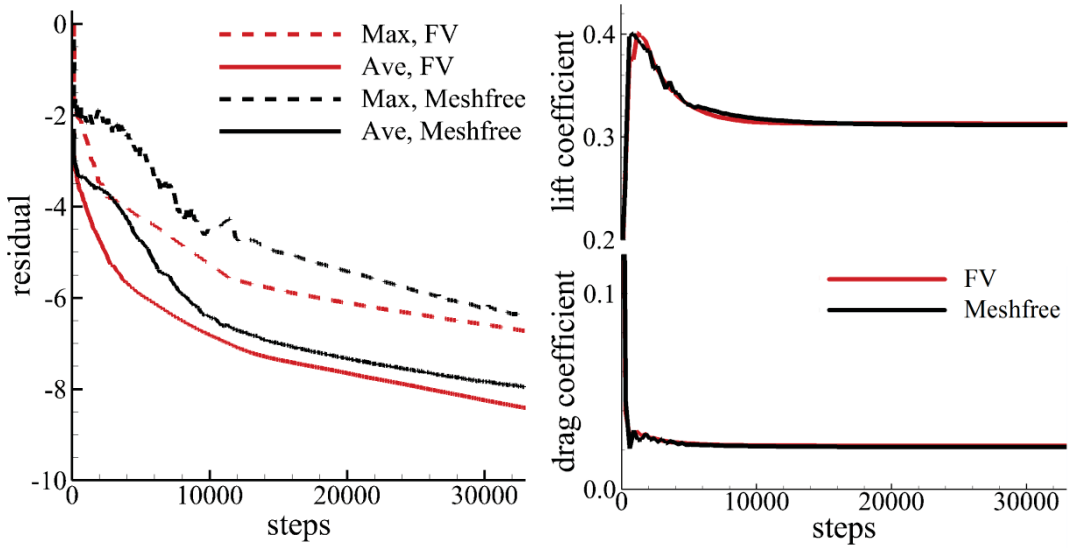
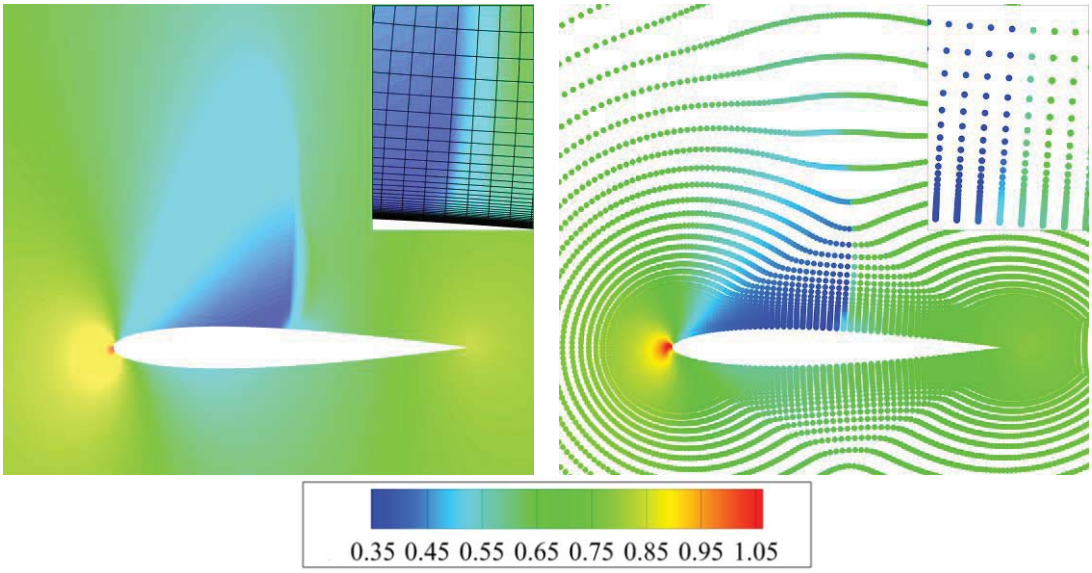


