Numerical Study of Flow Phenomena Concerning Interaction of Flow and Complex or Moving Boundaries Using Immersed Boundary Method

张, 晏榕

https://hdl.handle.net/2324/1959133

出版情報：九州大学, 2018, 博士（工学）, 課程博士
バージョン：
権利関係：

Numerical study of flow phenomena concerning interaction of flow and complex or moving boundaries using immersed boundary method

Yanrong Zhang

Department of Aeronautics and Astronautics

Kyushu University

July, 2018

Contents

List of figures

List of tables

Nomenclature

a	Constant
$\vec{a}$	Acceleration of the model
$\overrightarrow{a_f}$	Acceleration of fluid
A	Undulatory amplitude
A_c	Reference area
A_0	Modelling parameters
b	Constant
b_{ij}^{SGS}	Anisotropy tensor
c	Constant
c_s	Size parameters of the fish model
Cl	Lift coefficient
Cd	Drag coefficient
C_B, C_{SGS}, C_l, C_ε, C_k	Modelling parameters
D	Diameter of circular cylinder in validation case
$\vec{f}$	Body force
$\overrightarrow{f_s}$	Surface force
$\vec{F}$	External force
F_d	Drag force
F_l	Lift force
h	Roughness height from peak to valley
$\vec{\imath}$, $\vec{\jmath}$, $\vec{k}$	Unit vectors of the inertia coordinate system
$\overrightarrow{\imath_s}$, $\overrightarrow{\jmath_s}$, $\overrightarrow{k_s}$ (s=1,2,…,7)	Unit vectors of the body-fixed coordinate system (s)
$[I_R]$	Inertia matrix
I	Constants
l	Size parameters of the fish model
l_x, l_z	Wavelengths of the roughness element in x- and z-directions

k_{SGS}	SGS turbulent kinetic energy (SGS turbulence energy)
m	Setting mass of the model
$\vec{n}$	Wall normal unit vector
p	Pressure
r_{1x}, r_{1y}, r_{1z}	Coordinates of the origin point of the body-fixed coordinate system (1) in the inertia coordinate system
$\vec{r}$	Location vector in the inertia coordinate system
$\vec{r_1}$	Location vector of the origin point of the body-fixed coordinate system (1) in the inertia coordinate system
$\vec{r_c}$	Location vector of the center of mass of the model in the inertia coordinate system
$\vec{r_n}$	Vector from the origin of $\vec{n}$ to the point to be identified
$\overrightarrow{r_{o1}}$	Location vector in the body-fixed coordinate system (1)
R	Size parameters of the fish model
Re	Reynolds number
RHS	Right-hand side
S_{ij}	Resolved strain-rate tensor
St	Strouhal number
t	Time
Δt	Scale of time step
$\vec{T}$	External torque
T	Temperature
$\vec{u}$	Velocity
$\vec{u}_b$	Velocity inside the immersed boundary
$\vec{u_s}$	Velocity of the origin point of the body-fixed coordinate system (s)
$\vec{u_\Gamma}$	Velocity at the immersed boundary
u_τ	Friction velocity
u_i	Velocity component in i direction

u_i^{n+1}, u_i^n	Velocity at node i at time step n+1, n
U_i	Velocity component in i-direction
U_∞	Far field velocity
$\vec{V}_{ib}$	Desired velocity inside the immersed boundary
x, y, z	Coordinates in the inertia coordinate system
x_i	Cartesian coordinate in i-direction
x_1, y_1, z_1	Coordinates in the body-fixed coordinate system (1)
$\vec{\alpha}$	Angular acceleration of the model
μ	Molecular viscosity
ρ	Density
ρ_f	Density of fluid
ν	Kinematic viscosity
Δ	SGS filter width
δ	Open-channel height
δ_{ij}	Kronecker delta
σ_0	Mean offset
ν_{SGS}	SGS eddy viscosity
ε_{SGS}	Dissipation rate of SGS turbulence energy
τ_{ij}	SGS-stress tensor
Γ_b	Immersed boundary
Ω_b	Field inside the body
Ω_f	Field of the fluid
Ω_b	Field inside the immersed boundary
ω	Undulatory angular velocity
$\vec{\omega}$	Angular velocity of the model
$\overrightarrow{\omega_s}$	Angular velocity of the body-fixed coordinate system (s)
ϕ	Phase difference between the neighboring parts of the model
$\overline{(..)}$	Filtered value

$\widehat{(..)}$	Test-filtering operator

Chapter 1 Introduction

1.1 About the boundary methods in computational fluid mechanics

Complicated or moving boundaries are common in normal life as well as various engineering applications. For example, free surface of the sea, flapping wings of a bird or beating tail of a fish and rotational propeller of an aircraft. Owing to the development of the computer technique, computational fluid dynamics (CFD) is becoming more and more important to study various flow phenomena in addition to the experiment. To perform the computational simulation for the problems concerning complicated or moving boundaries, dealing with the boundaries is a crucial point.

The treatment of the boundary in simulation is based on the scheme by which the movement of fluid is described. To describe the movement of fluid mathematically, there are the Eulerian method and the Lagrangian method as two different approaches. In the Eulerian method, the flow is expressed as a field in which observing points are fixed inside. By obtaining the physical quantities such as the velocity of fluid instantaneously passing these observing points, the movement of fluid is clarified. In the Lagrangian method, on the other hand, the flow field is not fixed, the flow is expressed by tracing the movement of fluid particles. Generally, in the numerical simulation of fluid flow, the Eulerian method is more popular than the Lagrangian method. While when there is free surface or moving boundary in a flow field, the Lagrangian method is also frequently used. Before simulation, it is necessary to discretize the simulation field, for example the structured meshes in the finite difference method. In the Eulerian approach, the discretized meshes are fixed during the simulation and the grid vertices could be regarded as the observing points for a flow. While in the Lagrangian approach, the grid vertices could be regarded as corresponding to fluid particles, thus could be made to move with the fluid particles. Therefore, the movement of a free surface or a moving boundary could not be directly captured by using the Eulerian approach and some additional processes to describe the movement of boundary need to be introduced. In contrast to this, in the Lagrangian approach, since the grid vertices are considered as the fluid particles, the moving boundary could be directly captured by tracing the grid vertices corresponding to the moving boundary. However, as the grid vertices move, remeshing is needed at each time step during the simulation, when in case of such as surface with large vibration or breaking wave, large movement of the corresponding grid vertices arises and the computation may be unavailable because of the collapsing of the mesh (数値流体力学編集委員会,1995). To improve this drawback of the Lagrangian method, Arbitrary Lagrangian-Eulerian (ALE) method was proposed. In the ALE, the moving velocity of the mesh that is independent of the velocity of the corresponding fluid particles is employed to prevent an excessive deformation of the mesh. To achieve that, a referential description of the fundamental equations is required. A conclusion about the properties of the ALE method as well as the Eulerian and Lagrangian methods is listed in Table 1.1.

Although describing the moving boundary by the Lagrangian and ALE methods is relatively straightforward, both of them need the remeshing during simulation. By using the Eulerian method, the remeshing could be avoided while additional processes are required to describe the moving boundary indirectly, such as height function, line segment, marker particle, density function and so on. Since by the Eulerian method the description of the moving boundary is not constrained by the mesh, it is relatively easier to deal with large deformation. However, the accuracy of computation is said to be generally lower due to a lower level of satisfaction of conservation near a moving boundary (数値流体力学編集委員会, 1995）.

Among such several strategies, the immersed boundary method that belongs to the Eulerian approach is a promising approach to deal with the complicated or moving boundary problems in CFD, in which the effect of a boundary is imposed by introducing an extra force into a fluid flow. Remarkable development has been made since it was proposed and numerous applications have been performed. While despite of the obtained achievement, applications of the immersed boundary are found usually confined to the problems with relatively low Reynolds numbers or relatively simple movement, applications including moving boundary with relatively complicated configuration or self-induced movement is also limited. The difficulty of obtaining a boundary layer accurately and describing the moving boundary in an unbody-fitted mesh is considered as one of the main reasons.

Table 1.1 Eulerian, Lagrangian and ALE methods (数値流体力学編集委員会, 1995）

Method	Velocity of mesh ($\vec{\hat{v}}$) and fluid ($\vec{v}$)	Properties
Eulerian method	$\vec{\hat{v}} = 0$	Computational mesh is fixed spatially; Additional process is needed for moving boundary; Algorithm is relatively simple because of no remeshing.
Lagrangian method	$\vec{\hat{v}} = \vec{v}$	Computational mesh moves corresponding to the velocity of fluid particles; Description of the moving boundary is relatively simple, the distortion of mesh becomes serious in case of large movement; Remeshing is required on each time step.
ALE (Arbitrary Lagrangian-Eulerian method)	$\vec{\hat{v}} \neq \vec{v}$	Computational mesh could move independently to the velocity of fluid particles; Description of the moving boundary is relatively simple, the distortion of mesh could be adjusted during simulation; Remeshing could be implemented only in the vicinity of moving boundary, thus relatively less cost is required.

1.2 About the simulation of turbulence

For engineering applications, the flows are mostly turbulent and they are highly unsteady in time and contain numerous vortices with the scales spanning widely. Generally, the turbulence energy transfers from large vortices to small ones and dissipate into heat at last. To generate the vortices of all scales, it is generally said that a grid number of $\mathcal{O}[Re^{9/4}]$ is necessary. Except for few limited cases, it is difficult to simulate all the vortices by using the capability and memory of current computers. Besides that, issues such as the preparation of initial conditions, discretization error and the chaos from the nonlinear make the accurate prediction of turbulence flow more impossible (梶島岳夫,1999).

Methods to carry out the simulation of turbulent flows include direct numerical simulation (DNS), large-eddy simulation (LES) and Reynolds-averaged Navier-Stokes simulation (RANS). For the DNS, the equations are discretized directly without performing any turbulence model, all vortices are supposed to be calculated directly. Because of an extremely high computational cost, DNS is usually used for obtaining basic data of turbulent flow phenomena, while for many engineering applications that are frequently turbulent at high Reynolds numbers, implementing DNS is quite difficult, although DNS has provided valuable results (for example, Moser, Kim, &Mansour, 1999) and is proved to be an effective method to investigate the turbulence besides the experiment. For the LES, on the other hand, only relative large eddies are computed while the effect of small ones is modeled. Filter function is employed to build the equations for discretization. The filter scale is usually set to be the same as the grid scale, thus the vortices are divided into the grid scale (GS) that is to be computed and the subgrid scale (SGS) that is to be modeled. The modeling methods for LES include the Smagorinsky model, the scale similarity model, the dynamic Smagorinsky model and so on. For the RANS, all vortices are modeled, the Reynolds averaging is implemented on the equations and the resulting unknown terms such as the Reynolds stress or diffusion flux in the Reynolds stress equations are closed by modeling. The eddy-viscosity model or the Reynolds stress equation model is adopted in the RANS calculation. Numerous achievements have been made on developing modeling methods for simulation of turbulence, however universal model that can be applied to every turbulent flow is still lacked (梶島岳夫, 1999).

1.3 Objectives of the present study

In the present study, applications of the immersed boundary method on the problems concerning rough or moving boundaries are carried out. For this purpose, a force feedback mechanism is added to an original program code “FrontFlow/Red” (小林敏雄, 2004; 畝村毅, 2003; Muto, Tsubokura, & Oshima, 2012; http://www.ciss.iis.u-tokyo.ac.jp/rss21/) that is introduced in Chapter 3 to simulate the movement in flow controlled by the hydrodynamic force. The performance of the immersed boundary method including the feedback mechanism is validated firstly and three-dimensional flow phenomenon in the interaction of the fluid and boundary is investigated, the present test cases include:

1. Validation case for the static immersed boundary: flow around a circular cylinder. Two cases with Reynolds number at 100 and 1000 are carried out, respectively. The obtained results are compared with those of other previous studies.
2. Validation case for the moving boundary involving the force feedback: single settling sphere particle in fluid. A sphere particle is set free statically and falls under the gravity effect and reaches a limit velocity when the gravity is balanced with the drag and buoyancy. The obtained results are compared with those of other previous studies.
3. Application of the immersed boundary method to build the rough structure in the study on the effect of an anisotropy-resolving subgrid-scale model on large eddy simulation predictions of turbulent open channel flow with wall roughness. The roughness on the bottom surface of channel is defined using the 'egg-carton' shaped-surface function (Bhaganagar et al., 2004). Grid dependency of the turbulence model (Abe, 2013) on the flow prediction near the rough surface is investigated in detail.
4. Three-dimensional simulation of a self-propelled fish-like body swimming in a channel where the fish model and movement are defined in a structured grid using the immersed boundary method. To satisfy the continuity inside the immersed boundary, a rigid-body system is adopted to build the fish model. A parametric study of kinematic factors is performed to study the hydrodynamics in fish swimming.

An introduction about the immersed boundary method is given in Chapter 2. Numerical simulation method is provided in Chapter 3. The details of the present computational cases are introduced in Chapter 4. The results are then discussed in Chapter 5, and the conclusions of the study are drawn in Chapter 6.

Chapter 2 Immersed boundary method

2.1 Introduction

As aforementioned, with applying the immersed boundary (IB) method, there is no need to modify the grid when carrying out the simulation with moving boundary, thus the corresponding computational time of remeshing can be saved. This is a considerable advantage especially when there is large displacement. The IB method is considered firstly to be proposed by Peskin (1972) to study the flow around heart valves where the boundary was replaced by a force and a semi-discrete analog of the δ function is introduced to link the representation of the boundary and fluid. An extending numerical study including the muscular heart wall was also carried out (Peskin, 1977). By now, many studies have been performed on this method as follows for examples. Iaccarino and Verzicco (2003) reported some results on the application of the IB method for turbulent flow simulations using both RANS and LES. Details of forcing definition and boundary treatment were illustrated. Tseng and Ferziger (2003) presented a second-order ghost-cell immersed boundary method for turbulent flows in complex geometries. In the study, the ease of implementation of this method in existing codes was demonstrated and the obtained computational results agreed well with the previous experimental and numerical results. Noor, Chern and Horng (2009) presented a direct-forcing IB technique to solve fluid-solid interaction problems. Taira and Colonius (2007) presented a formulation of the IB method modeled algebraically identical to the traditional fractional step method to simulate the flows over bodies with prescribed motion. In the study, regularization and interpolation operators were introduced to satisfy the no-slip constraint on the immersed surfaces, and the boundary force was regarded as the Lagrange multiplier like pressure. Slip and non-divergence-free components of the velocity field were removed by a projection in the approach. Fadlun et al. (2000) showed the effect of the interpolation of forcing over a grid on the accuracy of the scheme and applied the IB method to the flow inside an IC piston/cylinder assembly at high Reynolds number. Goldstein, Handler and Sirovich (1993) simulated the no-slip flow boundary using the IB method and determined the needed surface body force by a feedback scheme in which the velocity is used to build the formula. Balaras (2004) presented a method based on the direct forcing approach to carry out LES around complex boundaries on fixed Cartesian grids. In the study, a novel interpolation scheme and a method to deal with SGS fluxes near the interface were introduced. Abundant developments show the IB method a promising way to deal with complex and moving boundaries.

2.2 Category of the immersed boundary method

When applying the IB method, in general, the effect of the boundary on a flow is realized by imposing a body force through some forcing functions. Besides this, there is also another approach: the so-called cut-cell approach, in which new cells that conform to the shape of a body is generated by truncating the Cartesian cells at the immersed boundary (Bandringa, 2010). For the first approach, the Navier-Stokes equation for an incompressible viscos flow including the body force is as follows (Hirose, 2013):

$$\rho\left(\frac{\partial\vec{u}}{\partial t}+\vec{u}\cdot\nabla\vec{u}\right)+\nabla p-\mu\Delta\vec{u}=\vec{f}, \tag{2.1}$$

$$\nabla\cdot\vec{u}=0 \text{ in } \Omega_f \text{ and} \tag{2.2}$$

$$\vec{u}=\overrightarrow{u_\Gamma} \text{ on } \Gamma_b. \tag{2.3}$$

Where $\vec{f}$ is the body force, ρ is the density, μ is the kinematic viscosity, $\vec{u}$ and p are the velocity and the pressure, respectively. Γ_b denotes the boundary, Ω_b and Ω_f are the field of body and fluid, respectively, which are divided by the boundary Γ_b.

To calculate the body force, various methods have been proposed as shown in Iaccarino, & Verzicco (2003). The body-force functions can be generally divided into the continuous forcing and the discrete forcing approach (Mittal, & Iaccarino, 2005).

By using the continuous forcing approach, it is straightforward to insert a computing unit into an original program because it is independent of the spatial discretization. While smoothing of the forcing function causes unavailability of the sharp representation of the immersed boundary, which makes the continuous forcing approach not suitable for high Reynolds number flows (Bandringa, 2010). Also, with increasing Reynolds numbers, the proportion of the grid nodes inside the immersed boundary that require to solve the governing equations increases, this lowers the computational efficiency. Many applications of the methods belonging to this category are found in biological and multiphase flows where elastic boundaries are usually seen, for which the continuous forcing approach is attractive. While there are some challenges for applying this approach to flows with rigid boundaries for the forcing terms generally do not work well in the rigid limit (Mittal et al. 2005).

The direct forcing approach was reported (Fadlun et al., 2000; Balaras, 2004) firstly introduced by Mohd-Yusof (1997). In contrast to the continuous forcing approach, a sharp representation of the immersed boundary can be made by the direct forcing approach, which is necessary for high Reynolds number flows. Without using specified parameters in the forcing, no extra stability constraints are introduced for the immersed boundary (Bandringa, 2010). While the forcing procedure of this approach is highly associated with the discretization and the implementation is not as straightforward as the continuous forcing approach, the inclusion of the boundary motion could be more difficult (Mittal, 2005).

In the present study, the direct forcing approach is adopted to perform the simulations of flows with immersed boundaries inside because of its advantages such as the capability of a sharp representation of boundary that is required in some following cases where the Reynolds number may be relatively high. The immersed bodies could be static or moving in setting modes, and by introducing some feedback mechanism, movement in flow induced by the hydrodynamic force can also be simulated. An introduction of the direct forcing approach is given in Section 2.3.

2.3 Direct forcing approach

The non-dimensional incompressible Navier-Stokes equations including the body force $\vec{f}$ is as follows (Hirose, 2013):

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} + \nabla p - \frac{1}{Re} \Delta \vec{u} = \vec{f}, \tag{2.4}$$

$$\nabla \cdot \vec{u} = 0 \text{ in } \Omega_f \text{ and,} \tag{2.5}$$

$$\vec{u} = \overrightarrow{u_\Gamma} \text{ on } \Gamma_b. \tag{2.6}$$

To avoid any misunderstanding, it is noted that the denotations that are the same as those in Section 2.2 mean the non-dimensional values here.

The time discretization form of Eq. (2.4) is as:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = RHS_i + f_i, \tag{2.7}$$

where the superscript denotes the time step and subscript denotes the spatial nodes, and $RHS_i = -\vec{u} \cdot \nabla \vec{u} - \nabla p + \frac{1}{Re} \Delta \vec{u}$. Let V_{ib}^{n+1} is the desired velocity at n+1 step inside the immersed boundary, the body force f_i could then be obtained as:

$$f_i = \frac{V_{ib}^{n+1} - u_i^n}{\Delta t} - RHS_i. \tag{2.8}$$

Outside the immersed boundary, there is no body force. The calculation of body force $\vec{f}$ is concluded as:

$$\vec{f} = \begin{cases} (\vec{u} \cdot \nabla)\vec{u} + \nabla p - \frac{1}{Re} \Delta \vec{u} + \frac{1}{\Delta t}\left(\overrightarrow{V_{ib}} - \vec{u}^n\right), & near\ \Gamma_b\ or\ in\ \Omega_f; \\ 0, & elsewhere. \end{cases} \tag{2.9}$$

The detailed computational procedure is as (Balaras, 2004):

1. Calculate the intermediate $\vec{u}^*$ by the discretized Navier-Stokes equations without considering the body force term. The obtained $\vec{u}^*$ does not fulfill the IB boundary condition.
2. Calculate $\vec{f}^{n+1}$ by Eq. (2.9). The body force could be imposed on the grid nodes in the vicinity of the immersed boundary or all inside. The desired velocity $\overrightarrow{V_{ib}}$ could be determined straightforwardly or through interpolation approaches depending on the location of node.

3. Calculate the new $\vec{u}^*$ by the Navier-Stokes equations including the body force term, this new velocity fulfills the IB boundary condition.
4. Calculate the pressure by Poisson equation.
5. Update the velocity and pressure.
6. Go back to step 1.

By setting different desired velocity (V_{ib}^{n+1}) field, different boundary conditions can be yielded. Noted that although the fluid inside the immersed boundary is regarded as a "body" part in computations, it is still controlled by the continuity equation, therefore the desired velocity field should be chosen properly. If the desired velocity field does not satisfy the continuity condition, the calculation may not go well for some undesirable flow flux may arise on the boundary if the flow is incompressible. Considering this constraint, the desired velocity field is set as certain velocity field in a rigid body system for simplicity in the present study. For example, $V_{ib}^{n+1} = 0$ corresponds to a static boundary or body; $V_{ib}^{n+1} = V_0$ corresponds to a translational boundary or body; and $V_{ib}^{n+1} = \omega_0 r$ corresponds to a rotational boundary or body. Combination of these movement is also available. The definition of desired velocity field for different cases may change depending on the computational goals and is illustrated in Chapter 4.

Chapter 3 Numerical simulation method

3.1 Solver of simulation

The simulation in the present study is performed using the open-source-computational fluid dynamics (CFD)-code "FrontFlow/Red" (小林敏雄, 2004; 畝村毅, 2003; Muto, Tsubokura, & Oshima, 2012; http://www.ciss.iis.u-tokyo.ac.jp/rss21/), which is exploited in the project of "Revolutionary Simulation Software" (http://www.rss21.iis.u-tokyo.ac.jp/) having started in 2005 as a part of the next-generation IT program sponsored by the Ministry of Education, Culture, Sports, Science and Technology (MEXT).

3.1.1 Fundamental equations (小林敏雄, 2004)

The present software mainly deals with flows that are at low Mach number and incompressible, although computations concerning compressible flow is also possible in principle. Here, the mass-conservation and momentum-conservation equations involved in the present study are illustrated as follows.

Mass conservation equation

The mass conservation equation is as follows:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_j\right) = \rho_0, \tag{3.1}$$

where u_j is the velocity component in the x_j direction, ρ is the density, ρ_0 is the generation term. Equation (3.1) is generally for compressible flow, while in case of the incompressible flow, the fluid density is constant and Eq. (3.1) becomes $\nabla \cdot \vec{u} = 0$.

Momentum conservation equation

The momentum conservation equation is as follows:

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_i u_j) = -\frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} + f_i, \tag{3.2}$$

where f_i is the body force in the x_i direction and τ_{ij} is the viscous stress tensor. In case of Newtonian fluid, τ_{ij} can be expressed as

$$\tau_{ij} = -\left(\mu_v - \frac{2}{3}\mu\right)\delta_{ij}\frac{\partial u_k}{\partial x_k} - \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \tag{3.3}$$

where p is the pressure, μ is the viscosity coefficient, μ_v is the volume viscosity coefficient, δ_{ij} is the Kronecker delta. The pressure p coincides with the thermodynamic pressure when misalignment from the thermal equilibrium is not significant.

3.1.2 Boundary conditions (小林敏雄, 2004)

Physically, boundary conditions contain inflow, outflow, pressure, periodic and so on. These boundary conditions can basically be categorized as Dirichlet condition and Neumann condition mathematically, as expressed in the following.

Dirichlet condition: $\phi_{BC} = \phi_{give}$, (3.4)

Neumann condition: $\left(\frac{\partial \phi}{\partial \vec{n}}\right)_{BC} = const$, (3.5)

where $\partial/\partial\vec{n}$ denotes the gradient normal to the wall.

According to the grid density near the wall and the turbulence model adopted, the wall-function is sometimes used. Although physical fields involved in the boundary setting contain velocity, temperature, chemical species and so on, the main concern is on the velocity field in the present study.

Inflow condition at the inlet

Inflow condition can be expressed as follows,

$$\vec{v}_{BC} = \vec{v}_{give}. \tag{3.6}$$

Outflow condition at the outlet

Generally, the outflow condition is set so that the change of variables is smooth and little influence from downstream side to upstream side arises. The generation of reversed flow in the outflow condition may cause numerical instability in computation and make it hard to convergence. The outflow condition can be expressed as,

$$(\frac{\partial \vec{v}}{\partial \vec{n}})_{BC} = 0. \tag{3.7}$$

Symmetric and free-slip condition

The free-slip condition means that no wall shear stress or mass exchange on the wall, and the normal gradient of the velocity component parallel to the wall is zero. The symmetric and free-slip condition can be expressed as,

$$\vec{v}_{BC} \cdot \vec{n} = 0, \left[\frac{\partial}{\partial \vec{n}}\{\vec{v} - (\vec{v} \cdot \vec{n})\vec{n}\}\right] = 0. \tag{3.8}$$

No-slip condition on wall

The no-slip condition means the fluid is adhesive to the wall and moves at the same velocity as that of the wall. It can be expressed as,

$$\vec{v}_{BC} = \vec{v}_{wall}, \tag{3.9}$$

where $\vec{v}_{wall}$ is the velocity of the wall.

3.1.3 Mesh and arrangement of variables (小林敏雄, 2004)

The methods for the discretization of the NS equations include finite difference, finite volume and finite element methods. About the coordinate system for computation, there are such as Cartesian coordinate system, cylindrical coordinate system and generalized coordinate system. Structured and unstructured grids are possible to be used. For the arrangement of the variables, there are staggered arrangement and collocated arrangement.

In the present software, the Cartesian coordinate system is adopted with the unstructured grid system, where a structured grid is treated as a part of the unstructured grid. It adopts an unstructured finite-volume procedure with a vertex-centered type storage on a grid that is described later. The solver deals with four types of cell structures, which are tetrahedron, pyramid, prism and hexahedron, as well as the mixture of them. Figure 3.1 shows these cell structures.

The finite volume method is also called "the control volume method". For applying the unstructured grid, two approaches can be adopted in choosing the type of the control volume. One is to consider the cell constituted by a polyhedron as one control volume. By this method, the inspection surface is defined as the face of the cell constituted by the surrounded grid lines, on which the fluxes of physical quantities are considered. The variables are defined in the center of cells, this approach is called "the cell-center method". The other approach is to define the variables on vertices and build virtual inspection surfaces around the vertices to establish the control volumes, this approach is called "the vertex-center method". The coding methods for the two approaches are called "the face-base method" and "the edge-base method", respectively. The present software adopts the vertex-center method. Figure 3.2 shows the schematic pictures for these two considerations.

3.1.4 Convection term (小林敏雄, 2004)

For the differential scheme, various options are provided in FrontFlow/red, such as the first order upwind difference, the second order central difference and the third order TVD schemes. Here, the first order upwind difference scheme and the second order central difference scheme for the interpolation of values on the surface of a control volume (SF) are illustrated.

First order upwind difference scheme

The value on the surface of a control volume is specified using the value of the control volume in the upstream as following,

$$\phi_{SF} = \phi_A \max\left(\frac{\vec{S}_{SF}\cdot\vec{u}_{SF}}{|\vec{S}_{SF}\cdot\vec{u}_{SF}|}, 0\right) - \phi_B \min\left(\frac{\vec{S}_{SF}\cdot\vec{u}_{SF}}{|\vec{S}_{SF}\cdot\vec{u}_{SF}|}, 0\right), \tag{3.10}$$

where ϕ_{SF} is the value on the surface that is shared by the control volume A and B, ϕ_A and ϕ_B are the values at control volume A and B, respectively. $\vec{S}_{SF}$ denotes the area vector and $\vec{u}_{SF}$ denotes the velocity vector on the surface.

Second order central difference scheme

The second order central difference scheme uses the distance-weighted average value to specify the value on the surface of a control volume, which is expressed as follows,

$$\phi_{SF} = w_A\phi_A + w_B\phi_B, \tag{3.11}$$

where w_A and w_B are the weights of distance from the shared surface to the control volume A and B, respectively.

3.1.5 Gradient and diffusion term (小林敏雄, 2004)

The gradient for the control volume A is calculated using the Gauss method as follows,

$$\nabla\phi_A = \frac{\sum_{i=1}^{SF\ on\ CV_A} \vec{S}_{SFi}\phi}{\delta\Omega}, \tag{3.12}$$

where δΩ denotes the volume of the control volume. The gradient on the surface of a control volume is calculated using the Defer-Correction-Term method proposed by Muzaferija (1994) as follows,

$$\nabla\phi_{SF} = \overline{\nabla\phi}_{SF} + \frac{\vec{r}_{AB}}{|\vec{r}_{AB}|^2}[(\phi_A - \phi_B) - \vec{r}_{AB}\cdot\overline{\nabla\phi}_{SF}], \tag{3.13}$$

where $\overline{\nabla\phi}_{SF}$ is the weighted average of the gradients of the control volume A and B that share the surface, $\vec{r}_{AB}$ is the vector from vertex A to B.

3.1.6 Time integration

Generally, in a problem concerning the incompressible flow, no time difference of the pressure is included. Therefore, to determine the distribution of the pressure, it is necessary to combine the mass conservation equation with the other governing equations to make it always fulfilled.

Implicit methods frequently run an internal loop in updating the velocity and pressure to satisfy the continuity equation by solving the discretized equation of the velocity and pressure correction. The semi-implicit method for pressure-linked equation (SIMPLE) method is a representative example as well as its improved versions such as the SIMPLER method, the SIMPLEST method, the SIMPLEC method, the PISO method and the FIMOSE method. On the other hand, the marker and cell (MAC) method, as an explicit method, is developed and widely used for its simplicity and ease to apply on the unsteady problems. In the present software, a simplified Poisson equation is applied based on the MAC method. Also, the SMAC (simplified MAC) method is improved so that both implicit and explicit methods can be used. Also, the second-order Crank-Nicolson method, the second-order Adams-Bashforth explicit method, the third-order Adams-Moulton implicit method and the forth-order Runge-Kutta method are also available (小林敏雄, 2004).

Crank-Nicolson method (荒川忠一, 1994)

For simplification, the explanation is given by taking an example of one-dimensional unsteady heat conduction equation in a nondimensionalized form as follows:

$$\frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial x^2}. \tag{3.14}$$

If the right-hand side term is computed using the averaged values of both the present (j) and next (j+1) steps, the equation could be discretized in an implicit form. When the weights of the two values are both chosen as 0.5, it is called the Crank-Nicolson implicit method, and the discretized equation is as follows

$$\frac{T_{i,j+1}-T_{i,j}}{\delta t} = \frac{1}{2}\left(\frac{T_{i+1,j+1}-2T_{i,j+1}+T_{i-1,j+1}}{\delta x^2} + \frac{T_{i+1,j}-2T_{i,j}+T_{i-1,j}}{\delta x^2}\right), \tag{3.15}$$

then

$$-rT_{i-1,j+1} + (2+2r)T_{i,j+1} - rT_{i+1,j+1} = rT_{i-1,j} + (2-2r)T_{i,j} + rT_{i+1,j}, \quad (3.16)$$

where $r = \frac{\delta t}{\delta x^2}$. Compared to the explicit approaches, larger time step can be set for a stable computation in the implicit method, where effective time marching can thus be achieved. However, too large time step may cause a decrease of accuracy even though the computational stability is maintained.

MAC method (荒川忠一, 1994)

The explanation of MAC method is introduced as follows. The equations of motion and continuity for an incompressible viscous flow are shown as

$$\frac{\partial \vec{v}}{\partial t} = -\nabla p + \frac{1}{Re}\nabla^2 \vec{v} - \nabla \cdot (\vec{v}\vec{v}), \quad (3.17)$$

$$\nabla \cdot \vec{v} = 0. \quad (3.18)$$

Rewrite the right-hand side of the Eq. (3.17) by a function f of $\vec{v}$ and p,

$$\frac{\partial \vec{v}}{\partial t} = f(\vec{v}, p), \quad f(\vec{v}, p) = -\nabla p + \frac{1}{Re}\nabla^2 \vec{v} - \nabla \cdot (\vec{v}\vec{v}). \quad (3.19)$$

Discretizing the equation by a forward difference,

$$\frac{\vec{v}^{n+1} - \vec{v}^{n}}{\Delta t} = f(\vec{v}^{n}, p). \quad (3.20)$$

Performing the divergence for Eq. (3.20), the following equation could be obtained,

$$\frac{\nabla \cdot \vec{v}^{n+1} - \nabla \cdot \vec{v}^{n}}{\Delta t} = \nabla \cdot f(\vec{v}^{n}, p^{n}) = \nabla \cdot f(\vec{v}^{n}, 0) + \nabla \cdot f(0, p^{n}) = \nabla \cdot f(\vec{v}^{n}, 0) - \nabla^2 p. \quad (3.21)$$

To fulfill the continuity equation at time step n+1,

$$\nabla \cdot \vec{v}^{n+1} = 0 \tag{3.22}$$

is yielded, the Poisson equation for the pressure can then be obtained as

$$\nabla^2 p = \frac{\nabla \cdot \vec{v}^n}{\Delta t} + \nabla \cdot f(\vec{v}^n, 0) = \frac{\nabla \cdot \vec{v}^n}{\Delta t} + \frac{1}{Re}\nabla^2(\nabla \cdot \vec{v})^n - \nabla \cdot (\nabla \cdot (\vec{v}\vec{v})^n). \tag{3.23}$$

The procedure for the computation is concluded as follows. From the velocity at time step n, the pressure is calculated by Eq. (3.23), the velocity at time step n+1 is then calculated explicitly by Eq. (3.21) as

$$\vec{v}^{n+1} = \vec{v}^n + \Delta t \cdot f(\vec{v}^n, p^n). \tag{3.24}$$

Iterating the above operation, since the velocity at new time step is obtained, the pressure at time step n+1 can then be calculated by using the Poisson equation, the time marching approach is then established. As noted, the time marching is explicit, therefore the time step can not be set too large. Although solving the Poisson equation is popular to obtain the pressure distribution to fulfill the continuity equation, it is also known to cost much for solving it.

SMAC explicit method (小林敏雄, 2004)

As the improvement of the MAC method, in SMAC, a simplified Poisson equation is adopted and the time marching is divided into two steps to improve the accuracy. This part describes the SMAC explicit method. The momentum conservation equation is shown as follows,

$$\frac{\rho^*\vec{v}^* - \rho^n\vec{v}^n}{\Delta t} = -\vec{R}_{CD}^n - \nabla p^n, \tag{3.25}$$

$$\frac{\rho^{n+1}\vec{v}^{n+1} - \rho^*\vec{v}^*}{\Delta t} = -\nabla \delta p, \tag{3.26}$$

where the superscript "*" means the intermediate value, $\vec{R}_{CD}^{n}$ means the sum of the convection term and the diffusion term, and $\delta p = p^{n+1} - p^{n}$ is the pressure correction. Suppose the mass conservation is fulfilled at the new time step (n+1), the Poisson equation of the pressure correction (δp) is then obtained by performing the divergence on Eq. (3.26) as follows,

$$\frac{1}{\Delta t}\left(-\nabla \cdot (\rho^{*}\vec{v}^{*})\right) = -\nabla^{2}\delta p. \tag{3.27}$$

By solving this equation, the pressure correction can be calculated, and the velocity at new time step (n+1) is obtained as

$$\rho^{n+1}\vec{v}^{n+1} = -\Delta t \nabla \delta p + \rho^{*}\vec{v}^{*}. \tag{3.28}$$

The computational procedure for the explicit SMAC method is concluded as:

1. Calculate the intermediate velocity $\vec{v}^{*}$ by Eq. (3.25);
2. Calculate the new pressure $p^{n+1} = p^{n} + \delta p$ by solving the Poisson equation of the pressure correction δp;
3. Calculate the new velocity $\vec{v}^{n+1}$ by Eq. (3.28);
4. Go to the next time step.

SMAC Euler implicit method (小林敏雄, 2004)

SMAC method is originally based on the explicit approach, while in the present software, an implicit approach is also available, the explanation of which is given below.

Based on the two-step procedure and replacing the explicit method in Eq. (3.25) by an implicit method, the following equations can be obtained,

$$\frac{\rho^{*m+1}\vec{v}^{*m+1} - \rho^{n}\vec{v}^{n}}{\Delta t} = -\vec{R}_{CD}^{*m+1} + \nabla p^{m}, \tag{3.29}$$

$$\frac{\rho^{m+1}\vec{v}^{m+1} - \rho^{*m+1}\vec{v}^{*m+1}}{\Delta t} = -\nabla \delta p, \tag{3.30}$$

where m means internal iteration number within one time step, $\vec{R}_{CD}^{m}$ is the sum of the convection term and the diffusion term, and

$$\rho^{*m+1}\vec{v}^{*m+1} = \rho^{m}\vec{v}^{m} + \rho'\vec{v}', \tag{3.31}$$

$$\delta p = p^{m+1} - p^{m}, \tag{3.32}$$

where $\rho'\vec{v}'$ denotes the velocity correction in the first stage, δp is the pressure correction. Performing the divergence on both sides of the Eq. (3.30), $\nabla \cdot (\rho^{m+1}\vec{v}^{m+1})$ is supposed to satisfy the continuity equation, the Poisson equation of pressure correction δp is then obtained. Iteration within one time step is carried out until the pressure and velocity corrections are small enough, and the newest iteration value (m+1) is taken as the new value at n+1 time step.

The computational procedure of the SMAC Euler implicit method is concluded as follows:

1. Calculate the intermediate velocity by Eq. (3.29);
2. Calculate $p^{m+1} = p^{m} + \delta p$ by solving the Poisson equation of the pressure correction;
3. Calculate the new velocity $\vec{v}^{m+1}$ by Eq. (3.30);
4. If the pressure correction is small enough and $\vec{v}^{m+1}$ satisfies the continuity equation, the convergence is considered to be achieved.
5. If the convergence is achieved at step 4, go to the next step. Otherwise go back to step 1 and perform the next internal iteration.

3.2 Application of the direct forcing approach

Figure 3.3 shows the schematic computational procedure of FrontFlow/red (Hirose, 2013). According to Eq. (2.8), in order to calculate the body force f_i, it is necessary to obtain the desired velocity V_{ib}^{n+1} and RHS_i. The desired velocity V_{ib}^{n+1} could be determined by setting beforehand or calculating through the feedback mechanism, while the RHS_i is the intermediate value during calculating $\vec{u}^{*}$, which could then be obtained by implementing a loop during the stage of solving the Navier-Stokes equations without the body force term. Once the body force is obtained, the new velocity could be calculated by the Navier-Stokes equations including the body force term. Figure 3.4 shows the schematic computational procedure including the direct forcing mechanism (Hirose, 2013). To fulfill the continuity equation (2.5), the Poisson equation is solved to update the velocity and pressure.

To carry out hydrodynamic movement in fluid, a force feedback mechanism is added into the original program. The feedback mechanism combines certain kinematic definition with the dynamic equations and uses the IB body forces as input to update the desired velocity. The corresponding schematic computational procedure is shown in Figure 3.5. To satisfy the continuity in the immersed boundary, a rigid body system is to be adopted in the present study. The involved rigid body dynamics in the study for a single rigid body is given as follows.

Center of mass and moment of inertia

A reference point $\vec{r}$ is known as the center of mass if it satisfies the condition

$$\sum_{i=1}^{N} m_i(\vec{r_i} - \vec{r}) = 0. \tag{3.33}$$

The moment of inertia $[I_R]$ related to the center of mass is defined as

$$[I_R] = -\sum_{i=1}^{N} m_i[\vec{r_i} - \vec{r}]^2, \tag{3.34}$$

where $[\vec{r_i} - \vec{r}] = \begin{bmatrix} 0 & -\Delta r_z & \Delta r_y \\ \Delta r_z & 0 & -\Delta r_x \\ -\Delta r_y & \Delta r_x & 0 \end{bmatrix}$.

Kinematics and dynamics

The velocity of certain point of the rigid body $\vec{u_i}$ is expressed by the velocity of the center of mass $\vec{u}$ and the angular velocity of the body $\vec{\omega}$ as

$$\vec{u_i} = \vec{u} + \vec{\omega} \times (\vec{r_i} - \vec{r}). \tag{3.35}$$

The dynamic equations of force $\vec{F}$ and torque $\vec{T}$ known as Newton's second law of motion for a rigid body take the form as

$$\vec{F} = m\vec{a}, \tag{3.36}$$

$$\vec{T} = [I_R]\vec{\alpha} + \vec{\omega} \times [I_R]\vec{\omega}, \tag{3.37}$$

where $\vec{a} = d\vec{u}/dt$, $\vec{\alpha} = d\vec{\omega}/dt$. When applying to the cases, some assumptions are made to simplify the computation of the dynamics. The details can be referred to Chapter 4.

An explicit scheme is adopted to perform the force feedback of dynamics. The IB force and torque at step *n* is taken as the input to update the movement of the rigid body using the above equations. The obtained locations and velocities of the involved vertices are then inputted into the IB subroutine to calculate the new IB force and torque at step *n*+1. The procedure is concluded taking the example of a single particle of radius of r_0 as follows.

1. Update the location $\vec{r}$ and orientation $\vec{\theta}$ of the particle by $\vec{r}^{n+1} = \vec{r}^n + \vec{u}^n \Delta t$ and $\vec{\theta}^{n+1} = \vec{\theta}^n + \vec{\omega}^n \Delta t$, respectively;
2. Calculate the acceleration of velocity ($\vec{a}^n$) and angular velocity ($\vec{\alpha}^n$) of the particle by Eqs. (3.36) and (3.37) using $\vec{F}^n$ and $\vec{T}^n$ as input;
3. Update the velocity and angular velocity of the particle by $\vec{u}^{n+1} = \vec{u}^n + \vec{a}^n \Delta t$ and $\vec{\omega}^{n+1} = \vec{\omega}^n + \vec{\alpha}^n \Delta t$, respectively;
4. Update the velocity of each vertex (i) by Eq. (3.35), obtain the $\vec{u}_i^{n+1}$;
5. Output new location $\vec{r}^{n+1}$ and velocity $\vec{u}_i^{n+1}$ for updating the IB force to obtain $\vec{F}^{n+1}$ and $\vec{T}^{n+1}$. In case of a single particle, it is enough to determine the new boundary with only $\vec{r}^{n+1}$ by judging whether a vertex at $\vec{r}_i$ satisfies $|\vec{r}_i - \vec{r}| \leq r_0$. For other cases, the orientation $\vec{\theta}^{n+1}$ may also be needed.

The computational grid is made by the Gridgen (Gridgen) that consists of vertices and edges that connect the vertices. For the identification of the immersed region, the edge-based loop is adopted. The shape of the immersed boundary is defined using some shape function. If both the two sides of an edge fulfill the shape function, the corresponding two vertices are identified as inside points. If only one side fulfills the shape function, the vertex that fulfills the function is taken as “Ghost cell” and the other one is identified as an outside point, the vertices in the vicinity of the immersed boundary could then be identified and may be used for interpolation. If both the two sides do not fulfill the shape function, the corresponding two vertices are identified as outside points. Note that in the present application, no interpolation for the velocity near the boundary is performed considering that the Reynolds numbers in the involved cases are relatively low so that the uneven boundary is thought not to bring significant influence on the computational accuracy compared to other factors, while the implementation of interpolation may cause an unstable problem.

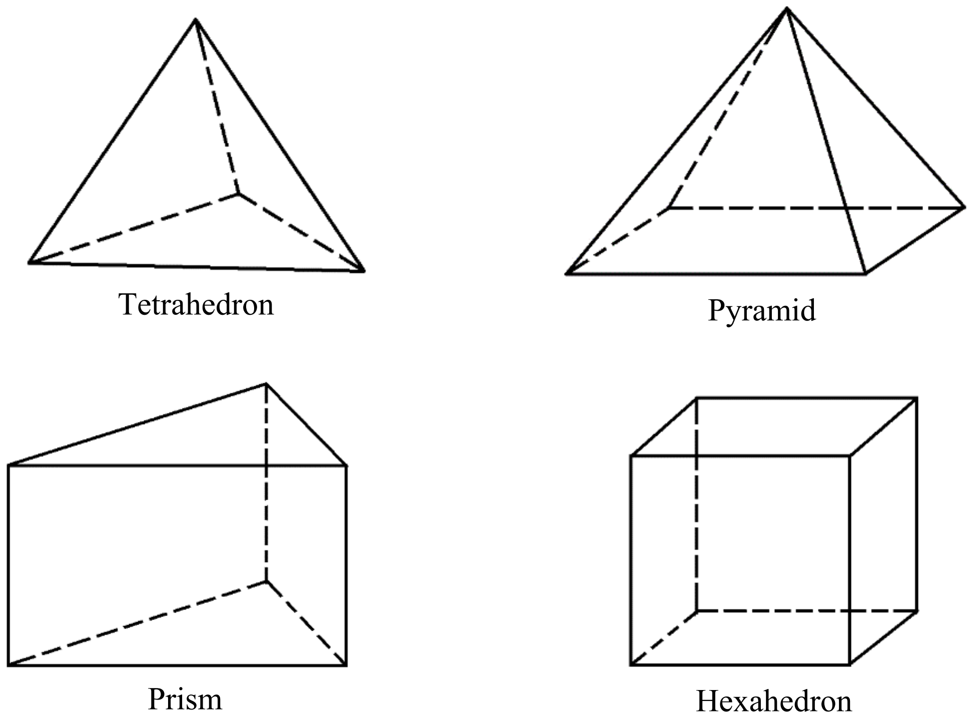


Figure 3.1: The category of the cell structures.

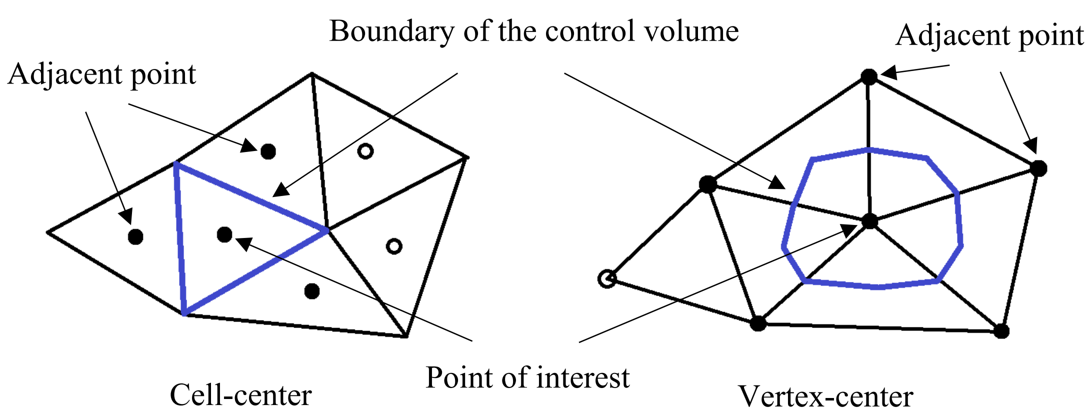


Figure 3.2: Schematic pictures of cell-center method and vertex-center method.

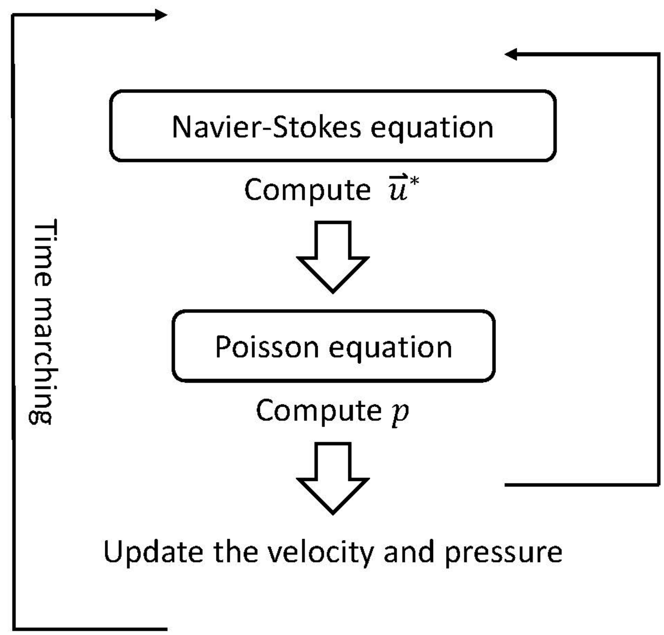


Figure 3.3: Schematic computational procedure of FrontFlow/red.