Figure 11 Convergence history of residual (left) and force coefficients (right)



a) FV method b) Meshfree method

Figure 12 Contour of non-dimensional pressure

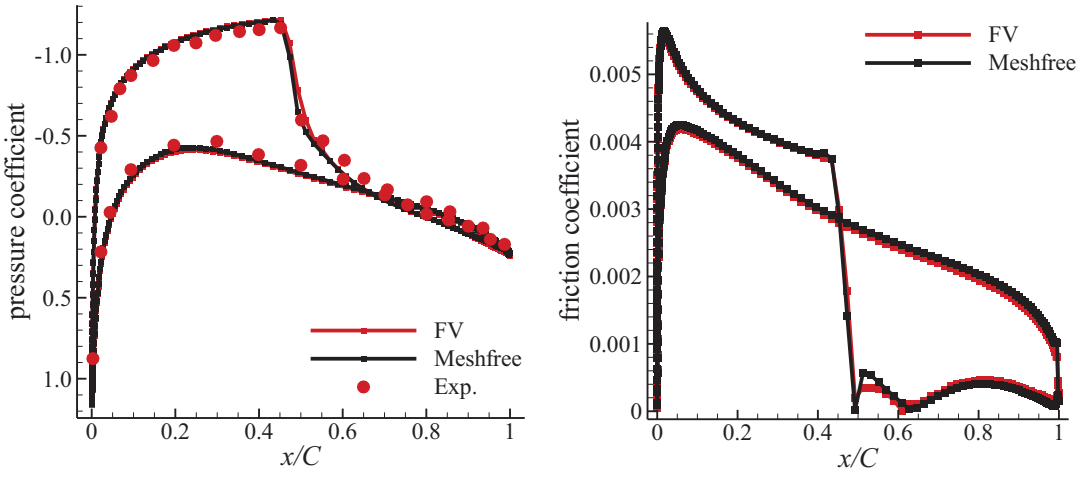


Figure 13 Comparison of pressure (left) and friction (right) coefficients on the airfoil