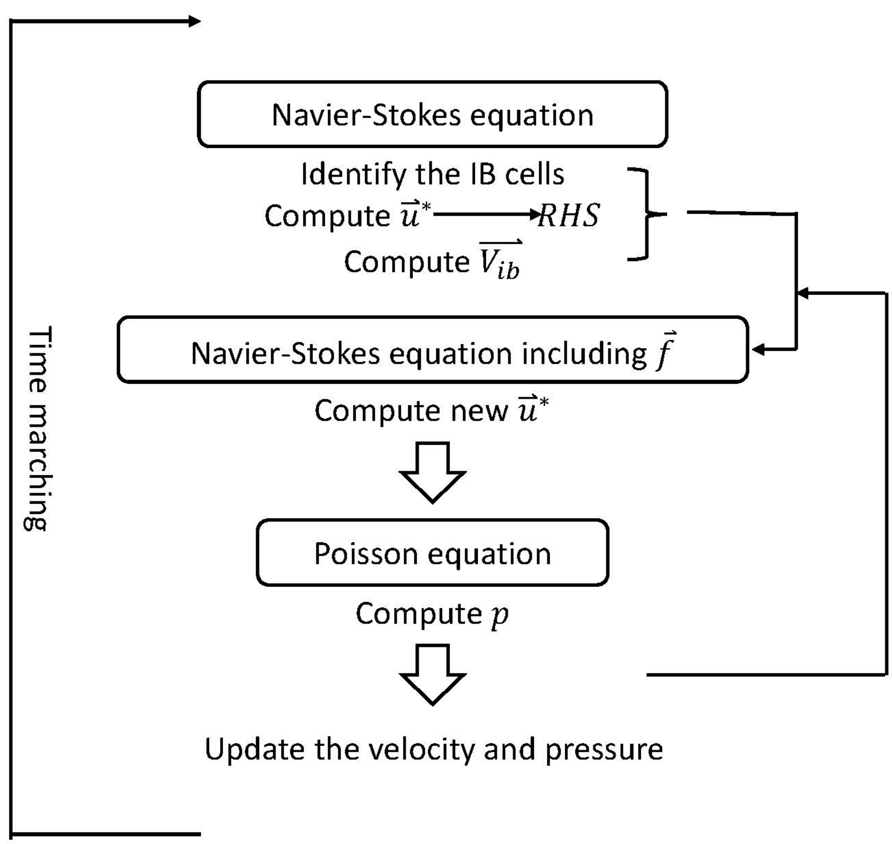


Figure 3.4: Schematic computational procedure including the IB mechanism.

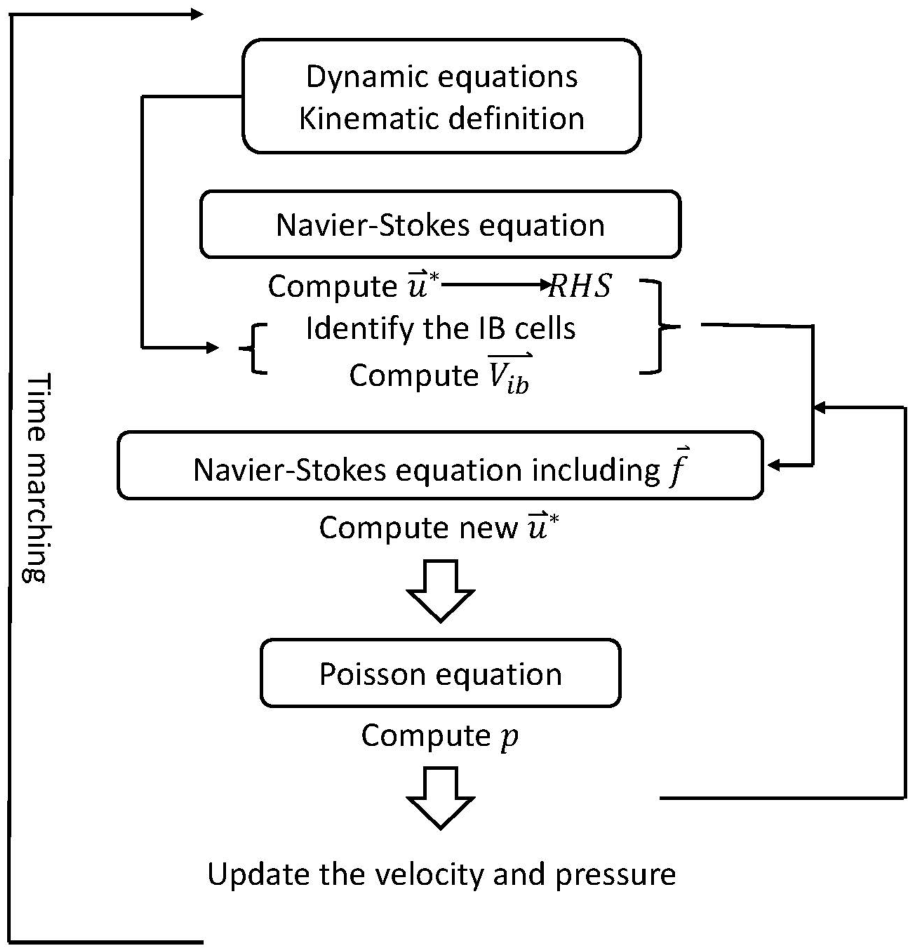


Figure 3.5: Schematic computational procedure including the IB and feedback mechanism.

Chapter 4 Computational cases

4.1 Introduction

In the present study, the direct forcing approach is firstly validated by performing the simulation of flow around a circular cylinder where the influence of the boundary is imposed by the IB method in the region of a normal structured grid and comparing the obtained results with those of previous studies. Furthermore, a dynamic feedback mechanism is inserted into the original program including the IB application so that the movement in flow controlled by the hydrodynamic force other than set in advance could be realized. To validate the program concerning the newly added feedback mechanism, a simulation of settling sphere particle in fluid is carried out and the obtained limit vertical velocity is compared with those of previous experimental and numerical results, the description of the computational settings and conditions is given in 4.2.

Based on the confirmation of the feasibility through the validation cases, the IB method is applied to the study of the turbulent flow in a channel with rough wall on one side to build the roughness, and then the study of a fish swimming performance where a fish-like model is made by the IB method and a self-propelled swimming movement is realized. By the IB method, the immersed body is described by identifying whether a grid vertex is inside the immersed boundary, the building of complex structures is convenient especially when there is a definition formula for the surface of structure. In the study of turbulent flow in rough channel, the IB method is adopted to build the static rough structure in the LES and the grid dependency of the turbulence model is investigated in detail. As an application combining the IB method and the feedback mechanism, a fish-like model is introduced in the present fish swimming study and a self-propelled swimming process is performed to carry out a parametric study of the kinematic. The details about the computational settings and conditions of these cases are shown in the following sections.

4.2 Validation cases

4.2.1 Flow around a circular cylinder

The present computational program, including the direct forcing approach, is first validated by performing simulations of flow around a circular cylinder. The Reynolds number is defined as $\mathrm{Re} = U_\infty D/\nu$, where D is the diameter of the cylinder, U_∞ is the far field velocity. The simulations are carried out at $\mathrm{Re} = 100$ and 1000, and the obtained results are compared with relevant numerical and experimental results by previous studies. For the case of $\mathrm{Re} = 100$, the computational domain is set to $l_x \times l_y \times l_z = 27D \times 20D \times 0.06D$ with grid nodes of $N_x \times N_y \times N_z = 265 \times 160 \times 3$. Note that the three-dimensional domain was made just for using the present program code because the flow for this test case is known to be basically two dimensional. For the case of $\mathrm{Re} = 1000$, the flow is known to become highly three dimensional and thus the computational domain is extended in the spanwise direction as $l_x \times l_y \times l_z = 27D \times 20D \times 2D$ with grid nodes of $N_x \times N_y \times N_z = 265 \times 160 \times 50$. Note that the grid plane perpendicular to the spanwise direction is the same for these two cases. This grid plane perpendicular to the spanwise direction is shown in Fig. 4.1, as well as the partial magnification near the cylinder. A resolution of 50 cells was used for the diameter of the cylinder. About the boundary conditions, the inflow and outflow conditions are applied at the inlet and outlet in the streamwise direction, respectively, the periodic condition is specified in the spanwise direction, and the free-slip condition is applied in the vertical direction.

The third-order upwind difference scheme is used to discretize the spatial derivatives. The time marching is based on the fractional step method, where the first-order Euler implicit scheme is used to derive the equations for the velocity. The coupling of the velocity and the pressure fields is based on the SMAC method. Note that this combination of the spatial and time-marching schemes was adopted mainly for the validation of those used in the fish-swimming simulation that is described in Section 4.4.

4.2.2 Single settling sphere particle in fluid

A single settling sphere particle in fluid is simulated to validate the program including both the IB method and the dynamic feedback mechanism. Figure 4.2 shows the schematic diagram of the simulation. A particle is subject to the gravity in an enclosed vertical channel and reaches a limit velocity when the gravity balances with the drag and buoyancy. When the particle reaches the bottom of the channel, a buffer force is imposed to decelerate the particle. To compare the obtained results with previous ones, the process from the start to the balance of reaching the limit velocity is mainly concerned in this study. The fluid density is $\rho = 1000\text{kg}/m^3$, the kinematic viscosity is $\upsilon = 9.0 \times 10^{-7} m^2/\text{s}$. The diameter of the particle is $\text{D} = 0.0005\text{m}$ and its density is set to $\rho_p = 2560\text{kg}/m^3$. The dimension of the computational field is $\text{x} \times \text{y} \times \text{z} = 0.004\text{m} \times 0.02\text{m} \times 0.004\text{m}$ with grid nodes of $N_x \times N_y \times N_z = 120 \times 600 \times 120$ and the wall condition is applied for all the boundaries. The center of the particle is initially located at the location of (0.002, 0.017, 0.002) and the static initial velocity is set as $\overrightarrow{u_0} = 0$. The main concern in the validation is the translational movement of the particle even though rotation is also simulated, which is considered not obvious. The obtained limit velocity is compared to the results shown by Mordant and Pinton (2000) and Luo et al. (2007).

The third-order upwind difference scheme is used to discretize the spatial derivatives. The time marching is based on the fractional step method, where the first-order Euler implicit scheme is used to derive the equations for the velocity. The coupling of the velocity and the pressure fields is based on the SMAC method. As is similar to the discussion in Section 4.4.1, concerning the use of the third-order upwind spatial scheme and the first-order time marching method, it is necessary to confirm their effects on the computational results. This issue is properly assessed and detailed discussion is provided in the results shown later.

4.3 Turbulent open channel flow with wall roughness

4.3.1 Background

Turbulent flow in a channel with surface roughness is very common in many engineering applications. Besides roughness brought by structures or materials, vapor bubbles that are attached to the wall in subcooled flow boiling can also resemble near-hemispherical roughness elements (Chatzikyriakou et al., 2015). A flow over a rough wall can be divided into two parts: an inner layer that is directly affected by the roughness elements, and an outer layer that is away from the wall where the influence of roughness is relatively weak. Different shapes or distributions of roughness elements may bring various influences on turbulence features from the near-wall region to the central flow. To date, abundant numerical and experimental results are available concerning this issue, as presented in the following for examples.

Direct numerical simulation (DNS) of turbulent incompressible flow in a plane channel with one of its walls covered with regular three-dimensional roughness elements (called ‘egg carton’-shaped surface) was performed by Bhaganagar et al. (2004). From the results, significant differences in processes associated with the wall-normal fluxes of the Reynolds shear stress and turbulence energy were demonstrated using the higher-order moments and the energy budgets between the smooth-wall and rough-wall sides. Chatzikyriakou et al. (2015) performed numerical simulations of turbulent channel flow with smooth walls and walls featuring rough hemispherical elements using both DNS and large eddy simulation (LES). In their research, a parametric study was conducted including the effect of shear Reynolds number, normalized roughness height and relative roughness spacing. The effect of the distribution pattern (regular square lattice vs. random pattern) of these rough elements on the walls was also verified. Shamloo et al. (2015) analyzed the effects of roughness density and flow submergence on flow features in open channel using LES. Jin et al. (2015) used both a finite volume method and a Lattice Boltzmann method to determine the detailed structures of turbulent flow through channels with smooth and rough walls. Marchis and Napoli (2012) investigated the effects of irregular 2D and 3D roughness on the turbulence by performing a wall-resolved LES of fully developed turbulent channel flows over two different rough surfaces. In their research, the roughness function obtained with 3D roughness is larger than that with a 2D one for the same mean height. Andersson et al. (2014) investigated the validity in assuming the roughness can be represented by a numerical model through performing Reynolds-averaged Navier-Stokes simulations over a surface with a large roughness and showed the importance of an appropriate surface description. Bailon-Cuba et al. (2009) performed DNS of a turbulent channel flow with 2D wedges of random height on the bottom wall. A DNS study of channel flow with rough walls comprising staggered arrays of cubes was presented by Leonardi and Castro (2010). Nagano et al. (2004) performed a DNS of heat transfer in turbulent channel flows with transverse-rib roughness to determine the effects of roughness on the statistical quantities of the velocity and thermal fields.

An experimental study of fully developed turbulent channel flow and an adverse pressure gradient turbulent channel flow over smooth and rough walls comprising two-dimensional square ribs was performed by Tsikata and Tachie (2013). Schultz and Flack (2007) presented experimental measurements for roughness-wall boundary layers covering a Reynolds-number range from hydraulically smooth to fully rough flow regimes. Keirsbulck et al. (2001) investigated the effect of roughness wall in a zero pressure-gradient turbulent boundary layer using hot-wire anemometry. Communication between the wall region and the outer region of a turbulent boundary layer was shown to be present. The experimental evidence provided by Krogstad and Antonia (1999) indicated that the surface geometry significantly affects the turbulent characteristics of the flow, although the two rough surfaces were designed to have nominally the same effect on the mean velocity profile.

LES is well known as a useful way to predict turbulent flow. When LES is applied, the grid-scale (GS) eddies are directly resolved and the subgrid-scale (SGS) eddies need to be modeled. The most widely used SGS model is the Smagorinsky model (Smagorinsky, 1963). In order to overcome the drawback of the Smagorinsky model, Germano et al. (1991) proposed an SGS eddy viscosity model in which a single universal coefficient is replaced by a dynamically computed value. The model coefficient was yielded based on the stresses obtained by the grid and test filters using the same concept of the Smagorinsky model for the anisotropic parts of SGS stresses. The method of evaluating the model coefficient was modified by Lilly (1992) using a least-square approach. Zang et al. (1993) modified the base model in (Germano et al., 1991) by employing the mixed model of Bardina et al. (1983), in which the modified Leonard term was explicitly calculated in the modified model, leading to much smaller model coefficient being obtained. Vreman et al. (1994) proposed a modification of the turbulent stress expression using alternative filter definition in the above dynamic mixed model (Zang et al., 1993) for stricter mathematical consistency. Recently, with a different perspective, an anisotropy-resolving subgrid-scale (SGS) model for LES was proposed by Abe (2013). In this model, the SGS-stress expression is constructed by combining an isotropic eddy-viscosity model with an extra anisotropic term (EAT). Applying the model to several fundamental test cases indicated the basic capability of this SGS modeling concept, particularly for coarser grid-resolutions. If this model also works well for predicting turbulent wall-shear flows with wall roughness, a considerable reduction in computational cost for LES is expected.

Hence, a LES of turbulent flow in a channel, with a rough wall on one side and a free surface on the other, is performed by adopting this anisotropy-resolving SGS model of Abe (2013). The primary concern of this study is how the present anisotropic SGS model can detail properly the effects of roughness on the mean velocity and turbulent stresses for coarser grid resolutions. The results obtained are compared with those of DNSs for both smooth and rough walls. The present DNSs are first validated using another reference DNS data for the smooth case. To investigate the grid dependency of the LES results resulting from the present SGS model for turbulence with wall roughness, three grid resolutions are tested for the above open plane channel flow under the same definition of roughness configuration. The immersed boundary method is used to define the roughness placed on the bottom surface. A detailed description of the turbulence model adopted for this study is given in Section 4.3.2. The computational conditions and roughness model are introduced in Section 4.3.3.

4.3.2 Turbulence model

The governing equations of LES for incompressible flow can be written as

$$\frac{\partial \bar{U}_i}{\partial x_i} = 0, \quad \frac{D\bar{U}_i}{Dt} = -\frac{1}{\rho}\frac{\partial \bar{P}}{\partial x_i} + \frac{\partial}{\partial x_j}\{\nu\left(\frac{\partial \bar{U}_i}{\partial x_j} + \frac{\partial \bar{U}_j}{\partial x_i}\right) - \tau_{ij}\}, \tag{4.1}$$

where $\overline{(..)}$ denotes a filtered value, and ρ , $\bar{P}$, $\bar{U}_i$ and ν denote density, filtered static pressure, filtered velocity and kinematic viscosity, respectively. Note that since incompressible flow is considered, the viscous term can also be written as $\nu\frac{\partial^2 \bar{U}_i}{\partial x_j \partial x_j}$. The SGS-stress tensor τ_{ij} is defined as $\tau_{ij} = \overline{U_i U_j} - \bar{U}_i \bar{U}_j$. For modeling τ_{ij}, a linear eddy-viscosity model (EVM), that is often used, takes the canonical form

$$\tau_{ij}^a = -2\nu_{SGS} S_{ij}, \quad S_{ij} = \frac{1}{2}\left(\frac{\partial \bar{U}_i}{\partial x_j} + \frac{\partial \bar{U}_j}{\partial x_i}\right), \tag{4.2}$$

where S_{ij} denotes the resolved strain-rate tensor, and $\tau_{ij}^{a} = \tau_{ij} - \tau_{kk}\delta_{ij}/3$. In the present simulations, an SGS-stress model is constructed by combining a representative isotropic EVM with an EAT proposed by Abe (2013). Its basic formulation is as follows,

$$\tau_{ij} = \frac{2}{3}k_{SGS}\delta_{ij} - 2\nu_{SGS}S_{ij} + 2k_{SGS}b_{ij}^{SGS}, \tag{4.3}$$

where the anisotropy tensor b_{ij}^{SGS} is defined as

$$b_{ij}^{SGS} = \frac{\tau'_{ij} - (-2\nu' S_{ij})}{\tau'_{kk} - (-2\nu' S_{kk})} - \frac{1}{3}\delta_{ij} = \frac{R'_{ij}}{\tau'_{kk}} \tag{4.4}$$

with $\tau'_{ij} = C_B(\overline{U}_i - \widehat{\overline{U}}_i)(\overline{U}_j - \widehat{\overline{U}}_j)$, a well-known 'scale-similarity model' proposed by Bardina et al. (1980); $\widehat{(..)}$ denotes the test-filtering operator. The equivalent eddy viscosity is modeled as

$$\nu' = -\tau'^{a}_{ij}S_{ij}/(2S_{mn}S_{mn}), \tag{4.5}$$

where $\tau'^{a}_{ij} = \tau'_{ij} - \tau'_{kk}\delta_{ij}/3$. In Eq. (4.4), $R'_{ij} = \tau'^{a}_{ij} - (-2\nu' S_{ij})$ is a 'SGS-stress anisotropy term' that is evaluated by subtracting an EVM form from the original Bardina model. In Eq. (4.3), $k_{SGS} = \tau_{kk}/2$ is the SGS turbulent kinetic energy, and the SGS viscosity ν_{SGS} is modeled as

$$\nu_{SGS} = C_{SGS}f_{SGS}\sqrt{k_{SGS}}\Delta, \qquad f_{SGS} = 1 - \exp\{-(\frac{y'_{\varepsilon}}{A_0})^{4/3}\}. \tag{4.6}$$

The constants of the model C_{SGS} and A_0 are set to $C_{SGS} = 0.05$ and $A_0 = 30$. The SGS filter width Δ is defined as

$$\Delta = \sqrt{\text{the maximum area among the faces of a cell}}; \tag{4.7}$$

in a Cartesian-structured grid system, it becomes

$$\Delta = \sqrt{\max(\Delta_x\Delta_y, \Delta_y\Delta_z, \Delta_z\Delta_x)}. \tag{4.8}$$

For the model function f_{SGS}, the wall-distance parameter y'_ε is modeled as $y'_\varepsilon = \left(\frac{u_\varepsilon y}{\nu}\right)\sqrt{C_l \frac{y}{\Delta}}$ with $C_l = 4$, where $u_\varepsilon = (\nu\varepsilon_{SGS})^{1/4}$ and $\varepsilon_{SGS} = C_\varepsilon \frac{k_{SGS}^{3/2}}{\Delta} + \frac{2\nu k_{SGS}}{y^2}$ with $C_\varepsilon = 0.835$. The SGS turbulent kinetic energy k_{SGS} is determined from the usual form of the transport equation using

$$\frac{Dk_{SGS}}{Dt} = \frac{\partial}{\partial x_j}\left\{\left(\nu + C_k f_{SGS}\sqrt{k_{SGS}}\Delta\right)\frac{\partial k_{SGS}}{\partial x_j}\right\} - \tau_{ij}\frac{\partial \bar{U}_i}{\partial x_j} - \varepsilon_{SGS}, \tag{4.9}$$

and setting $C_k = 0.1$.

4.3.3 Computational conditions and roughness model

A two-dimensional open channel-flow configuration is adopted as the test case, where one of the surfaces is a no-slip wall with/without roughness and the other is a free surface. Several studies of this kind adopted a channel-flow configuration with a rough wall on one side and a flat wall on the other side, which, however, greatly affects the balance of the wall-shear stress on each surface, resulting in a considerable shift in the center position of the mean velocity profile. Because this balance also depends on the performance of the SGS model as well as the grid resolution, the point of issue may become obscure. This is the main reason why the aforementioned open-channel configuration was adopted in this study, as the whole wall-shear stress must be covered by the rough wall and the position of maximum mean-velocity must reside on the free slip wall. Therefore, the point of issue in this study, the predictive performance of the SGS model for different grid resolutions, can be focused on properly.

In this simulation, the periodic boundary condition is specified in the streamwise and spanwise directions. To investigate the grid dependence arising from the present SGS model, three Cartesian-structured grids are made with the same computational domain of $4.8\delta \times \delta \times 2.4\delta$ in *x*-, *y*-, and *z*-directions, respectively (Table 4.1). The Reynolds number based on the mean friction velocity u_τ and the open-channel height δ is set to 395 for all test cases. First, two simulations without roughness (i.e., a smooth wall) are performed, where DNS is performed for a fine grid and LES is performed for a coarse grid. The data obtained are used to validate the present simulation by comparing with the corresponding DNS data from a previous study (Moser et al., 1999). Although the reference DNS was performed for a plane channel flow, it is expected that the fundamental features in the near-wall region are similar to those of the present simulation (see Chapter 5).

The direct forcing approach of the immersed boundary method is employed to build the roughness structure, in which a body force is added to the flow field where the roughness locates to express the influence of rough elements. The bottom surface is configured with a surface roughness defined using the 'egg-carton' shaped-surface function $\sigma(x, z)$ introduced in Bhaganagar et al. (2004),

$$\sigma(x, z) = \sigma_0 + \frac{h}{4}[-1 + (1 + \sin(\frac{2\pi x}{l_x} + \frac{2\pi z}{l_z})) \times (1 + \sin(\frac{2\pi x}{l_x} - \frac{2\pi z}{l_z}))], \tag{4.10}$$

where h is the roughness height from peak to valley, σ_0 is mean offset, l_x and l_z are streamwise and spanwise wavelengths of the roughness elements, the parameters are now set as $h = 0.0506329\delta$ ($h^+ = \frac{hu_\tau}{\nu} = 20$), $\sigma_0 = \frac{h}{4}$, $l_x = 0.6\delta$, and $l_z = 0.3\delta$. According to the results employing minimal-span channels by Chung et al. (2015), the scale of the present channel with the above roughness setting is believed sufficient to catch the flow properties such as mean velocity and turbulence stresses.

The second-order central difference scheme is used to discretize the spatial derivatives, except for the convection term of the transport equation of k_{SGS}, which is discretized by the second-order upwind scheme. Time marching is based on the fractional step method, in which the second-order Crank-Nicolson scheme is used for the velocity equations. Concerning the transport equation for k_{SGS}, the first-order Euler implicit scheme is used. The coupling of the velocity and pressure fields is based on the SMAC method. Note that our previous studies (Abe, 2013; Abe, 2014) had investigated the effects of a first-order time-marching scheme and a second-order upwind scheme adopted for the transport equation of k_{SGS}. These investigations confirmed that the time-integration and space-discretization schemes used here did not have any crucial effect on the computational results. More detailed descriptions of the computational schemes are given in Abe (2013). In addition to the simulations using the aforementioned anisotropic SGS model, the present study conducted simulations using an isotropic eddy-viscosity type of SGS model without an EAT, which is actually composed of only the first two terms on the right-hand side of Eq. (4.3). From a comparison of the results obtained by these two SGS models, we can analyze the effects of the EAT on the predictive performance for this kind of turbulent flow with wall roughness.

Table 4.1: Grids for the three test resolutions with roughness.

	Coarse	Medium	Fine
$N_x \times N_y \times N_z$	$65 \times 121 \times 65$	$129 \times 121 \times 129$	$257 \times 121 \times 257$
Δx^+	29.6	14.8	7.4
Δy^+	1.2 - 4.5	1.2 - 4.5	1.2 - 4.5
Δz^+	14.8	7.4	3.7

Figure 4.3 depicts a grid as well as a configuration of roughness for the bottom surface. The rough-wall elements composing the bottom surfaces as defined by Eq. (4.10) are presented for comparison in Fig. 4.4. Even with the same functional definition, the three roughness configurations are somewhat different. As the grid coarsens, an obvious deviation from the 'egg-carton' profile appears, as evident in the coarse grid. This feature arises from the immersed boundary method used for building the roughness configuration. Without operation of interpolation near the immersed boundary, the roughness is composed of small blocks of the same size as the corresponding local grid units in generating a discretized approximate structure of the 'egg-carton' surface. As a result, the coarse grid brings obvious large steps into the rough surface, which may look a little different from the original shape. The influence brought by this deviation is discussed in Chapter 5.

4.4 3D self-propelled fish swimming

4.4.1 Background

Fishes are sophisticated swimmers that exhibit interesting hydrodynamic features. It has been shown that separation elimination, turbulence reduction and energy extraction from oncoming flow are used in fish-like locomotion (Triantafyllou et al., 2002). Flow phenomena concerning swimming fish have attracted research interest from a variety of fields. This can yield a better understanding of the flow control mechanism for efficient propulsion and high stability, and inspire such engineering applications as bionic underwater machines (Liu & Hu, 2010).

The swimming mode adopted by a single type of fish is a compromise between propulsion efficiency and other aspects, such as predator avoidance (Sfakiotakis, Lane, & Davies, 1999). Moreover, factors like turbulence may also affect swimming behaviour (Liao, 2007), and the kinematics of the models of burst and efficient swimming can be different (Kern & Koumoutsakos, 2006). The swimming modes of fish can be generally divided into body/caudal fin (BCF) undulation and median/paired fin (MPF) movement (Sfakiotakis et al., 1999). Most fish use BCF for propulsion whereas MPF is frequently used for stability and manoeuvrability (Borazjani & Sotiropoulos, 2008). To study the swimming fish, in addition to theoretical analysis (Wu, 2001), numerical simulations of deforming surfaces, imitation by robotic fish or experiments employing real organisms are considered effective approaches to gaining a comprehensive understanding of aquatic locomotion (Lauder & Drucker, 2002). As a fundamental approach to studying the swimming phenomena, numerous experiments have been carried out, and have provided valuable data to help understand the kinematics and hydrodynamics of fish locomotion. Many such experiments employed real fishes (Webber, Boutilier, Kerr, & Smale, 2001), where dead fish were used in some cases (Beal, Hover, Triantafyllou, Liao, & Lauder, 2006; Long, Mchenry & Boetticher, 1994). In addition to using real fishes, artificial underwater mechanisms, such as oscillating foils (Anderson, Streitlien, Barrett, & Triantafyllou, 1998) and fish-like robots (Barrett, Triantafyllou, Yue, Grosenbaugh, & Wolfgang, 1999; Epps, Alvarado, Youcef-Toumi, & Techet, 2009), have also been frequently adopted.

With the development of computational technology, numerical simulations are becoming ever more important to investigate various flow phenomena. Many case studies have applied computational fluid dynamics (CFD) to these problems (Narasimha, Brennan, & Holtham, 2007; Stamou, Katsiris, Politis, & Schaelin, 2008; F. Wang, Reitz, Pera, Z. Wang,

& J. Wang, 2013). For studies of swimming fish, numerical studies have also provided insightful results, some of which may be difficult to obtain experimentally. For example, three-dimensional (3D) flow fields and vorticity structures can be directly visualised and studied by performing 3D simulations (Liu & Kawachi, 1999; Zhu, Wolfgang, Yue, & Triantafyllou, 2002), and parametric studies, such as those for Re and St, can be carried out by controlling the computational conditions (Borazjani & Sotiropoulos, 2008).

To address issues related to swimming fish, many numerical studies have used experimental data based on live fish as inputs to the kinematics used for simulations. While there seems not much result about virtual swimming, such as that which is generated by changing the combination of parameters related to swimming kinematics in real cases and may not be employed by fishes in the experimental observations. This kind of virtual swimming is challenging to implement in experiments using real fishes and may help better understand flow phenomena. One example is the study on the effects of body shape and kinematics on the hydrodynamics of BCF swimming by Borazjani and Sotiropoulos (2010). They conducted simulations of self-propelled virtual swimmers, such as those with anguilliform kinematics but carangiform body shapes. On the other hand, 3D simulations of self-propelled models have rarely been examined. Limitations on movement are normal in these kinds of studies; for example, movement was restricted in the streamwise direction in the study by Borazjani and Sotiropoulos (2010).

Therefore, the goal of this study is to conduct a 3D simulation of self-propelled swimming in which not only the model itself, but also the movement are in three dimensions. A parametric analysis of tail-beat frequency, phase difference and tail-beat amplitude is performed in the context of virtual swimming. In contrast to the work in (Borazjani & Sotiropoulos, 2010) where the governing equations are formulated in a non-inertial frame of reference to simulate self-propelled swimming, the governing equations are formulated here in an inertial coordinate system, and the shape and kinematics of a fish-like model, formulated in several body-fixed frames of references, are transformed into the inertial frame of reference. The simulation of a fish-like body in flow requires considering fluid-structure interaction (FSI), for which the immersed boundary method has been frequently adopted (Borazjani, Ge, & Sotiropoulos, 2008; Fauci & Peskin, 1988; Luo, Wang, & Fan, 2007). In this study, the direct forcing approach is used to generate a model of fish-like swimming motion. The movement of the fish body is defined in relation to other structures, rather than by designating one function to represent the entire body, as has usually been the norm in the literature. This modelling method is considered more representative of real fishes, and is more applicable to artificial machines, such as robotic fishes. The feedback mechanism is used to implement the beginning of the swimming process driven by a hydrodynamic force generated by the undulatory movement of the model. 3D swimming performance and the flow field in virtual modes are investigated. More informative and accurate results concerning the hydrodynamics of fish swimming are expected. The following section provides illustrations and descriptions of the proposed computational model and the relevant conditions.

4.4.2 Modelling and computational conditions

To satisfy the continuity condition of Eq. (2.5), the fish model is designed as a rigid-body system analogous to the robotic mechanism employed in the literature (Barrett et al. 1999). Figure 4.5 shows the schematic geometrical configuration and size of the simplified model, which consists of an ellipsoid head with semi-principal axes with lengths of $1.5l, 0.8l, 0.6l$ $(l = 0.15)$, and a body part composed of four blocks and a triangular tail, l is the length of the tail. The undulatory movement of the body is implemented by controlling the relative rotation around the vertical axes (3, 4, 5, 6) as shown in Fig. 4.5. Dorsal or pectoral fins are not considered for the sake of simplicity.

Figure 4.6 shows the body-fixed coordinate systems used to define the boundaries of the immersed body and the relative rotation of each part. The number of rotational axes corresponds to those in Fig. 4.5 (axis number 7 is attached to the end of the tail). The x-z planes of all body-fixed coordinate systems are confined to one plane. Thus, all y-axes are parallel. The distances between pairs of neighboring origins (for example $\overline{o_3 o_4}$) are fixed. This setting allows for a transverse undulation of the model to imitate fish-like swimming. As indicated in Fig. 4.6, the first coordinate system for the ellipsoid head coincides with the second one, and the last coordinate system, numbered 7, has unit vector along the same direction as the preceding one, numbered 6, but with a different origin.

The formulas for the inside-outside judgement for the ellipsoid, block and cylindrical surfaces used in constructing the model are listed in Table 4.2, which are derived in the body-fixed coordinate systems. In case of the ellipsoid, for example, the location of an inside point P is denoted by $\vec{r}(= \mathrm{x}\vec{\imath} + \mathrm{y}\vec{\jmath} + \mathrm{z}\vec{k})$ in the inertial coordinate system, which is equal to $\vec{r_1}(= r_{1x}\vec{\imath} + r_{1y}\vec{\jmath} + r_{1z}\vec{k}) + \overrightarrow{r_{o1}}(= x_1\overrightarrow{\imath_1} + y_1\overrightarrow{\jmath_1} + z_1\overrightarrow{k_1})$, where the subscripts x, y and z refer to the components of the inertial system. (x_1, y_1, z_1) is the coordinate in the body-fixed system calculated from the coordinates in $\vec{r}$ and $\vec{r_1}$ through the matrix in Table 4.2, which is then used to perform inside-outside judgement in the equation for the ellipsoid. For the block and cylindrical surfaces, a unit vector $\vec{n}$ is set for position .

The center of mass is assumed to always be at the center of the ellipsoid for the sake of simplicity; the movement of the model is then presented at this point. Given an initial location and velocity, the kinematics of the corresponding body-fixed system 1 (2) in Fig. 4.6 are computed using Newton's second law of motion for a single rigid body as shown by Eq. (3.36) and (3.37):

$$\vec{F} = m\vec{a}, \tag{4.11}$$

$$\vec{T} = [I_R]\vec{\alpha} + \vec{\omega} \times [I_R]\vec{\omega}, \tag{4.12}$$

where $\vec{F}$ and $\vec{T}$ are the external force and the torque acting on the body, respectively. In the above equations, m is the setting mass of the fish model, $\vec{a}$ is the acceleration of the center of mass, $\vec{\omega}$ is angular velocity and $\vec{\alpha}$ is angular acceleration. The inertia matrix $[I_R]$ is assumed to take the simple form $\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$, where I is a constant. Equation (9) can thus be rewritten as:

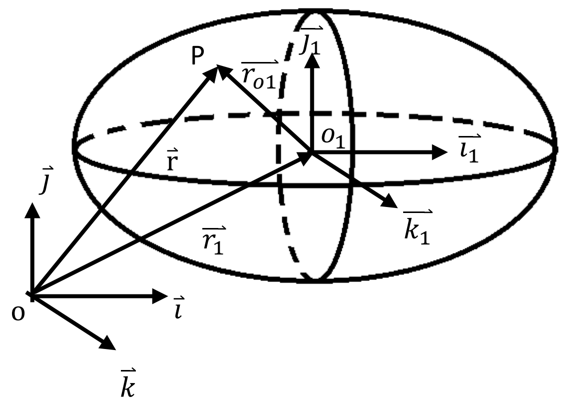
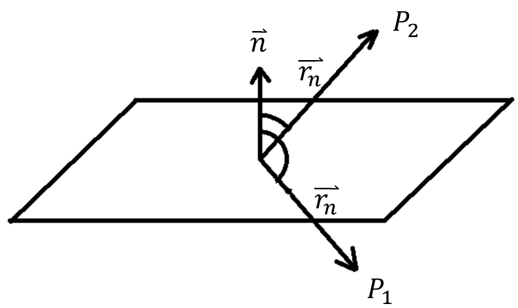
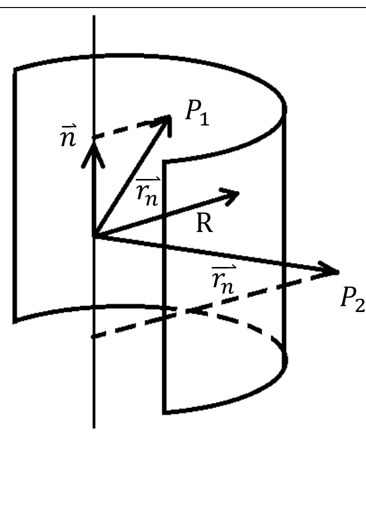
$$\vec{T} = I\vec{\alpha}. \tag{4.13}$$

For control volume ΔV_i with surface ΔS_i inside,

$$\vec{f_\imath}\Delta V_i + \overrightarrow{f_{s\imath}}\Delta S_i = \rho_f \Delta V_i \overrightarrow{a_{f\imath}}, \tag{4.14}$$

where ρ_f is the density of the fluid, and $\vec{f_\imath}$ and $\overrightarrow{f_{s\imath}}$ are the body force and the surface force imposed by the surrounding fluid on the control volume, respectively.

Table 4.2: Formula for inside-outside judgement

	Schematic diagram	Formula
Ellipsoid		$\begin{cases} \frac{x_1^2}{a^2}+\frac{y_1^2}{b^2}+\frac{z_1^2}{c^2} \le 1, & inside \\ \frac{x_1^2}{a^2}+\frac{y_1^2}{b^2}+\frac{z_1^2}{c^2} > 1, & outside \end{cases}$ where $\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} =$ $\begin{pmatrix} i_{1x} & j_{1x} & k_{1x} \\ i_{1y} & j_{1y} & k_{1y} \\ i_{1z} & j_{1z} & k_{1z} \end{pmatrix}^{-1} \begin{pmatrix} x - r_{1x} \\ y - r_{1y} \\ z - r_{1z} \end{pmatrix}$ The subscripts x, y, z denote the components in inertial system.
Block plane		$\begin{cases} \overrightarrow{r_n} \cdot \vec{n} \le 0, & inside\ (P_1) \\ \overrightarrow{r_n} \cdot \vec{n} > 0, & outside\ (P_2) \end{cases}$ where $\vec{n}$ is the wall normal unit vector, $\overrightarrow{r_n}$ is the vector from the origin of $\vec{n}$ to the point to be identified.
Cylindrical surface		$\begin{cases} \lvert\overrightarrow{r_n} \times \vec{n}\rvert \le R, & inside\ (P_1) \\ \lvert\overrightarrow{r_n} \times \vec{n}\rvert > R, & outside\ (P_2) \end{cases}$ or conversely. where $\vec{n}$ is the unit vector, with direction along the central axis of the cylinder, $\overrightarrow{r_n}$ is the vector from the origin of $\vec{n}$ to the point to be identified.

The integral form of Eq. (4.14) can be written as

$$\oiiint \vec{f}\, dV + \oiint \overrightarrow{f_s}\, dS = \oiiint \overrightarrow{a_f} \rho_f\, dV. \tag{4.15}$$

The second term on the left-hand side of Eq. (4.15) is the total surface force acting on the model. The term on the right-hand side is correlated with the kinematics of the model and reflects the effect of the inertia of the fluid inside the model. The acceleration of each control volume inside may be different, thus the calculation of this term requires a differential of velocity to obtain the acceleration for each node. This is thought to affect the stability of the simulation as an explicit scheme is used to update the model's velocity, especially when the acceleration is large as in the initial period. Therefore, the right-hand term in Eq. (4.15) is ignored in modelling for the stability of the simulation. Note that the effect of this term on accuracy is thought to become obvious under the condition of relatively high acceleration and low velocity. In the immersed boundary method, the body in flow is modelled by imposing a body force on the fluid inside the immersed boundary. The model can thus be considered part of the fluid in the flow field, and the acceleration and angular acceleration can then be calculated using Eq. (4.15) as:

$$m\vec{a} = \oiint \overrightarrow{f_s}\, dS = -\oiiint \vec{f}\, dV, \tag{4.16}$$

$$I\vec{\alpha} = -\oiiint (\vec{r} - \overrightarrow{r_c}) \times \vec{f}\, dV, \tag{4.17}$$

where $\overrightarrow{r_c}$ denotes the location vector of the center of mass that is $o_1(o_2)$ here. Note that although the right-hand term in Eq. (4.15) is ignored in the modelling, its effect persists, and is considered an additional mass $-\oiiint \overrightarrow{a_f}\rho_f\, dV/\vec{a}$ close to the fluid mass inside the fish model, which is approximately -0.0133. The real density of the fish model is considered close to that of the fluid. Therefore, to offset the influence of the additional mass on computational accuracy, especially at the starting stage, the setting mass is chosen to be approximately twice the fluid mass inside the model as $m = 0.03$. Since rotation is not the main concern in this study, the moment of inertia is set to be relatively large, mainly for the computational stability, as $I = 0.01$.

Once the movement of the first (second) body-fixed coordinate system is obtained by the above dynamic feedback, the computational process used to determine the orientations of the body-fixed systems 3 to 6 (denoted by the subscript "s" below), as shown in Fig. 4.6 at time step n + 1, is then set to implement the undulatory movement as:

$$\overrightarrow{J_s}^{n+1} = \overrightarrow{J_{s-1}}^{n+1}, \overrightarrow{\theta_s}^{n+1} = \overrightarrow{J_s}^{n+1}\theta_s(t), \tag{4.18}$$

$$\overrightarrow{\iota_s}^{*n+1} = \overrightarrow{\iota_{s-1}}^{n+1} + \overrightarrow{\theta_s}^{n} \times \overrightarrow{\iota_{s-1}}^{n+1}, \overrightarrow{\iota_s}^{n+1} = \overrightarrow{\iota_s}^{*n+1}/\left|\overrightarrow{\iota_s}^{*n+1}\right|, \tag{4.19}$$

where $\theta_s = \tan(A\cos(\omega q))$ and $q = \max(t - (s-3)\phi/\omega, 0)$. The parameter A is the relative amplitude of neighbouring parts. ω and ϕ are the relative angular velocity and the relative phase difference, respectively, and t is time. The initial posture is obtained by letting $\mathrm{t} = 0$ when $\theta_s = \tan(A)$. This setting provides a smooth starting process to prevent the abrupt alteration of velocity which may cause unphysical impulse. $\overrightarrow{\iota_1}(\overrightarrow{\iota_2})$, $\overrightarrow{J_1}(\overrightarrow{J_2})$, $\overrightarrow{k_1}(\overrightarrow{k_2})$ are initially set parallel to the corresponding inertial coordinate axis $\vec{\iota}$, $\vec{J}$, $\vec{k}$, respectively. From the top view of the model, Fig. 4.7 (a) gives a schematic posture to show the relationship between the parameters θ_s and the kinematics of the model. To illustrate the effect of ϕ on the undulatory mode, the schematic postures of the model using the above definition in a tail-beat cycle with $A = 1/4, \phi = 0, \pi/6, \pi/3, \pi/2$ are shown in Fig. 4.7 (b). The corresponding velocity and angular velocity are then updated as:

$$\overrightarrow{u_s}^{n+1} = \overrightarrow{u_{s-1}}^{n+1} + \overrightarrow{\omega_{s-1}}^{n+1} \times \overrightarrow{\iota_{s-1}}^{n+1} c_{s-2}, \tag{4.20}$$

$$\overrightarrow{\omega_s}^{n+1} = \overrightarrow{\omega_{s-1}}^{n+1} + \frac{d(arctan\theta_s)}{dt}\overrightarrow{J_s}^{n+1}. \tag{4.21}$$

In this study, a 3D channel is used through a structured grid with domain $3 \times 1 \times 1$ and grid numbers $360 \times 120 \times 120$ along the *x*- (longitudinal), *y*- (vertical) and *z*-axes (lateral), respectively, as shown in Fig. 4.8. The outlet boundary condition is specified along the longitudinal direction, and the non-slip wall condition is specified in the lateral and vertical directions. The Reynolds number based on the length of the fish model and the longitudinal velocity ranges approximately from 30 to 2,300. Note that to improve the efficiency and stability of the simulation, the wall condition is employed, which is not the typical situation for a normal swimming fish and may yield the wall effect on swimming performance. Webb (1993) investigated the swimming performance of steelhead trout in channels with solid or porous walls and highlighted the wall effect on kinematics. However, considering the relatively low Reynolds number in this study, the effect is considered insignificant. A simple assessment of effect of the distance between the pair walls along the lateral and vertical directions, is shown in the results and discussions.

The third-order upwind difference scheme is used to discretise the spatial derivatives. The time marching is based on the fractional step method, where the first-order Euler implicit scheme is used to derive the equations for the velocity. The coupling of the velocity and the pressure fields is based on the SMAC method. Due to the use of the third-order upwind spatial scheme and the first-order time marching method, it is necessary to confirm their effects on the computational results. This issue is properly assessed and a detailed discussion is provided the validation of the computations.

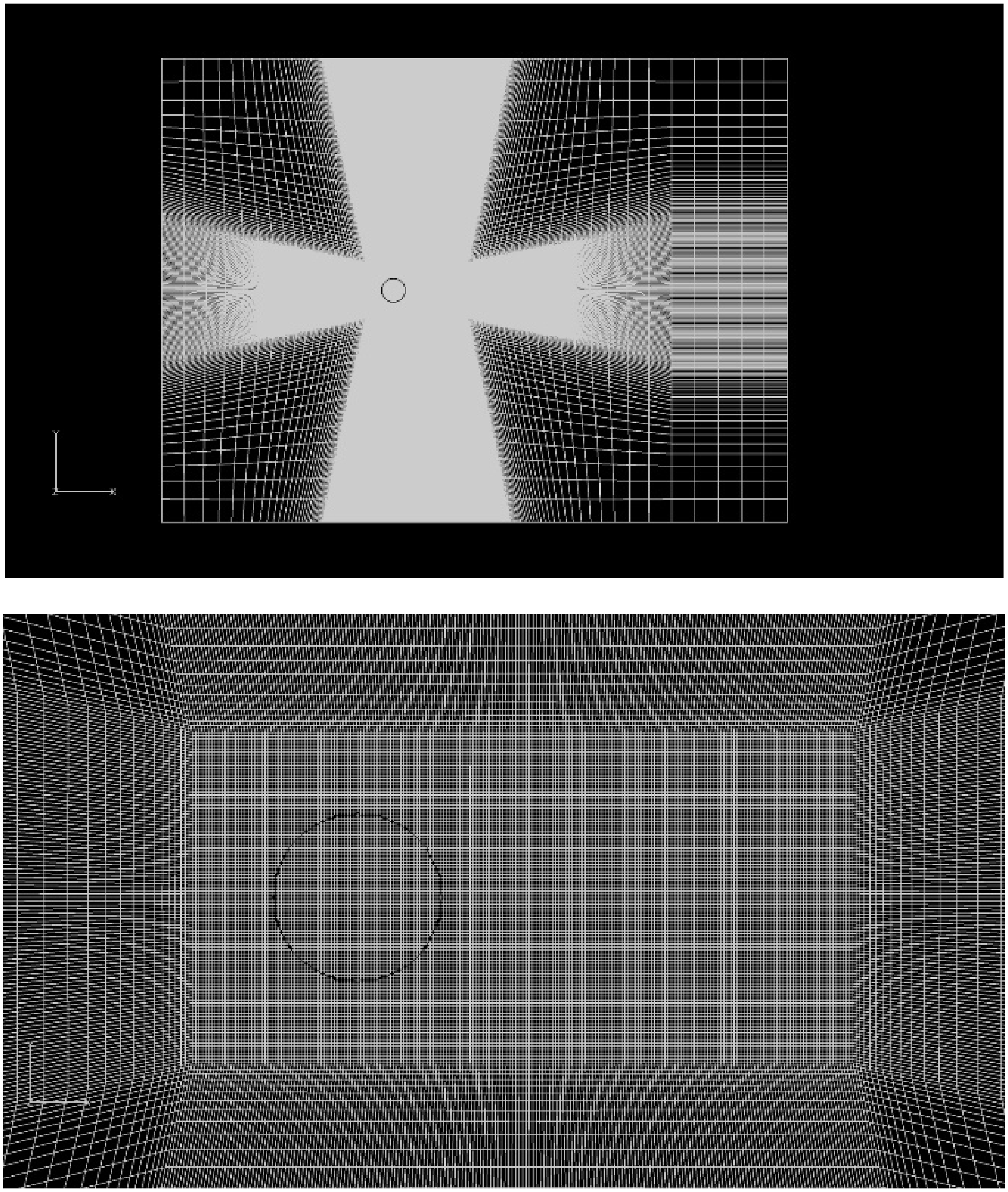


Figure 4.1: Grid plane perpendicular to the spanwise direction: entire view (upper) and partial magnification near the cylinder (lower).