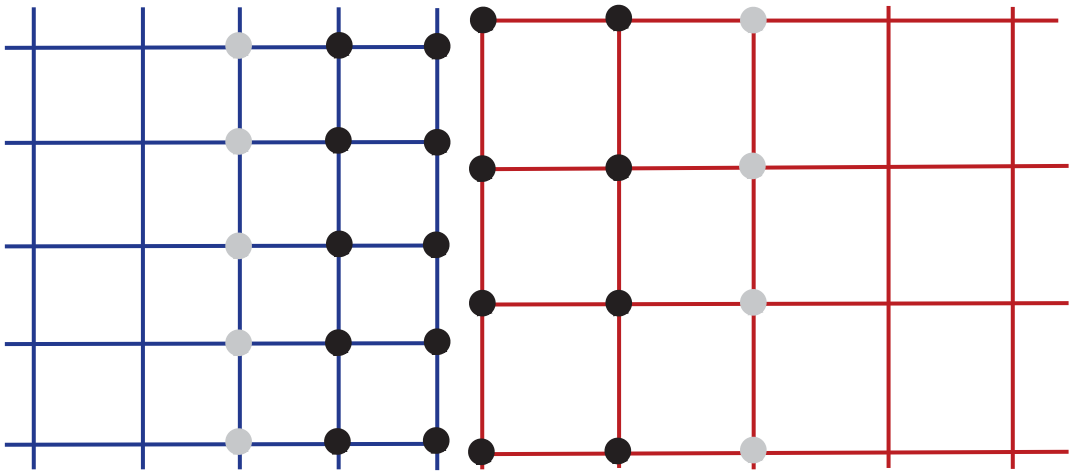


Figure 14 Meshfree method for information exchange between two FV domains

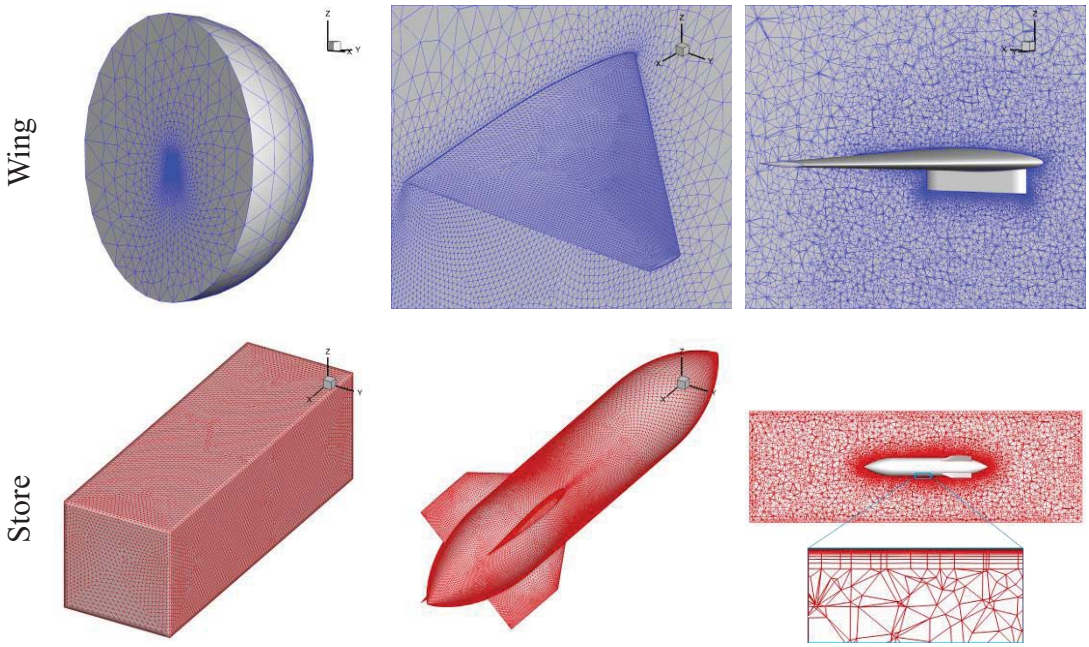


Figure 15 Meshing of two domains

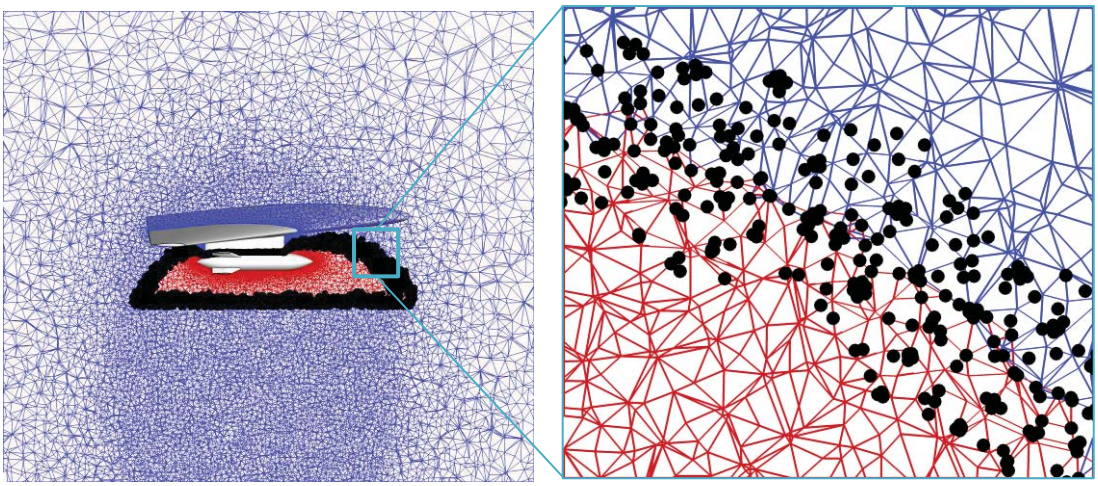
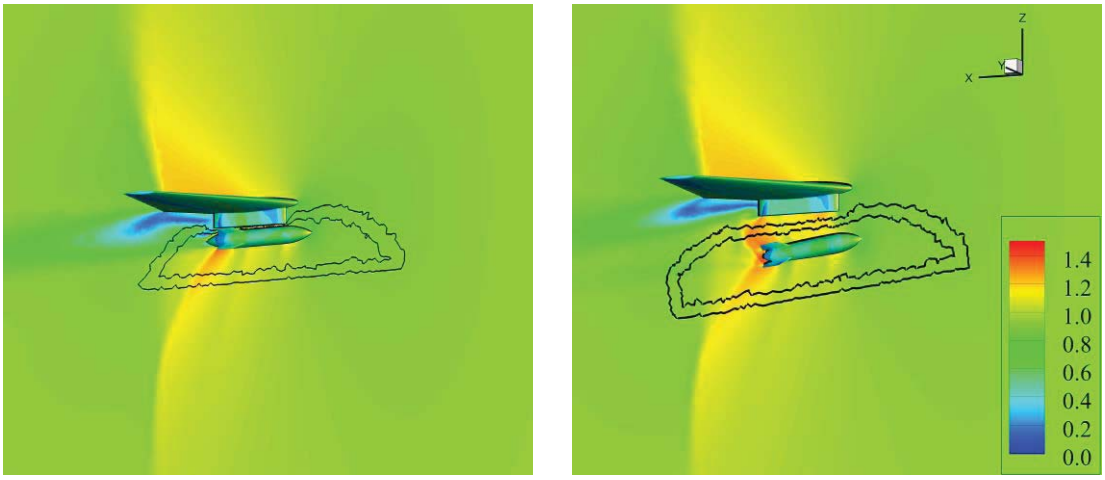