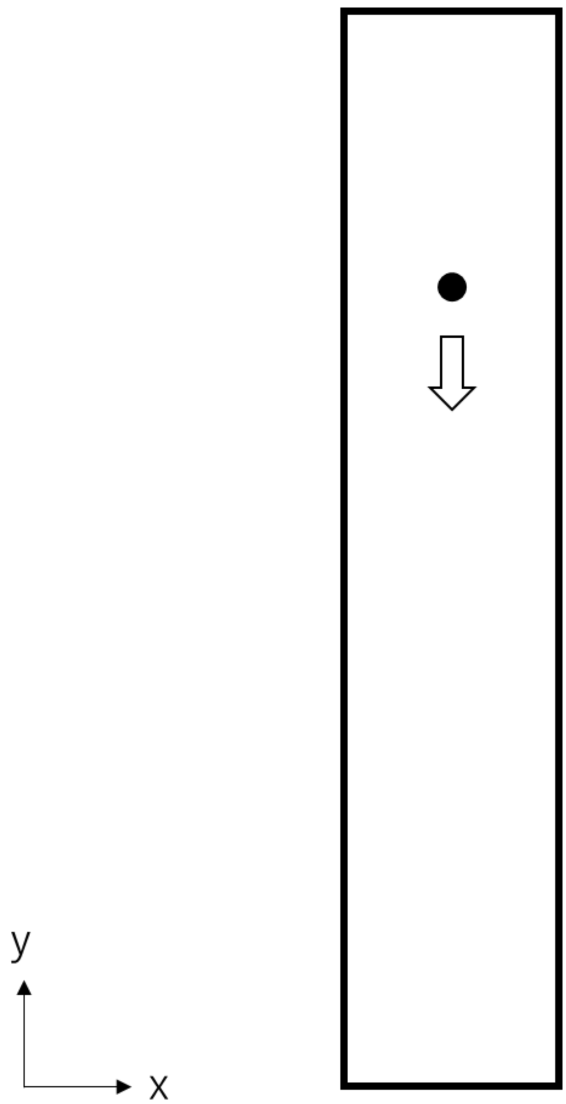
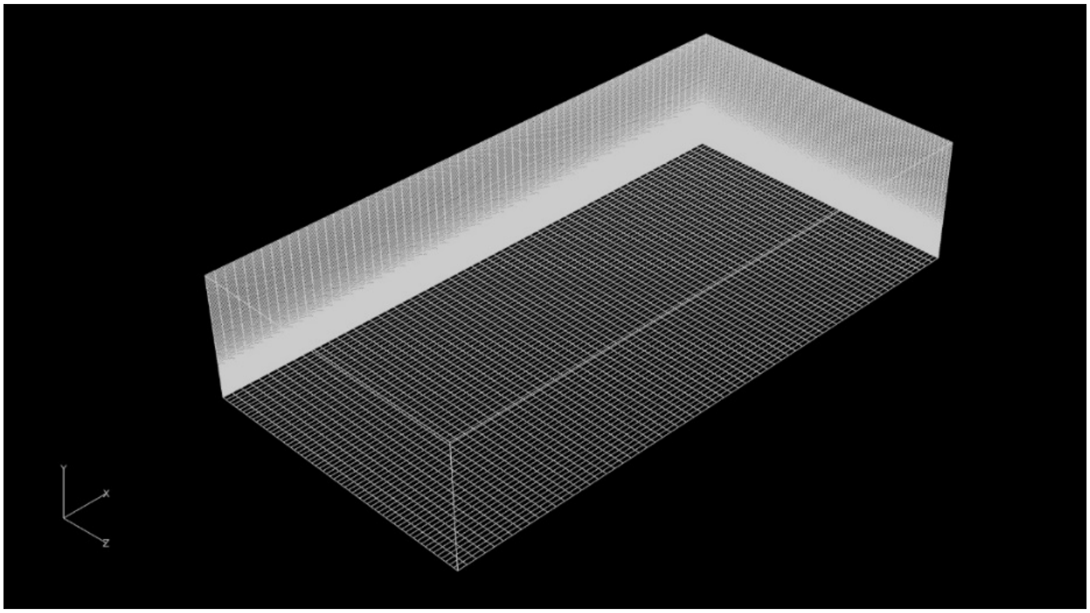
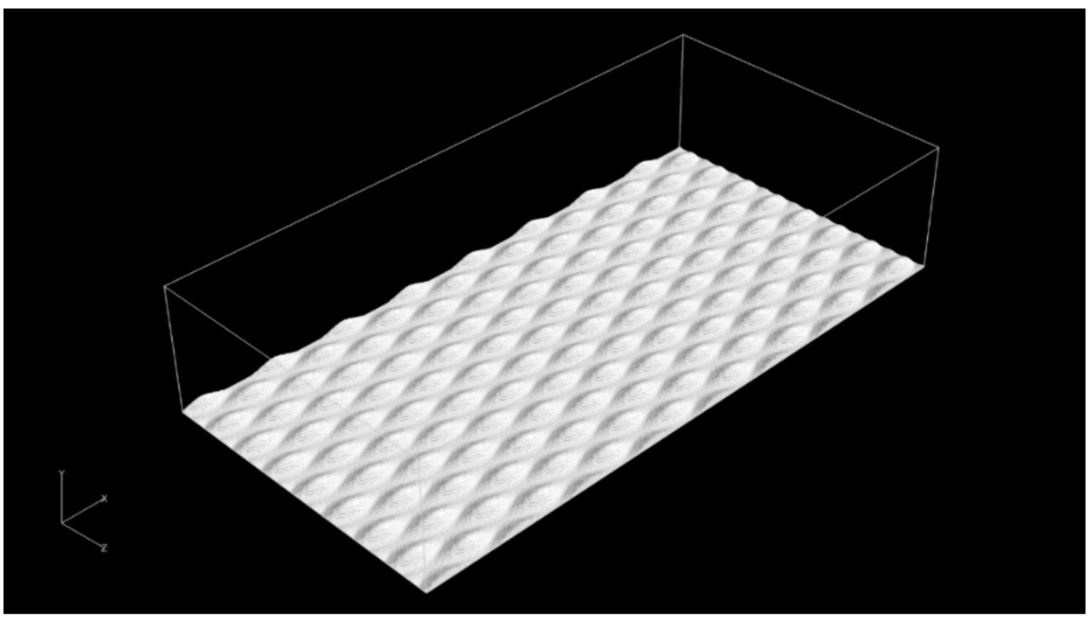


Figure 4.2: Schematic diagram of the single particle settling.



Grid



Roughness

Figure 4.3: Grid (coarse) and roughness (fine)

Fine

Medium

Coarse

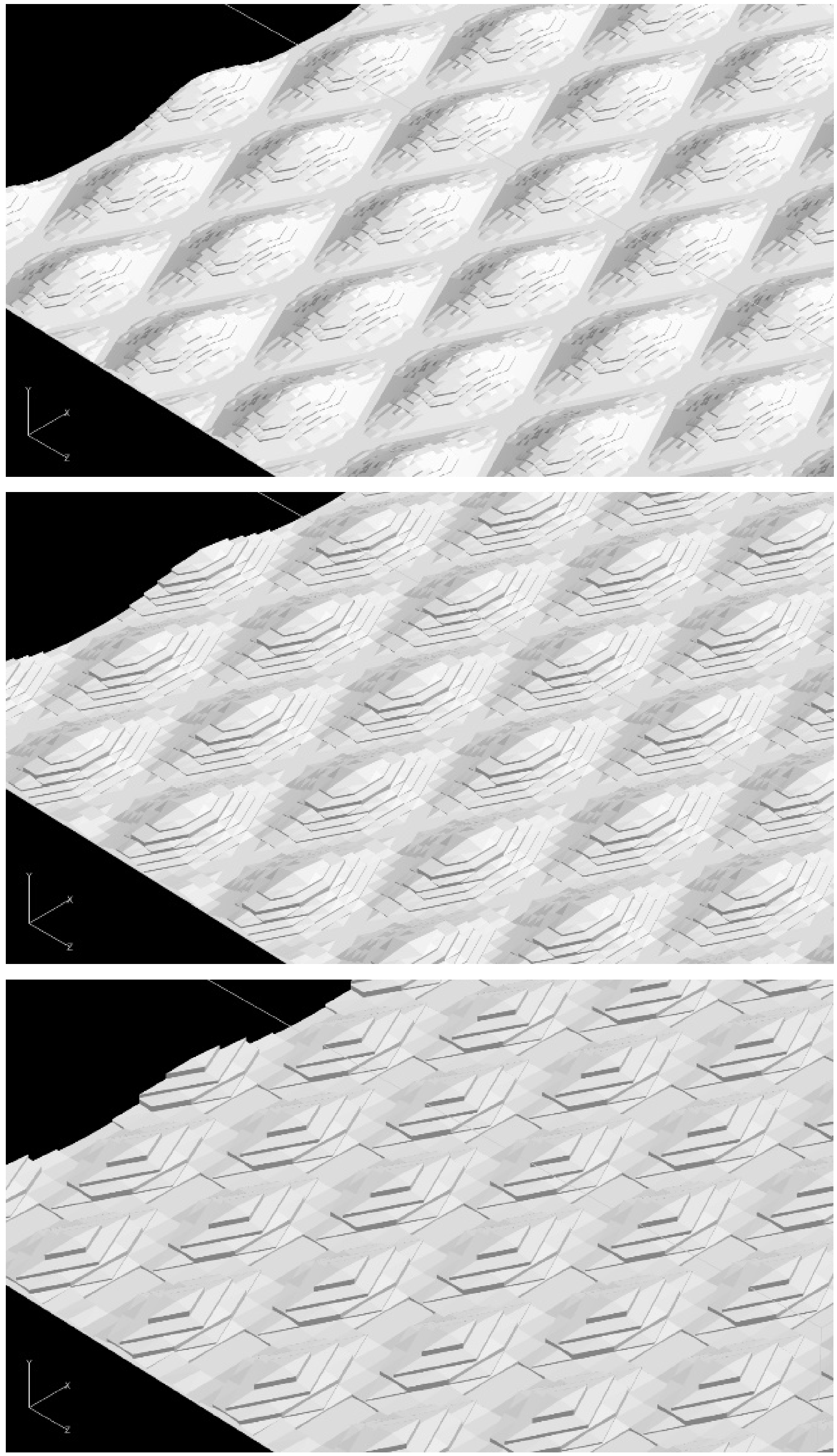


Figure 4.4: Close-up views of the three rough bottom surfaces

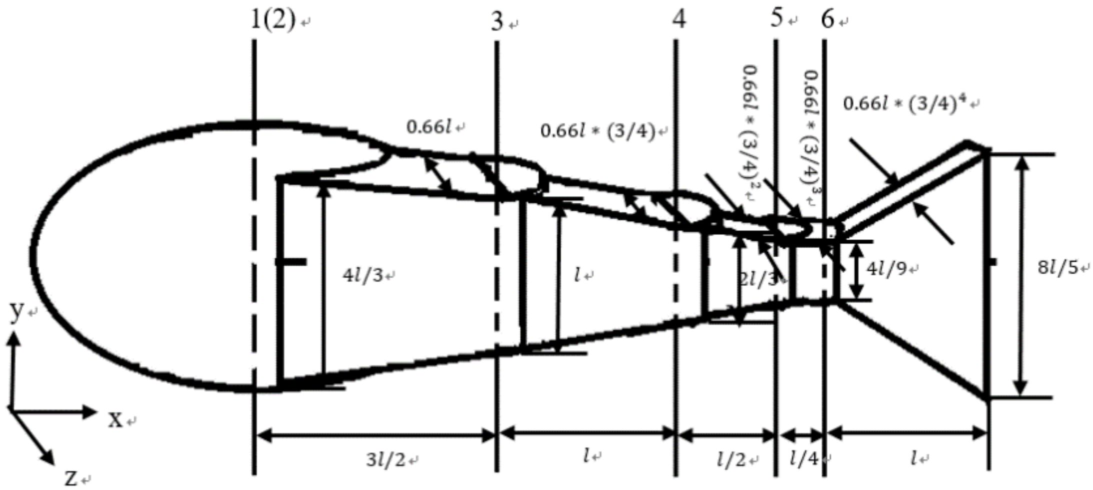


Figure 4.5: Computational model and dimensions ($l = 0.15$, the length of the tail).

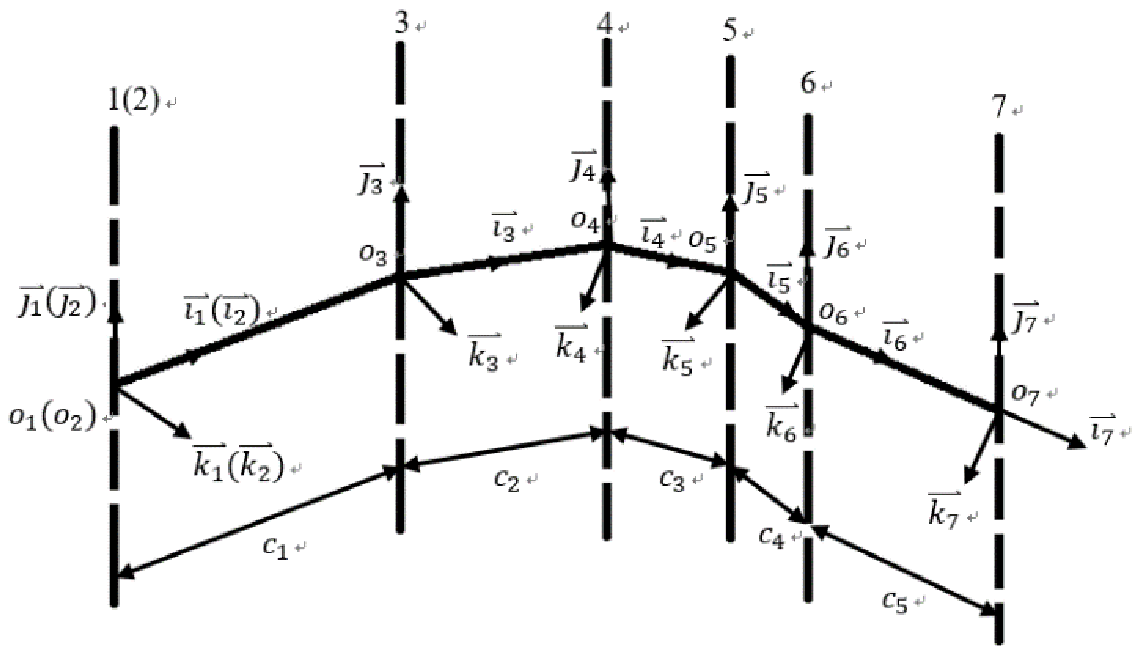
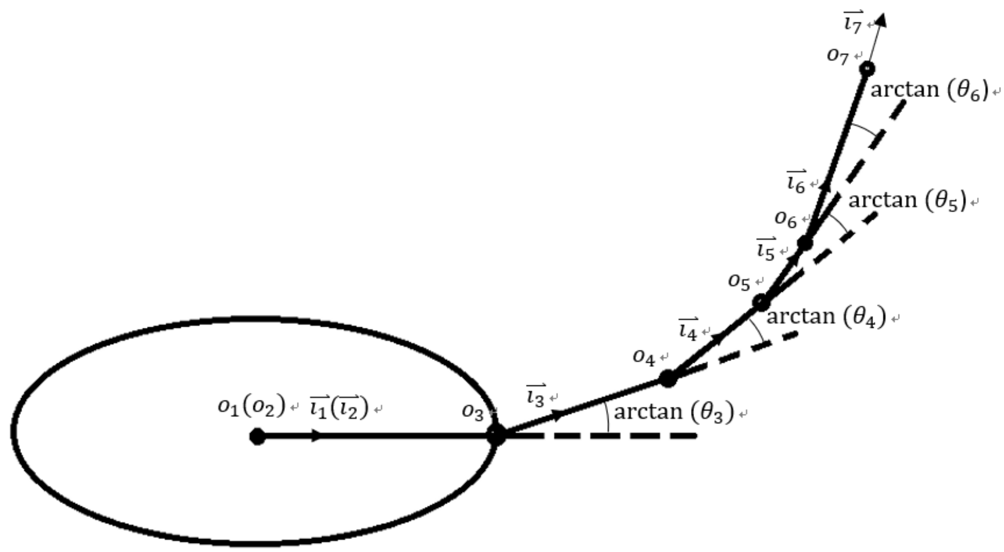
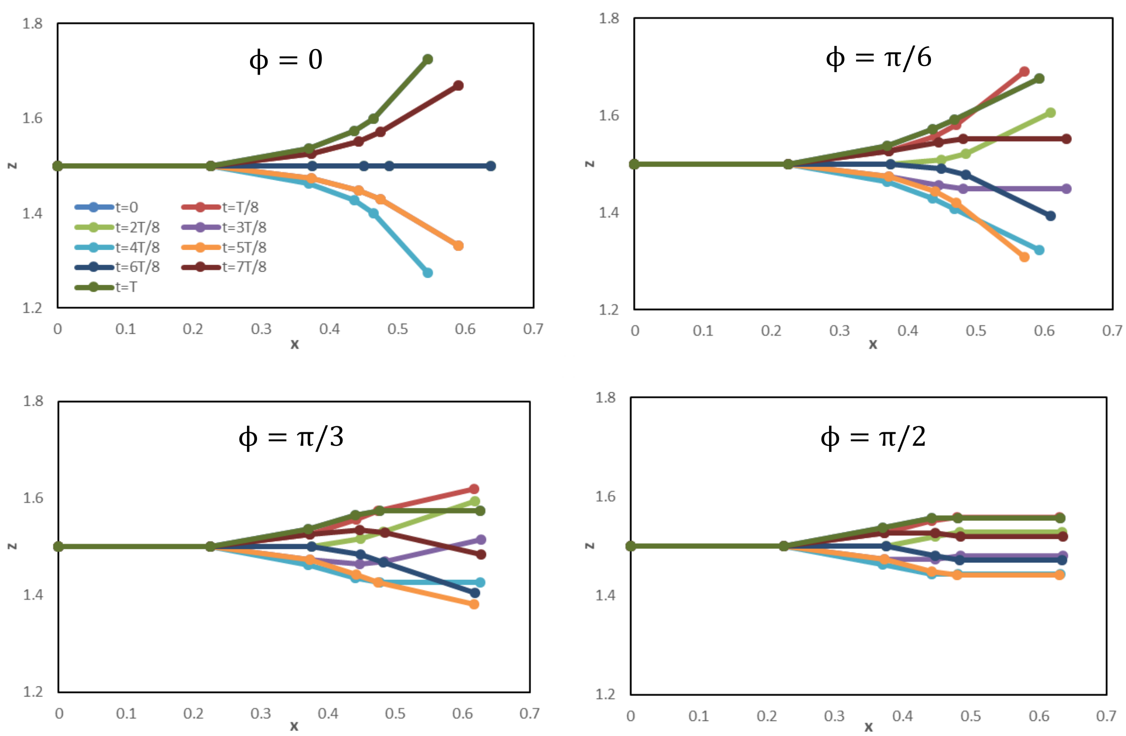


Figure 4.6: Body-fixed coordinate systems.



(a)



(b)

Figure 4.7: (a) Definition of the kinematics (top view); (b) Schematic postures in a tail-beat cycle for different phase differences ($A = 1/4$)

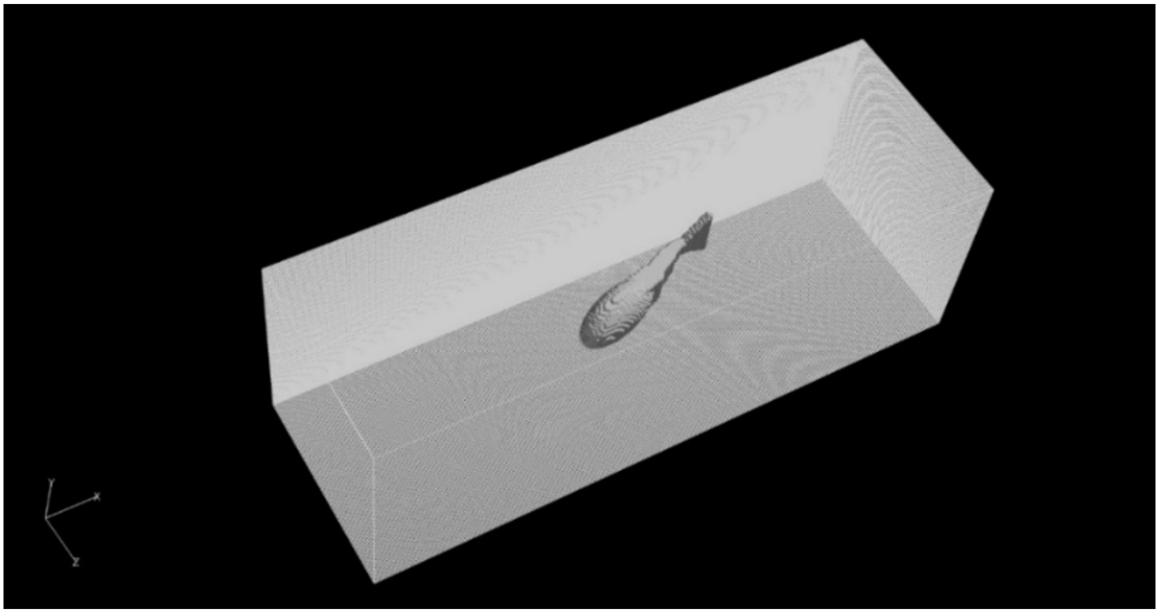


Figure 4.8: Grid for the simulation.

Chapter 5 Results and discussion

5.1 Validation cases

5.1.1 Flow around a circular cylinder

Table 5.1: Comparison of drag coefficient and Strouhal number with those in previous studies.

	Re = 100			Re = 1000		
	Cd	St	Cl (rms)	Cd	St	Cl (rms)
The present study	1.35	0.165	0.22	1.13	0.217	0.20
Kim et al. (2001)	1.33	0.165	---	---	---	---
Jordan & Fromm (1972)	1.28	0.16	---	1.24	0.206	---
Naito & Fukagata (2012)	1.33	0.165	0.23	1.09	0.21	0.21
Tritton (1959)	1.24-1.26	---	---	---	---	---
Braza et al. (1986)	1.30	0.16	---	1.20	0.21	---
Tseng & Ferziger (2003)	1.42	0.164	0.29	---	---	---

Figure 5.1 shows the lift coefficient ($\mathrm{Cl} = F_l/((1/2)\rho_f U_\infty^2 A_c)$) and the drag coefficient ($\mathrm{Cd} = F_d/((1/2)\rho_f U_\infty^2 A_c)$) against time (where F_l and F_d are the lift and drag forces, respectively, and $A_c = Dl_z$ is the reference area). The time-averaged drag coefficient, the Strouhal number ($\mathrm{St} = fD/U_\infty$, where f is the oscillation frequency) and the RMS-averaged lift coefficient are compared with those from previous studies in Table 5.1. The

results are found to agree well. Figure 5.2 shows the contours of the instantaneous streamwise velocity on the center planes for the two cases, and the vorticity contours are shown in Fig. 5.3. A Karman vortex street can be seen in the wake flow for $\mathrm{Re} = 100$, which indicates that the vorticity field was well captured. Figure 5.4. shows the three-dimensional vortex structures visualized by the second invariant of the velocity-gradient tensor for the case of $\mathrm{Re} = 1000$. We can successfully find a trend very similar to that in a previous DNS study (Naito & Fukagata, 2012).

5.1.2 Single settling sphere particle in fluid

Mordant and Pinton (2000) performed an experimental study on a settling particle for different Reynolds numbers and density ratios, and obtained a single exponential relation between vertical velocity and time to describe the movement as:

$$\frac{v_p}{V_1} = 1 - \exp(-\frac{3t}{\tau_{95}}), \tag{5.1}$$

where v_p is the particle velocity, V_1 is the limit velocity, t is time and τ_{95} is the characteristic time taken for the particle reaching 95% of the limit velocity.

Figure 5.5 shows the vertical velocity of the particle against time, which agrees well with the profile predicted by Eq. (5.1). The comparison of limit velocities is provided in Table 5.2. It is evident that a satisfactory agreement is obtained with previous studies for the same conditions.

Table 5.2: Comparison of the limit velocity with previous studies

	D (m)	ρ_p (kg/m^3)	V_1 (m/s)	Re
Mordant and Pinton (2000)			0.0741	41
Luo et al. (2007)	0.0005	2560	0.0781	44
The present study			0.0745	41

Figure 5.6 shows the velocity contours on the vertical central plane from *t*=0.05s to *t*=0.3s. As can be seen from Fig. 5.5, the descending velocity has been close to the limit velocity

by t=0.05s, and convergence at the limit velocity till t=0.1s. The descending process includes a short acceleration and mainly proceeds at a uniform velocity. From the velocity contours, although the descending velocity becomes stable in a relatively short time from the starting of the sediment. The development of the flow field continues with an increasing region of wake flow until the particle reaches the bottom of the channel. In the wake flow, it is noticed that a region of relatively high descending velocity can be seen in dark blue, which develops quickly as the particle falls and reaches a stable scale shortly from the start of sediment. Outside this region, a growing green area can be seen in which the velocity is relatively lower and the convergence of its scale is achieved later than the blue region. An axisymmetric wake flow field is maintained throughout the sediment and no significant turbulence is observed. Vortex structures investigated in different values (10 and -20) of the second invariant of the velocity-gradient tensor can be seen in Fig. 5.7. The results show that both vortices become stable shortly from the start and it takes more time for the outer vortex (-20) to reach a certain shape. Corresponding to velocity distributions, the vortex around the settling particle is axisymmetric and no apparent turbulent structure is observed.

In the present settling particle simulation, the time step Δt was firstly set to 10^{-4}. To assess the dependency of the time-step scale, two cases with larger time steps ($\Delta t = 2 \times 10^{-4}, 3 \times 10^{-4}$) were considered and the obtained results are compared in Fig. 5.8. As it shows, the profiles of settling velocities in the three cases involving different time steps coincide with one another, indicating that the dependency of the time-step scale was not a serious problem in this simulation as far as moving boundaries during flow were concerned. In addition to this validation case, the effect of the time-step scale was also assessed for the problem of swimming fish (A=0.25, $\omega = \pi$, $\phi = \pi/6$) through two test cases with time steps of 1×10^{-3} (Baseline) and 2×10^{-3} (Baseline_1). Grid dependency was also investigated for the case of swimming fish by performing a simulation under the same condition as the baseline case using a finer grid consisting of $N_x \times N_y \times N_z = 540 \times 180 \times 180$ (Fine). The longitudinal displacement and velocity are shown in Fig. 5.9. By comparing the Baseline and Baseline_1 cases, as in the case of validation, the effect of the time-step scale was not obvious. Concerning the grid dependency, although there was some discrepancy between the Baseline and the Fine cases, the overall feature of the swimming process was not considerably affected. Thus, the discrepancy is regarded insignificant and the grid dependency was not considered serious in this study.

5.2 Turbulent open channel flow with wall roughness

The basic performance of the present SGS model for a smooth wall has been validated in a previous study by Abe (2013). However, the present study adopts a different flow configuration, where one of the surfaces is treated as shear free. Therefore, to confirm the validity of the present simulation in the near-wall region on the bottom surface, Fig. 5.10 compares the mean streamwise velocity obtained by the DNS for a smooth wall using the fine grid (Table 4.1) with the results for a conventional channel-flow DNS reported by Moser et al. (1999). Note that to confirm the basic performance of the present SGS model for coarse grid resolution, the results of the LES with coarse grid (Table 4.1) are also included. It is found from Fig. 5.10 that the present DNS data correspond well to those by Moser et al. (1999). Moreover, the LES even with coarse grid yields satisfactory predictions for the mean streamwise velocity, although a slight difference is still observed. The components of the Reynolds normal stresses as well as the Reynolds shear stress are compared in Fig. 5.11. The results for both show generally good agreement in the region close to the no-slip wall on the bottom. Although there are considerable differences between the two DNS data in the upper region close to the free surface (center line in the reference DNS), no crucial effect is evident in the predictive performance on the bottom side, which is the main concern in this study. It is also notable that even the predicted Reynolds stresses near the free surface correspond well to what is generally known for this kind of flow. Concerning the LES with coarse grid, the belief is that generally reasonable results are obtained, although slightly higher streamwise and lower spanwise predictions are still seen in the near-wall region compared with the DNS data. Concerning the contribution of each term in Eq. (4.3), according to the analysis in Abe (2014) on the results of the channel flow case, the time-averaged SGS stresses calculated with the linear EVM term ($-2\upsilon_{SGS}S_{ij}$) mainly contributes to the SGS shear stress while its effect is very small for normal stresses. In contrast, the EAT ($2k_{SGS}b_{ij}^{SGS}$) plays a dominant role for the normal-stress predictions.

The mean-velocity distributions for the three test cases with wall roughness are compared in Fig. 5.12, as well as the aforementioned open-channel DNS data for smooth and rough walls for reference. Note that the LES data without the EAT (i.e., isotropic SGS model) using the same grids are also involved for further comparison. The DNS for rough wall was performed using the grid of $N_x \times N_y \times N_z = 385 \times 181 \times 385$ ($\Delta x^+ = 4.9$, $\Delta y^+ = 0.8 - 4.0$, $\Delta z^+ = 2.5$) with the same domain as well as the boundary conditions for the present LES cases. Concerning the results of the DNS, the velocity profile with wall roughness shifts downward compared with that obtained in the smooth-wall DNS, but maintaining a similar slope, which agrees with the general knowledge. For the LES results, the velocity profiles of the fine and medium cases with and without EAT generally agree well with the DNS data, while less downward shift is shown for the coarse cases. Note that this may arise as an effect of the quality of reproduction of the roughness configuration besides the grid dependency of the SGS models. By comparing the results between LESs with EAT (i.e., anisotropic SGS model) and without EAT (i.e., isotropic SGS model) in the figure, for the fine and medium grid cases, the two models yield close profiles, while a definite difference is seen in the coarse grid case. Even considering the effect of the distortion of the roughness configuration as is described later, an obviously better performance is obtained by adopting the SGS model with EAT than that without EAT, especially in the region exhibiting the logarithmic law, although both of them show some change in shape of the logarithmic region. The isotropic SGS model without EAT shows a much larger discrepancy from the DNS data compared with the anisotropic SGS model with EAT. This fact indicates that the EAT contributes to reducing the grid dependency of the predictions, even for near-wall turbulence with wall roughness. The modification ΔU^+ from roughness effects is often called the roughness function. In the present study, it is evaluated to be about 3.0 as can be seen in Fig. 5.12.

To investigate in more detail the effect of the roughness configuration reproduced by the immersed boundary method, the instantaneous velocity-vector plots obtained from the results with EAT as well as the roughness configurations generated for each case are shown in Fig. 5.13. As mentioned in Chapter 4, the roughness is composed of small blocks of the same size with corresponding local grid units. Therefore, in the finer grid case, a better shape of 'egg carton' surface can be realized. The distortion is more obvious with the coarse grid; not only is the rough surface modified with bigger stages, but also the height of the roughness peak decreases more significantly than with the fine or medium grid. This may be one reason for the lower roughness function ΔU^+ seen in Fig. 5.12. However, even considering this effect of shape distortion, the downshift of the profile of the coarse case without EAT is still insufficient, which is more likely caused by a higher grid dependency of the isotropic SGS model. The results also indicate that there is a low velocity region on the rear side of the rough mountain, while no obvious recirculation is observed in the present three test cases.

From the distributions of turbulence energy shown in Fig. 5.14, the results of the DNS with roughness exhibit a lower peak compared with the DNS without roughness, and the location of the peak value is shifted away from the wall. The results of fine cases with and without EAT agree well with the DNS data. The profile of the medium case without EAT shows a slightly higher peak, while lower value can be seen in the outer layer that is similar to that with EAT. Obviously higher peak values are obtained in the coarse grid for both LES cases in the near-wall region, even higher than that in smooth DNS. Among them, however, the prediction by the model with EAT is clearly better than that without EAT. In the outer layer, lower energy is obtained for both coarse cases, and even lower value is shown for the case with EAT.

The Reynolds normal stresses as well as the Reynolds shear stress for the rough-wall LESs are compared with each other, as well as with those of the smooth-wall and rough-wall DNSs in Fig. 5.15. As is found by comparing the rough-wall and smooth-wall DNS results, the roughness causes obviously lower peak for the streamwise component as well as a shift away from the wall for all the components. The results of fine cases with and without EAT agree well with those of the DNS data. For the cases with EAT using the coarse grid, a higher peak value can be seen in the streamwise stress, whereas the peak values decrease in the wall-normal and spanwise directions, as found by comparing with the DNS results. This may be caused by a general feature of LES that coarse grid resolutions produce a stronger anisotropy. Moreover, the location of the peak value slightly changes in the wall-normal direction, which is likely to move farther away from the wall than with the fine and medium grids. Comparing with the rough-wall DNS results, the departure mode with the coarse grid is similar to that for the smooth-wall cases when comparing the LES and DNS profiles (Fig. 5.11), which implies that the exist of the roughness may not obviously influence the performance of the present SGS model on turbulence stresses, although the peak value decreases more apparently in the rough wall case. By comparing the cases with and without EAT, the results for the fine and medium grids generally agree well with each other, although slightly lower peak values are shown for the medium case with EAT than that without EAT in the normal stress components. The difference in peak values is more obvious for the coarse cases, which indicates that the exist of EAT tends to cause lower normal stresses when the grid gets coarser in the present study. This may lead to giving a higher peak value of the turbulence energy in the results without EAT.

To investigate the effect of the SGS stresses on the total values, Fig. 5.16 compares the SGS components with and without EAT for the coarse grid case. It is found that the model with EAT successfully provides the SGS-stress anisotropy, while the model without EAT returns isotopic distributions. This is a notable feature of the present anisotropic SGS model. An important issue can be seen in the prediction in the wall-normal direction, where the model with EAT shows the correct near-wall limiting behavior, while the model without EAT shows wrong one. This wrong behavior gives a considerable overprediction of the wall-normal component, leading to a decrease of the prediction accuracy for the total value of the wall-normal Reynolds stress. Concerning the SGS shear stress, similar profiles are obtained for both models, although there can be seen a slight difference in shape. Such being the case, it is confirmed that the EAT contributes to improving the distributions of the SGS stresses, resulting in better predictions of the total stress components.

Figure 5.17 compares the GS and SGS components of the Reynolds shear stress for the cases with and without EAT. For the fine cases, no obvious SGS component can be seen either with or without EAT compared with the GS component. As the grid becomes coarser, the SGS component increases, especially in the near wall region. In the coarse case with EAT, the influence of the SGS stress spreads farer than that without EAT into the main flow. A decrease of the GS component can be found in the corresponding region, this indicating a transfer of stress from the GS component to SGS component. The total value of the case with EAT is closer to the DNS result than that without EAT, more loss of the shear stress appears for the case without EAT during the transfer from GS component to SGS component.

Figure 5.18 compares the GS and SGS components of the turbulence energy for the cases with and without EAT. For the fine cases, the SGS components of both cases with and without EAT are insignificant. As the grid becomes coarser, the SGS components in the cases both with and without EAT increase. For the coarse cases, the GS component of the case with EAT decreases in the outer layer while remains in the inner layer compared with the fine case. In the case without EAT, the GS component also decreases in the outer layer, while the peak value increases, which combining with a larger SGS component, gives an more obvious overestimated peak than in the case with EAT compared with the DNS profile.

Figure 5.19 compares the results of the instantaneous vortex structures and streamwise velocity contours of rough-wall cases with EAT as well as those for the smooth-wall and rough-wall DNSs. From the figure, the vortex structures for the coarse grid show a slightly different aspect from the fine and medium grids under a certain contour parameter for plotting, whereas the latter two are similar to each other, as well as the rough-wall DNS results. For the fine and medium grids, more vortex structures are observed in the channel compared with for the coarse grid, suggesting the existence of a grid dependency on the size, density, and intensity of the generated vortices. However, one important feature is that even with the coarse grid, vortex structures are clearly seen in the region close to the wall surface (marked in blue). In contrast, having looked at the velocity contours, in the inner layer of LES cases where the roughness exists, the flow seems to change compared with that with a smooth wall. A thicker boundary layer is generated by the effect of the roughness. However, from the top of the rough elements to the center region, the development of turbulence in the streamwise direction seems not significantly influenced by the roughness. Thus, the effect of roughness on the flow field is similar to bringing a rise of the wall boundary by a height related to the size of the rough element that is also seen in the mean streamwise velocity profile in Fig. 5.12.

From the above discussion, it can be seen that the difference between the results with and without EAT mainly appears in the coarse grid case. This may be caused by the relatively fine grid resolutions for the present LES cases. To better understand the effect of the EAT, cases with even coarser grid resolutions can be tested and the results being compared with those without EAT in a future study. In that case, distortion of roughness configuration by immersed boundary method is thought to be more serious, thus the body fitted grid may be considered more appropriate.

To illustrate the reduction of computational cost quantitively, the cost defined by ($\text{Number of grid points} \times \text{time step size (which is defined here as } 10^{-4}/\Delta t))$ is calculated for the present rough-wall cases and listed in Table 5.3 with the proportion to the cost of the DNS case. Note that the influence of grid dependency on the convergence time is not considered here. It has to be pointed out that the reduction of computational cost is an important problem that is closely related to the accuracy requirement. Although the cost of the coarse grid is quite lower than that of the fine and medium grids, its prediction accuracy generally decreases. However, considering the fact that the case with EAT yields a better mean velocity profile than that without EAT even for the coarse grid, the computational cost of the present SGS model can be less than that without EAT under a certain accuracy requirement.

Table 5.3: Computational cost for the rough-wall cases.

	DNS (rough)	Fine	Medium	Coarse
Cost	1.341×10^7	2.664×10^6	4.027×10^5	1.022×10^5
Cost/Cost_DNS	1.0	0.20	0.030	0.0076

5.3 3D self-propelled fish swimming

Concerning the three-dimensional fish-swimming case, the effect of the tail-beat frequency on the swimming performance was first investigated. Figure 5.20 shows the displacement (left) and velocity components (right) of the center of mass of the fish-like model against time, for $\omega = \frac{\pi}{4}, \frac{\pi}{2}, \pi\ and 2\pi$ with $\phi = \pi/6$ and A=0.25. The model started at the location (x=1.8, y=0.5, z=0.5), and the time it took to reach the end of the channel decreased almost linearly with increasing ω. The indicated linear relationship between the swimming velocity and the tail-beat frequency has also been reported in previous studies (Long et al., 1994; Epps et al., 2009) and can be found for the lateral component. The forward thrust was the counter-force of pushing the fluid backward. With larger ω, more backward fluid was generated at a higher velocity, resulting in a higher forward thrust and, thus, higher swimming velocity. On the contrary, the drag increased with swimming velocity and a limit velocity was achieved when thrust was balanced by the drag. This linear relationship between ω and swimming velocity and, thus, longitudinal displacement indicated a linear similarity between the functions---one between the thrust and ω, and the other between drag and swimming velocity---for the Reynolds numbers and swimming modes used. Compared with the longitudinal movement along the x-axis, a more obvious oscillation was observed in the lateral direction. Positive displacements were observed in all cases; they might have been obtained due to the initial setting of the body posture, such that in the first tail-beat cycle, the counter-clockwise torque from the top view generated by the undulation of the body turned the head of the model slightly to the left and the deflection persisted in the following swimming process. Each lateral velocity component had one maximum and one minimum in a tail-beat cycle, whereas there were two crests and two troughs for each component of longitudinal velocity. The different values of the two troughs in one cycle were thought to have occurred due to the deflection of the longitudinal central axis of the body from the x-axis. The vertical velocity was low compared with the other two velocity components, and a larger fluctuation tended to appear in the case of higher angular velocity.

Figure 5.21 shows the velocity vectors and color contours for the cases with different angular velocities. Although the wake wave becomes intensified as the angular velocity increases, the structure of the wake flow does not change significantly. This could also be seen from the vortex structures in Fig. 5.22. As the angular velocity increases, more vortices generate in a compact series of vortex rings, and the vortex surrounding the body part begins to grow.

Figure 5.23 shows the velocity vectors and longitudinal velocity contours on the plane of $y = 0.5$ during one tail-beat cycle from $t = 10$ to $t = 12$ for the case where $\omega = \pi$ with $\phi = \pi/6$ and A=0.25. The corresponding vortex structures are shown in Fig. 5.24. Two peaks of longitudinal velocity in the cycle can be seen in Fig. 5.20 and correspond approximately to the times at $3T/8$ and $6T/8$, when the tail was close to crossing the body's vertical central plane. A backward wake flow with oscillation is seen in Fig. 5.23. Note that a region of high backward velocity was generated on the leeward side of the tail and shed into the wake flow when the swinging movement changed the direction. As the high velocity was not induced on the windward side, this implies that rather than by ‘pushing’ the fluid, the thrust was generated more by ‘pulling’ it. The mechanism of thrust generation can also be investigated from the perspective of vortex structures in Fig. 5.24. A series of vortex rings shed from the trailing edge of the tail to form a reverse Karman vortex street in the wake flow, which moved backward along the walls after having reached them. The body component of the model was surrounded by weak vortex structures while no obvious shedding of body vortex was observed.

Figure 5.25 shows the pressure distributions on the surface of the fish model in one tail-beat cycle. Because of the symmetry of the body shape and movement in the lateral direction (x-y plane), only one side of the model is displayed. The pressure difference is mainly generated on the tail, where high pressure is induced on the windward side and low pressure is on the leeward side. In the first half cycle from 0 to 3T/8, the displayed tail plane plays the role as the windward surface. A concentrated high pressure is firstly generated near the region connecting the body and the tail, and then spreads into most part of the windward surface with a lower peak value and more even distribution as the tail swings to the other side. In the following the half cycle from 4T/8 to 7T/8, the displayed tail plane plays as the leeward surface. A low pressure is generated on the end of the tail and spreads to the whole leeward surface in this process.

In addition to the tail-beat frequency, another important factor that affects the swimming performance significantly is the undulatory mode, in which anguilliform and carangiform are common types (Sfakiotakis et al., 1999; Gillis, 1996). For different undulatory modes, the propulsive wavelength can be different. The smallest propulsive wavelength for a species decreases with increasing number of vertebrae (Long & Nipper, 1996). In this study, as the number and size of the segments of the model were fixed, the relative phase difference was altered to change the undulatory mode, and its effect was investigated. Figure 5.26 compares the displacement (left) and velocity components (right) against time along the three directions for $\phi = 0, \frac{\pi}{12}, \frac{\pi}{6}, \frac{\pi}{4}, \frac{\pi}{3}, \frac{\pi}{2}, \frac{3\pi}{4}$, and π, with $\omega = \pi$ and A=0.25. As the phase difference increased, the longitudinal swimming distance to the initial position decreased, except in cases where $\phi = \pi/12$ and $\phi = 3\pi/4$, for which the swimming distance was slightly larger than that for $\phi = 0$ and $\phi = \pi/2$, respectively. When the phase difference was greater than $\pi/2$, there was no apparent displacement along any direction. By comparing the peak-to-peak height, stronger oscillations for the longitudinal and lateral velocities were obtained when the phase differences were smaller, which indicated that more intense acceleration and deceleration process occurred in a mode where the fish body behaved "stiffer", and this corresponded to higher swimming velocity.

Figure 5.27 shows the velocity vector and color contours for structures for $\phi = 0, \frac{\pi}{4}, \frac{\pi}{3}$, and π, with $\omega = \pi$ and A=0.25, while Fig. 5.28 shows the corresponding vortex structures. As the phase difference increased, the backward jet flow weakened and fewer vortex structures were observed, Particularly in the case of $\phi = \pi$, the tail plane was always parallel to the vertical central plane of the body and swung in a narrow region. It is noteworthy that the proposed fish model has only four rotation joints, which may not be adequate to express the features of certain swimming patterns, especially those with larger phase differences. It is thought that models with more and smaller segments are needed to reproduce large phase differences or smaller propulsive wavelengths more properly.

To illustrate the effect of body amplitude, Fig. 5.29 shows the displacement (left) and corresponding velocity components (right) against time in three directions for parameter A changing from 0.2 to 0.5 while keeping $\omega = \pi$ and $\phi = \pi/6$. As can be seen, the largest longitudinal displacement is obtained when A is between 0.3 and 0.35, and the oscillation of longitudinal velocity increases as the amplitude getting larger. The swimming velocity seems to be sensitive to the body amplitude in a narrow range. The lateral displacement and velocity changes little and a slight peak shoulder appears in the lateral velocity profile when A is larger than 0.3. Note that as is different from the discussion on the tail-beat frequency or phase difference, stronger oscillation may occur even at a lower longitudinal velocity when changing the amplitude. When the amplitude is higher than the optimal value where the largest mean longitudinal velocity is obtained, the peak-to-peak height of the longitudinal velocity increases, while the swimming becomes to slow down. Because of the plane symmetry of the model and the lack of manipulation of swimming stability and direction, the vertical movement was unstable. The tail-beat frequency and phase difference were fixed to focus on the effect of body amplitude, and it should be noted that swimming performance depends on a combination of many factors rather than a single one. Thus, the optimal amplitude may change under different conditions, and this requires a further study.

Figure 5.30 shows the enlarged profiles of the longitudinal velocities for four cases with different body amplitudes from A=0.2 to A=0.5, during which the optimal value is obtained (0.3~0.35). The postures of the fish model at several times in one tail-beat cycle for the case of A=0.3 is also included as an example to illustrate the effect of swinging movement of the fish body on swimming velocity. Figure 5.31 shows the velocity vectors and color contours for the four cases, while the corresponding vortex structures are shown in Fig. 5.32.

As can be seen from the profile of the longitudinal velocity, in a tail-beat cycle, the swing movement of the fish body, especially the tail part, does not always bring the effect of acceleration. There are two accelerating processes corresponding to the time from about 10s to 10.6s and 11s to 11.5s, when the swinging movement of the body has a backward component and induces a reversed Karman vortex street. Between the acceleration there are two decelerating processes corresponding to time from about 10.6s to 11s and 11.5s to 12s, when the swinging movement has a forward component against the swimming direction. The mean swimming velocity depends on the integrated effect of the two processes. When the amplitude is larger than the optimal value, the effect of deceleration exceeds that of the acceleration, which causes an overall decrease of swimming velocity. From Figure 5.31, as the amplitude increases, the maximum of the swinging velocity increases, generating a stronger wake flow, while a more intense reversed flow region is also yielded. From Fig. 5.32, when A=0.5, an obvious expansion of the wake vortices on the pair vertical walls against the swimming direction can be observed compared with other cases.

In this study, the boundary conditions were set as non-slip walls along the vertical and lateral directions, and the vortex was thus confined to the channel and could not diffuse into the far field. This might be a little different from the normal condition where the flow field was considerably wider. Therefore, to assess the effects of the limitations imposed by the walls on swimming performance, simulations using enlarged grids were carried out. Two cases (cases 2 and 3) were examined with grids expanded along the lateral direction, with double width for case 2 and triple width for case 3, meanwhile, case 4 (double height) and 5 (triple height) with grids expanded along the vertical direction were examined. We used the computational condition for the aforementioned case 1 with parameters A = 0.25, $\omega = \pi$ and $\phi = \pi/6$, and the same initial conditions was imposed for cases 2, 3, 4 and 5. The displacements and velocity components against time in case 1, 2 and 3are compared in Fig. 5.33, the results of case 1, 4 and 5 are shown in Fig. 5.34.

From Fig. 5.33, the effect of the distance between the lateral walls on swimming performance was shown to be insignificant along all three directions. Figure 5.34 showed that the effect of the channle height was also unobvious, while a slight instability in the vertical direction was observed. The relatively low Reynolds number in this study may be the reason leading to the absence of significant reflection from the wall. Meanwhile, on the other hand, even though an additional drag arised due to the wall, it may be offset to some extent by increasing thrust.