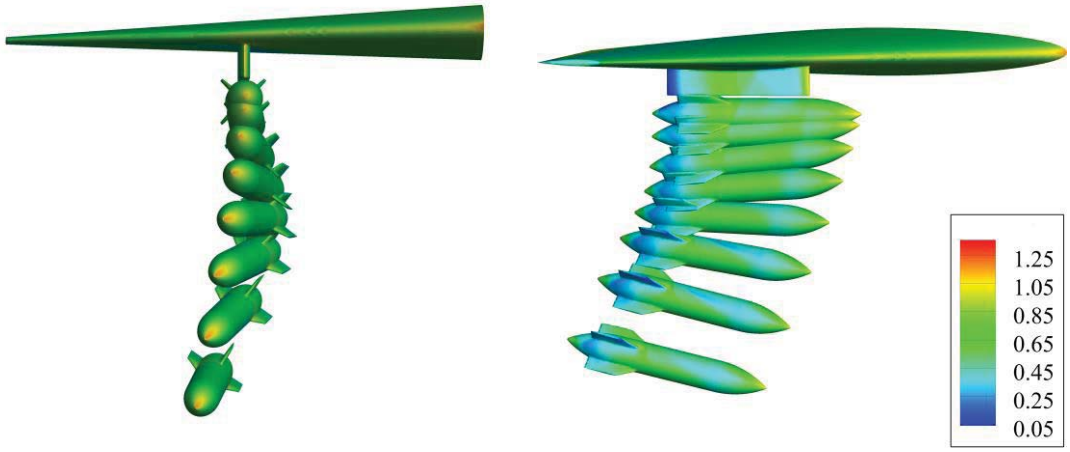


Figure 16 Mesh and point zones for hybrid FV method and AV meshfree method



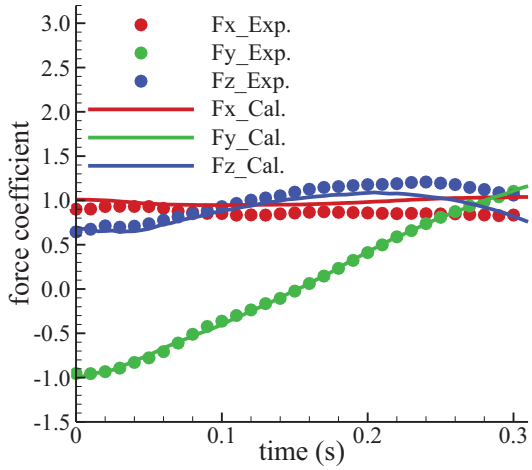
a) 0.05s b) 0.2s

Figure 17 Slices of Mach number field (wall surface contour is pressure)