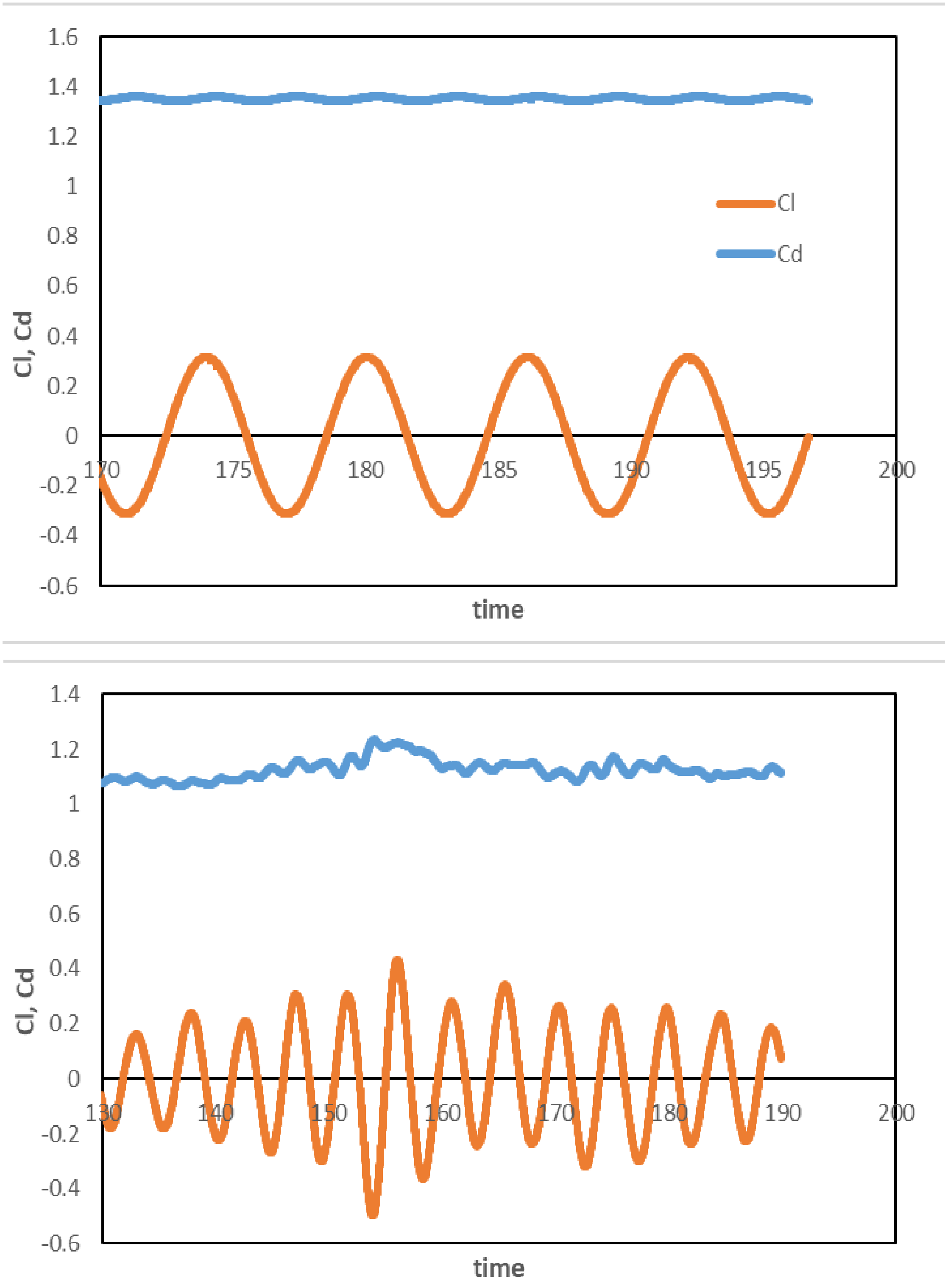


Figure 5.1: Lift and drag coefficients against time for Re = 100 (upper) and 1000 (lower).

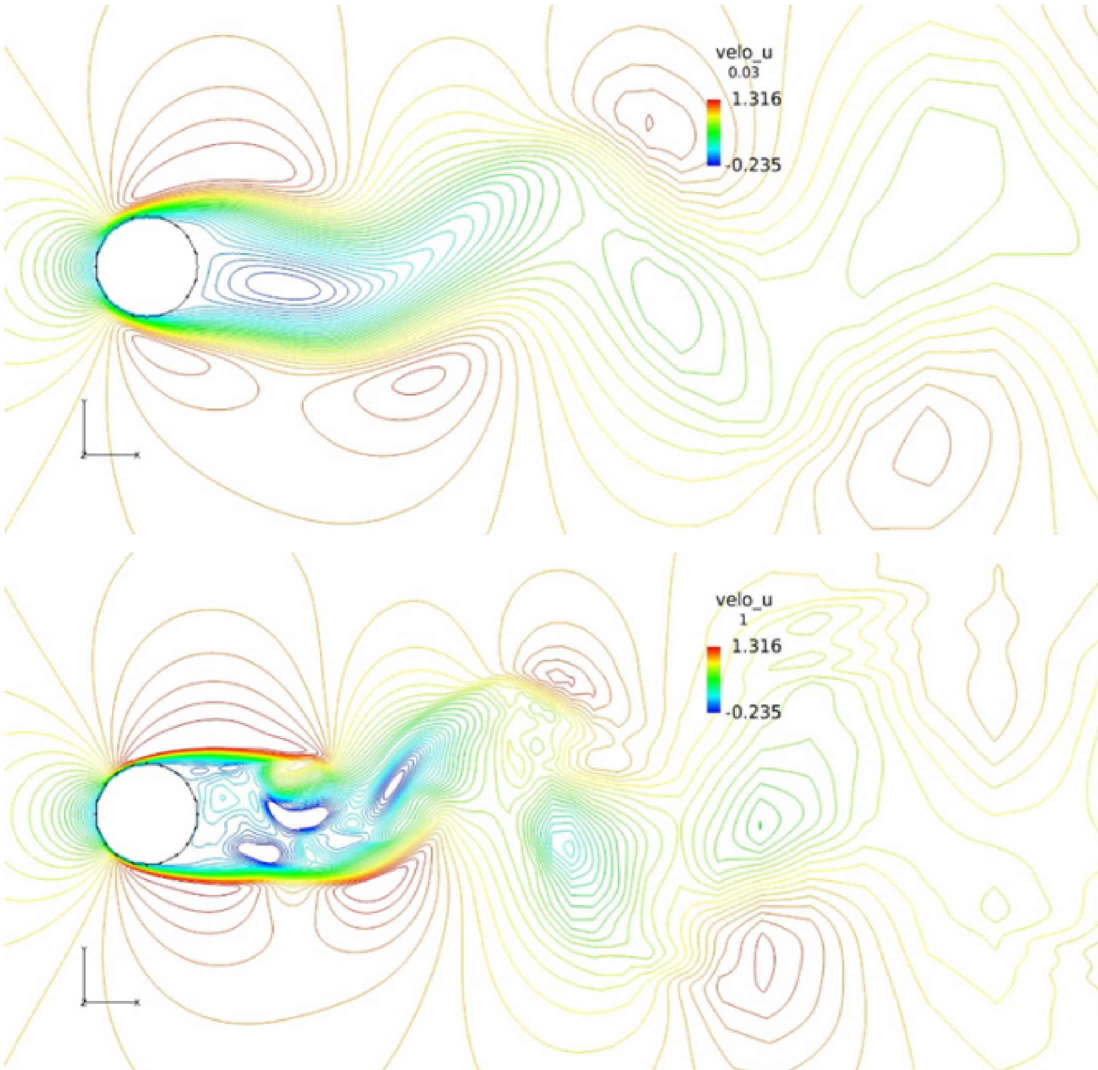


Figure 5.2: Instantaneous streamwise velocity contours on the center plane for Re=100 (upper) and 1000 (lower).

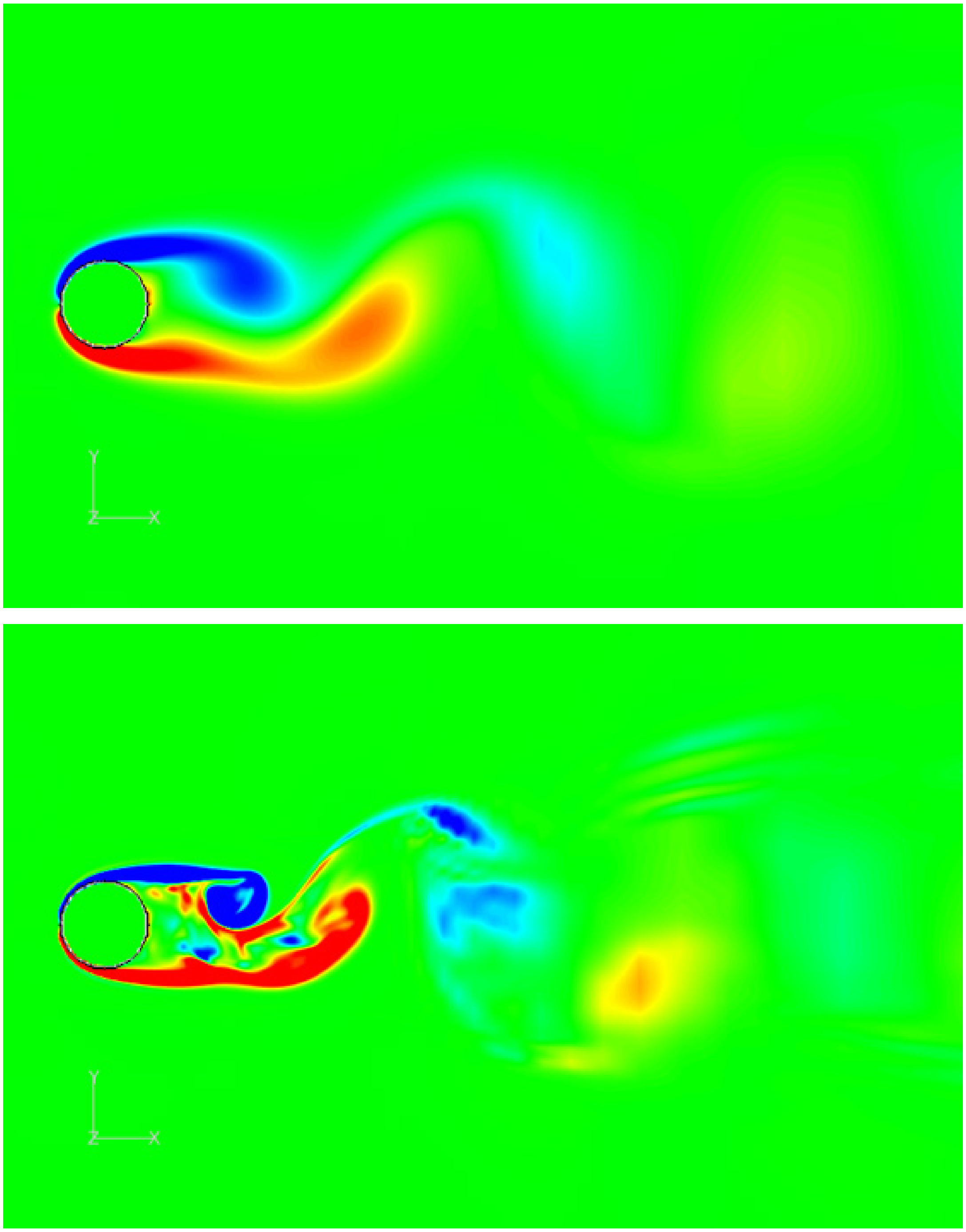


Figure 5.3: Vorticity contours on the center plane for Re=100 (upper) and 1000 (lower), where blue for clockwise and red for counter-clockwise.

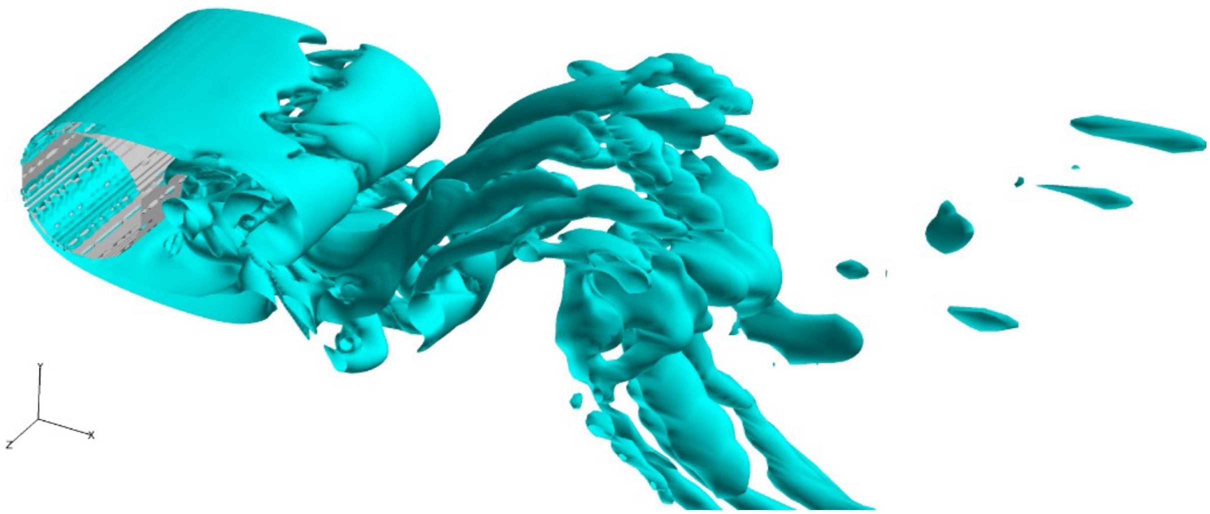


Figure 5.4: Vortex structures for the case of Re=1000.

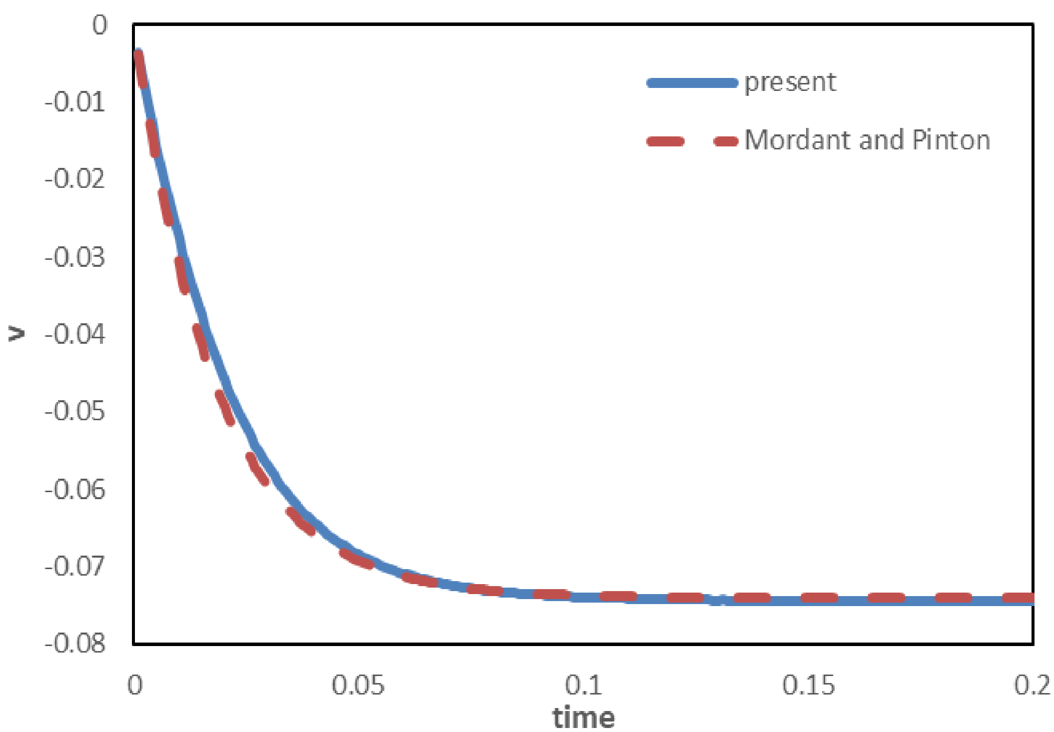


Figure 5.5: Velocity profiles for a single settling particle.

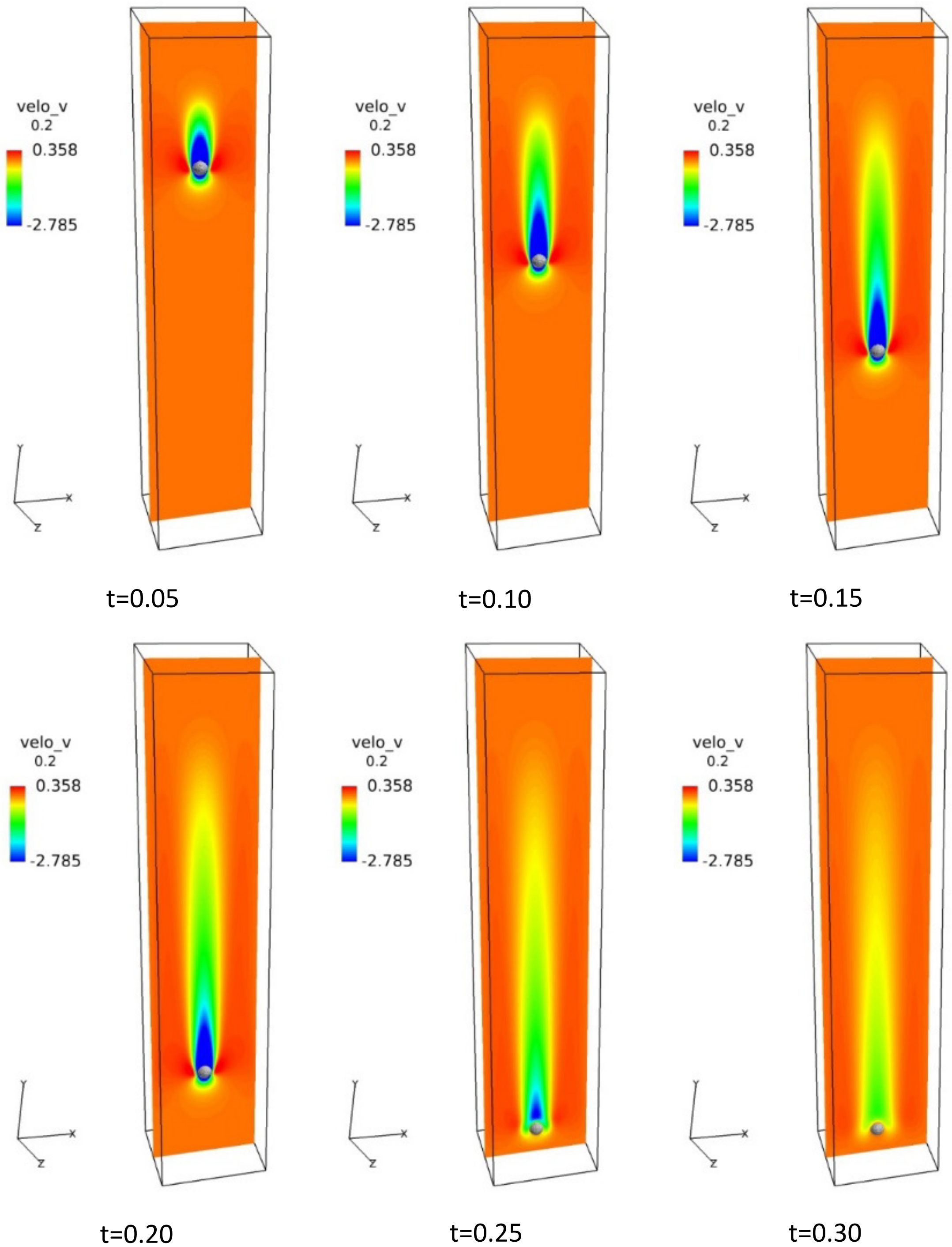
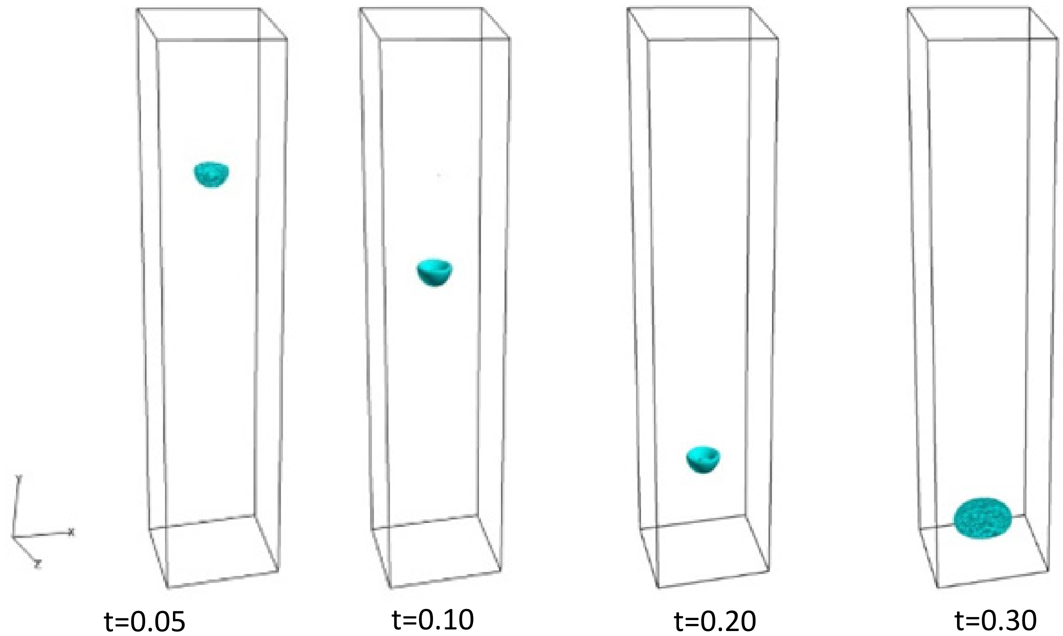
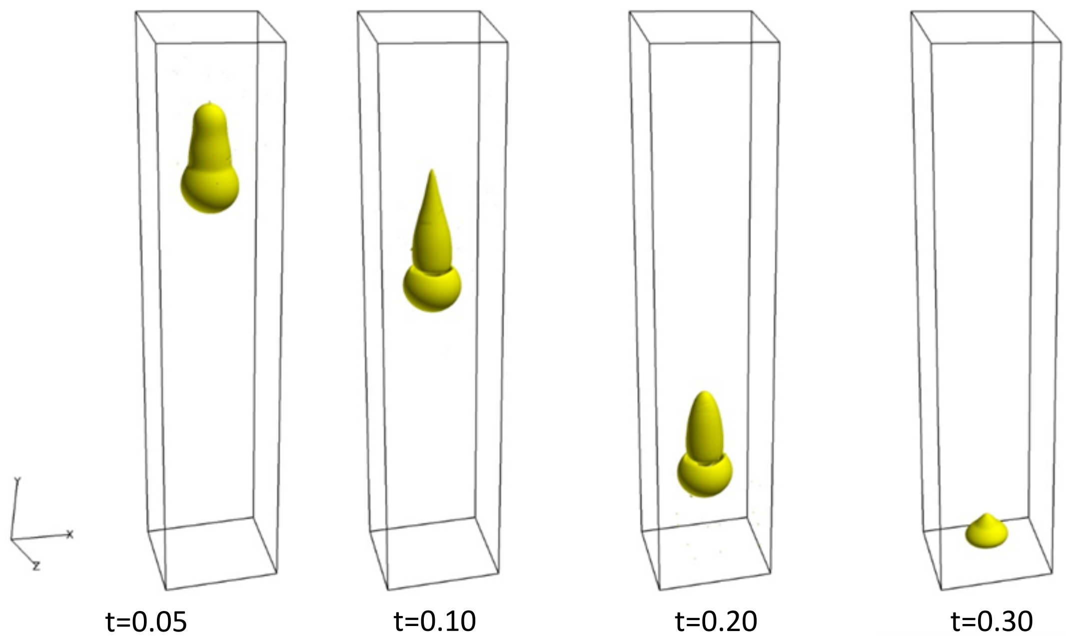


Figure 5.6: Color contours of velocity on the central plane.



(a)



(b)

Figure 5.7: Vortex structures around the settling particle with second invariant of the velocity-gradient tensor of (a) 10 and (b) -20.

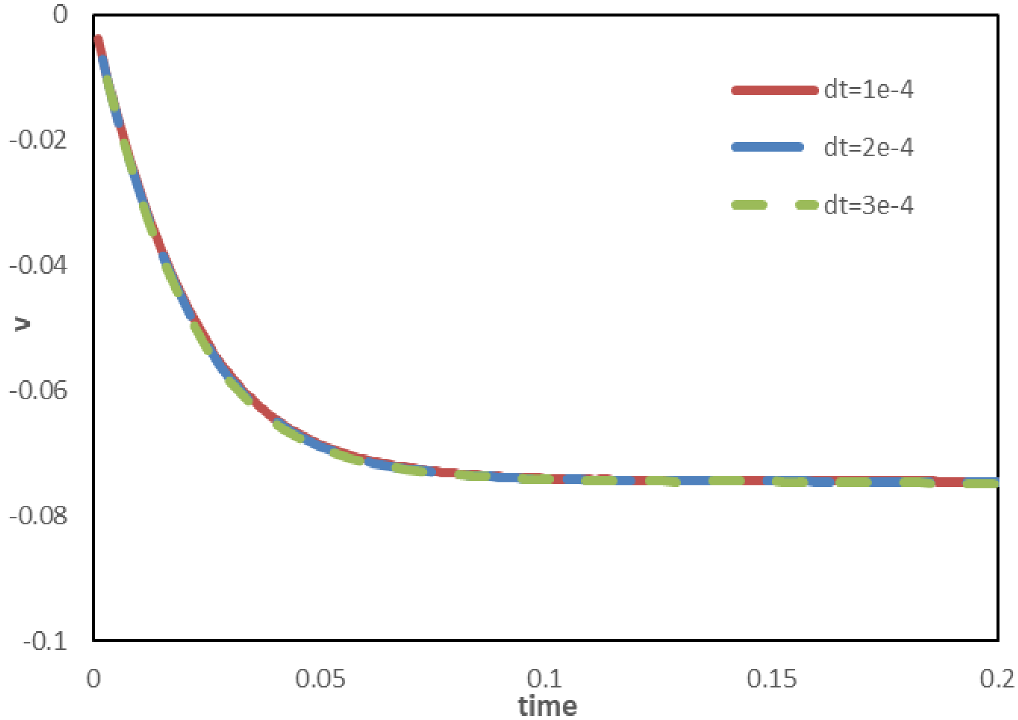


Figure 5.8: Velocity profiles for a single settling particle with different time steps.