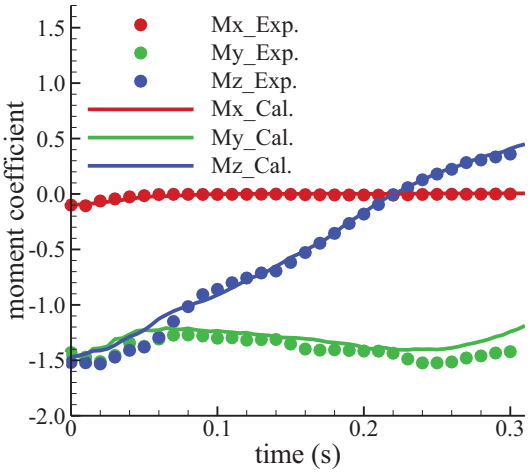


a) Front view b) Side view

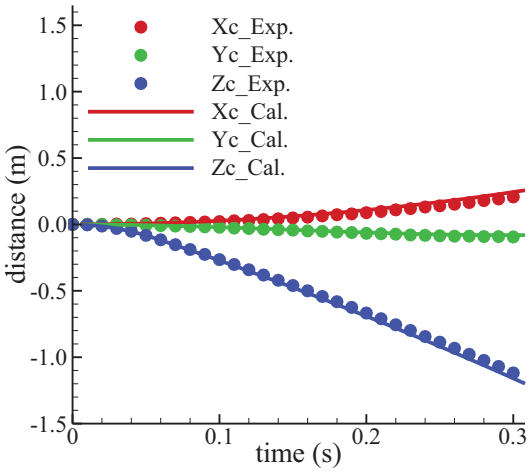
Figure 18 Trajectory of the store (every 0.05s, contour is non-dimensional pressure)



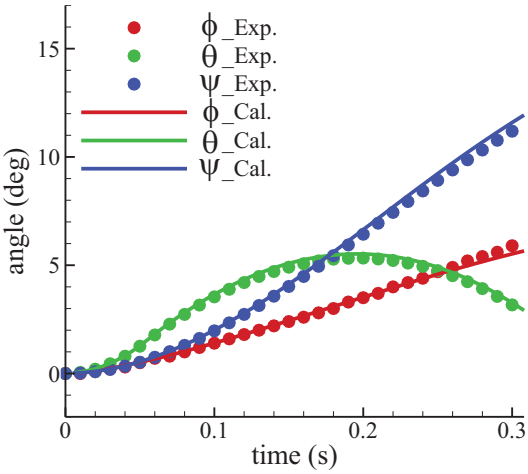
a) Force



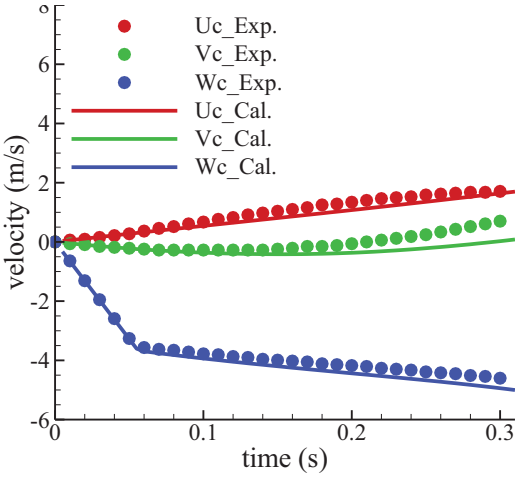
b) Moment



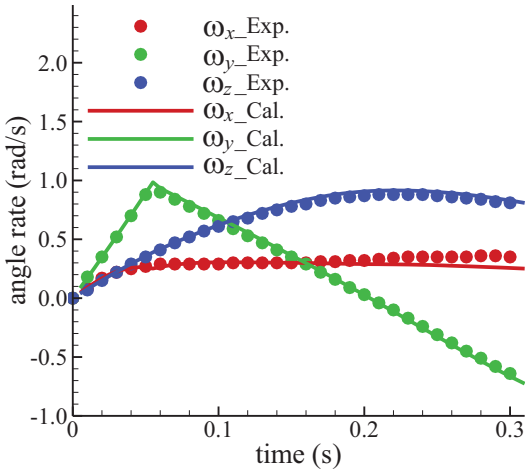
c) Displacement



d) Angle



e) Velocity



f) Angle rate

Figure 19 Compared to experiment data