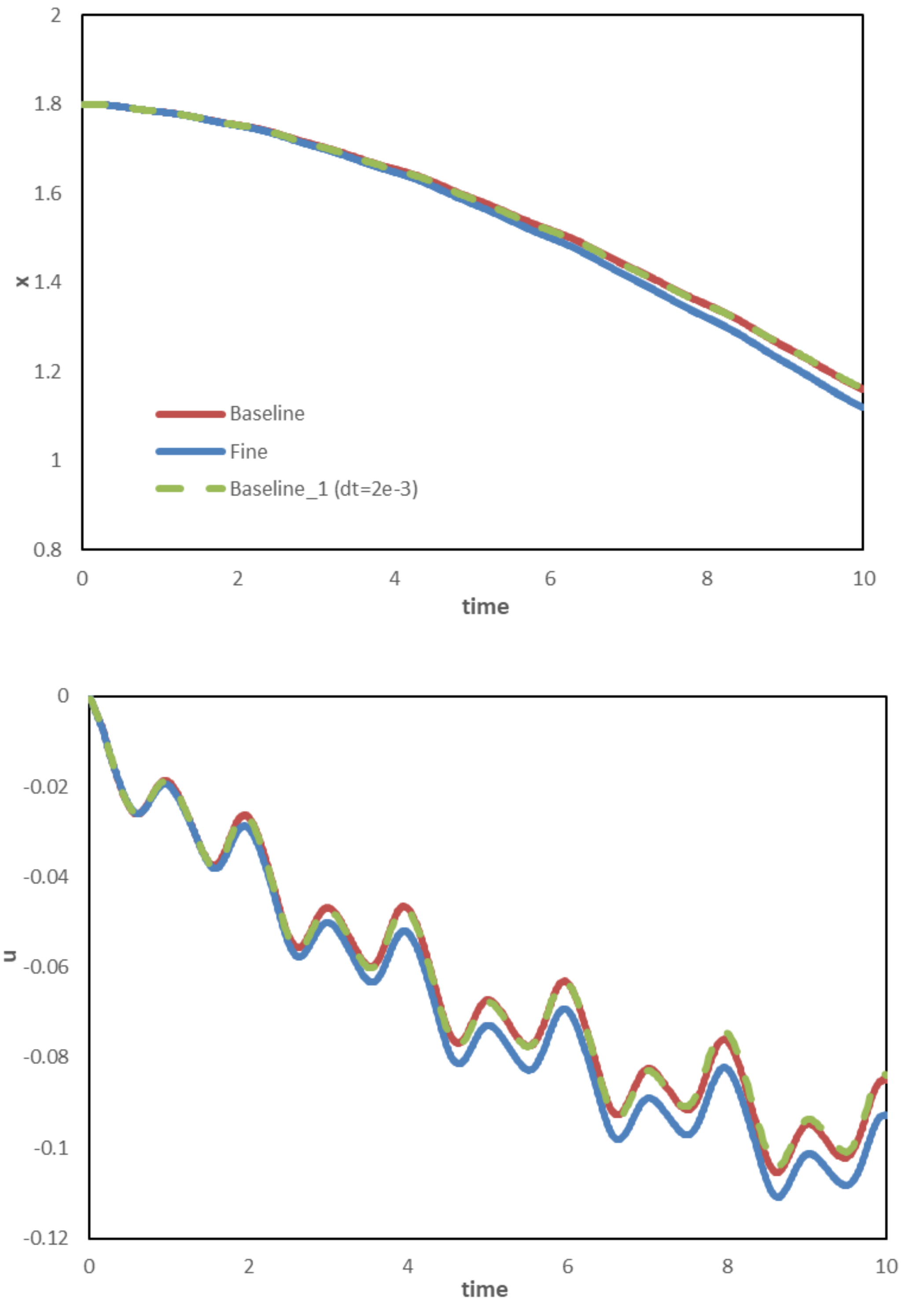


Figure 5.9: Comparison of longitudinal displacement (upper) and velocity (lower) against time between the cases with different time steps or grid nodes (A=0.25, $\omega = \pi$, $\phi = \pi/6$).

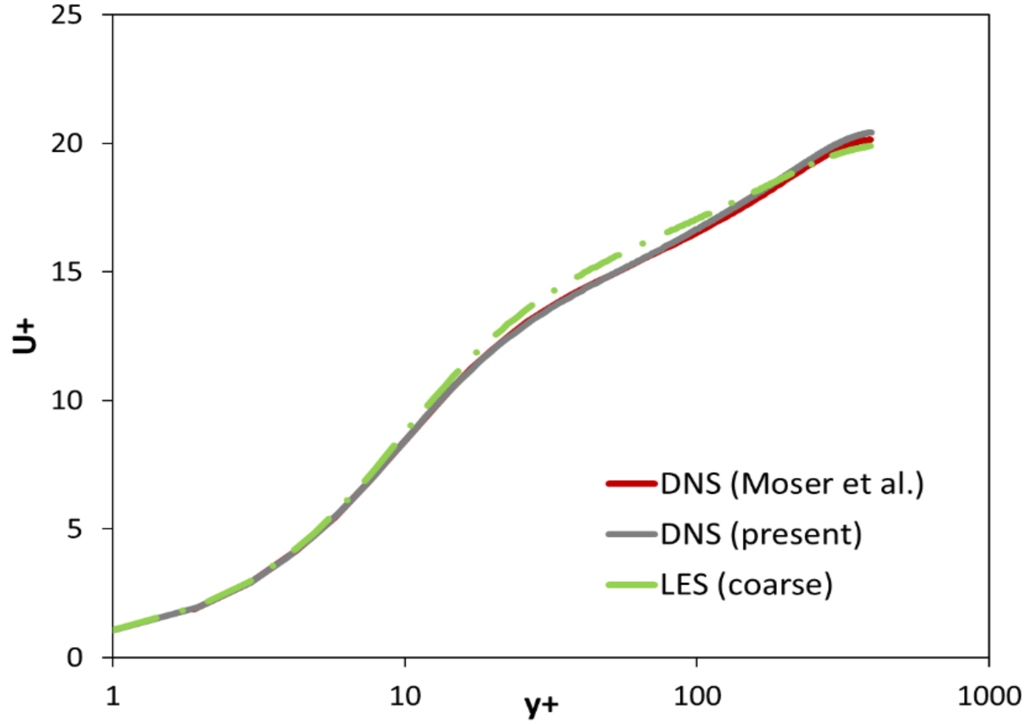
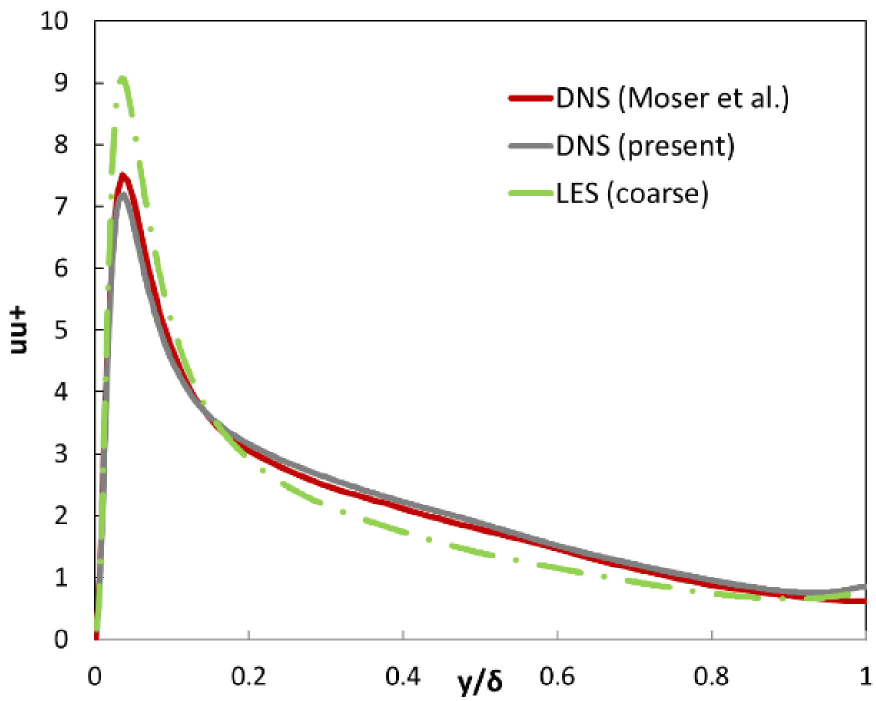
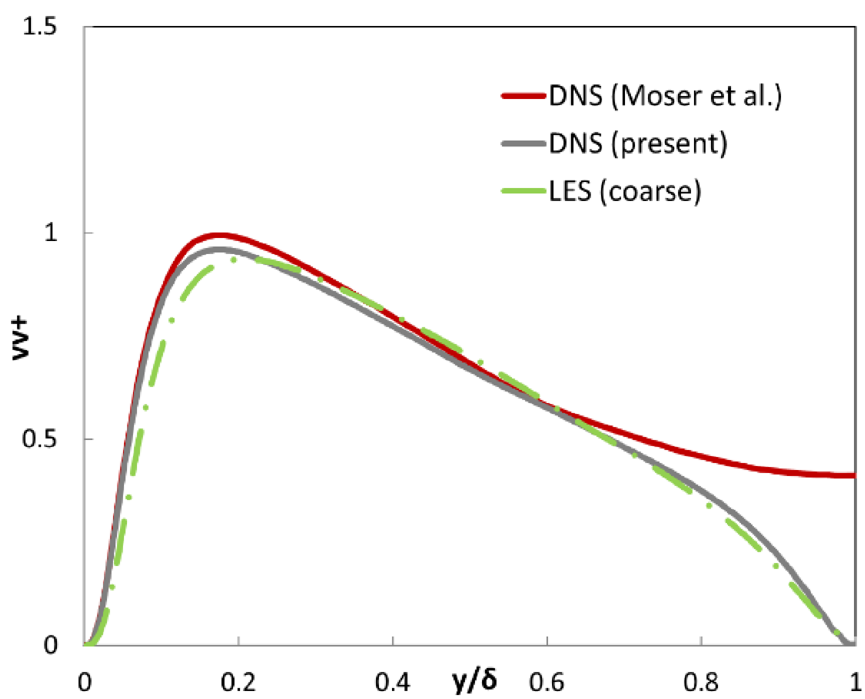
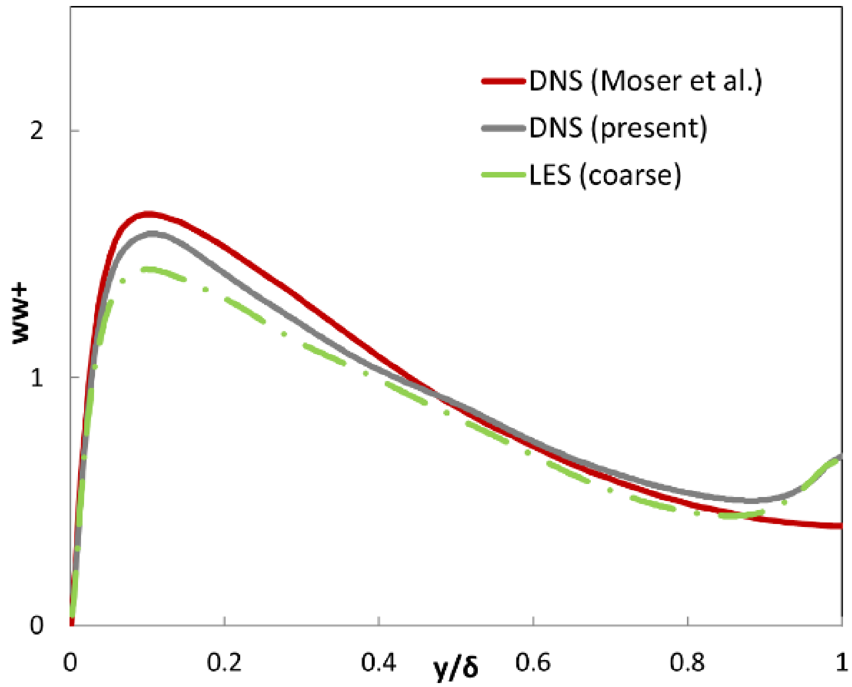


Figure 5.10: Mean streamwise velocity for the smooth wall


DNS (Moser et al.)
DNS (present)
LES (coarse)
uu+
y/δ
0
1
2
3
4
5
6
7
8
9
10
0
0.2
0.4
0.6
0.8
1


DNS (Moser et al.)
DNS (present)
LES (coarse)
vv+
y/δ
0
0.5
1
1.5
0
0.2
0.4
0.6
0.8
1



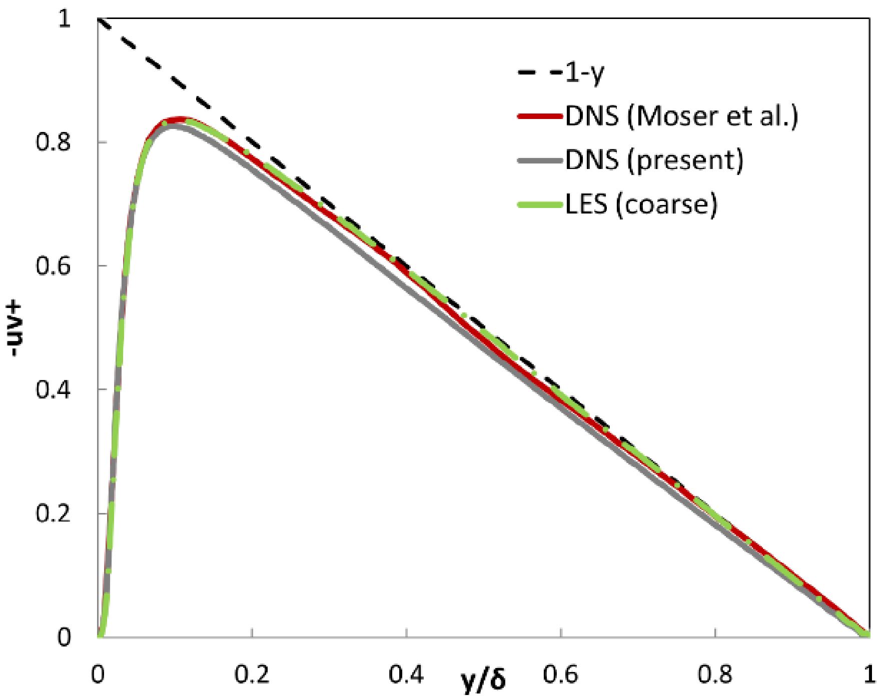


Figure 5.11: Comparison of Reynolds stresses for flow along a smooth wall

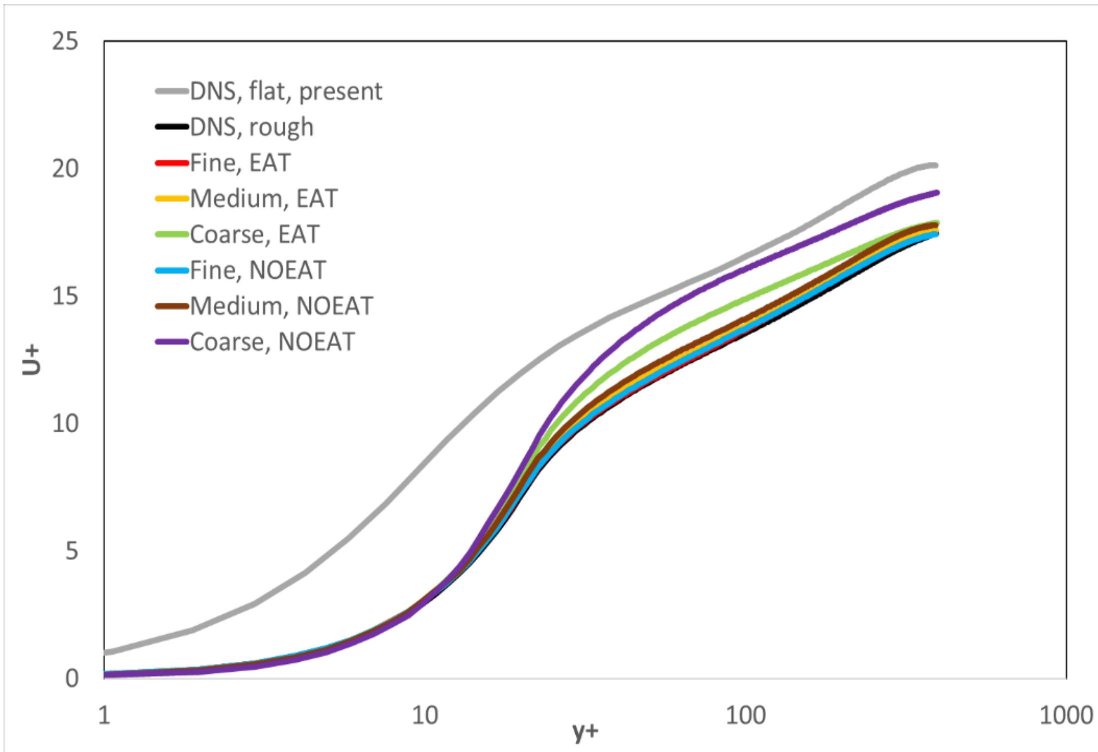


Figure 5.12: Comparison of the mean streamwise velocity for rough-wall cases with smooth-wall DNS

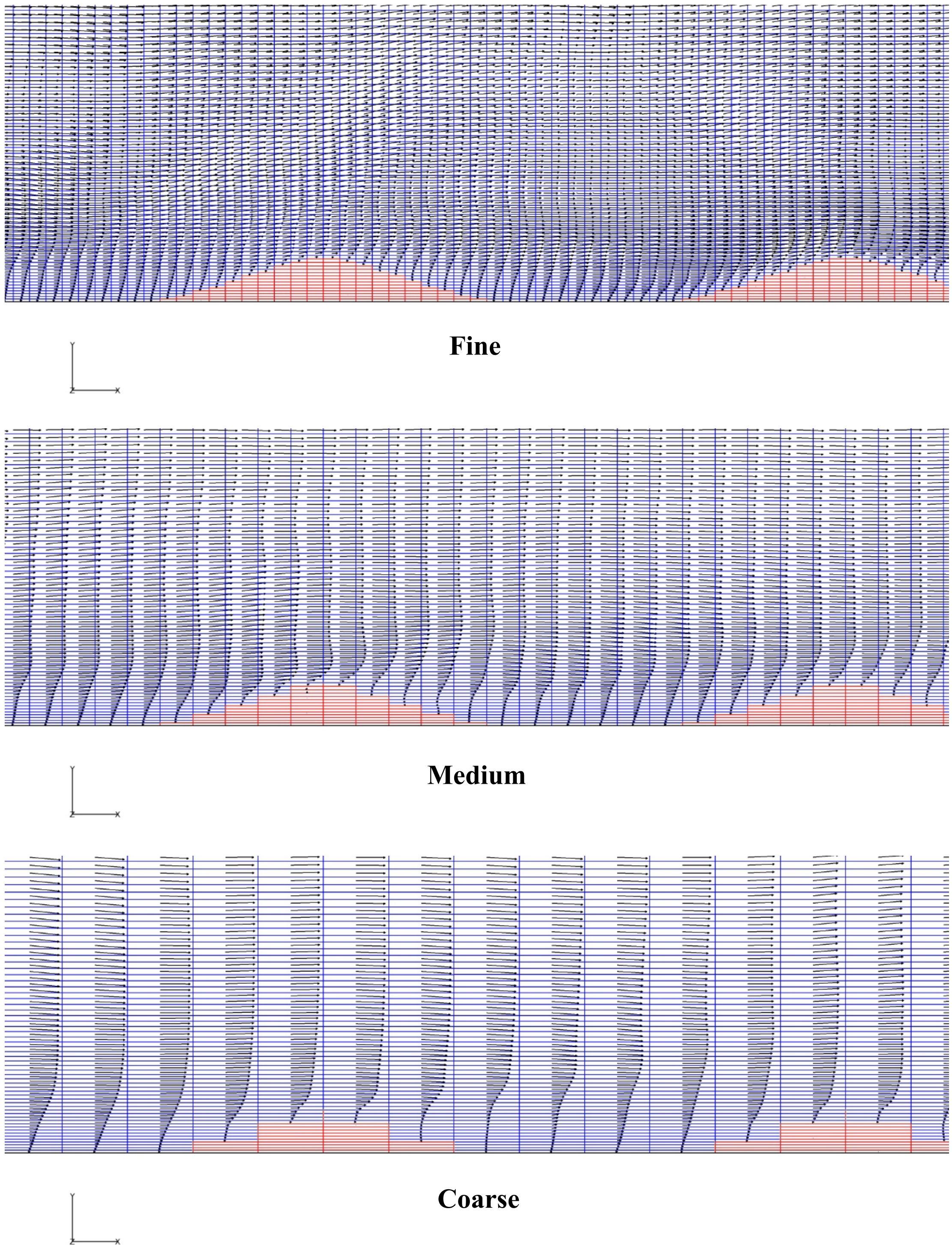


Figure 5.13: Instantaneous velocity with roughness shape at z=0 (with EAT)

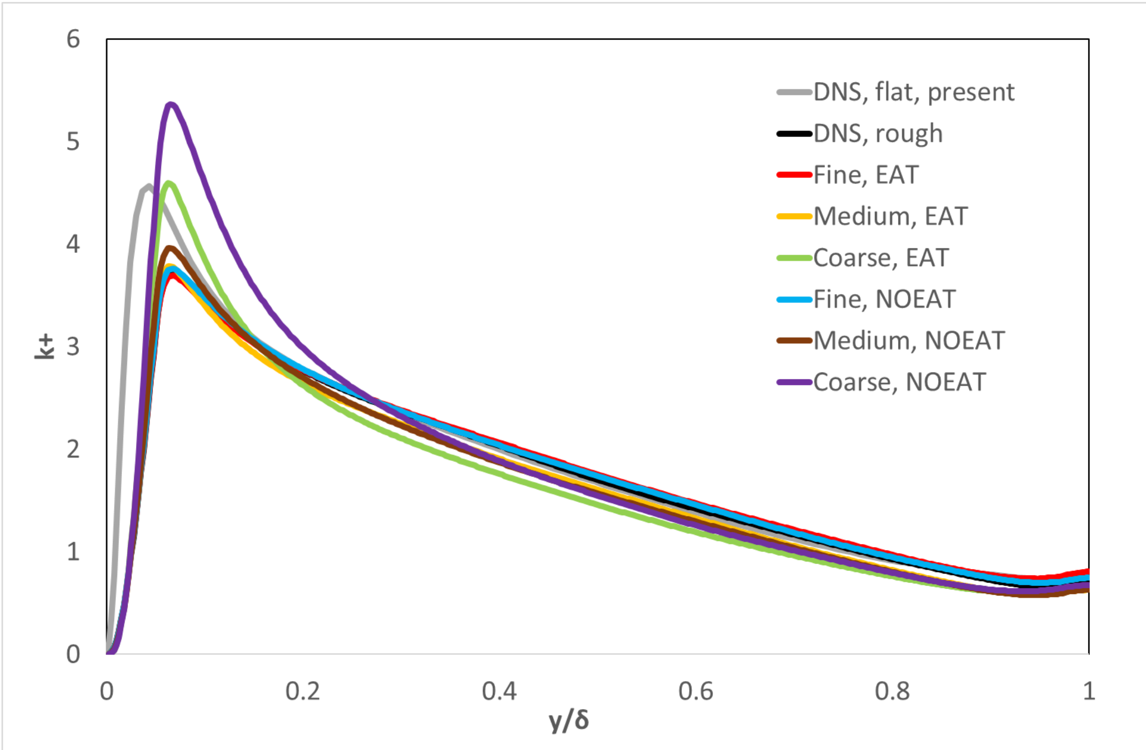
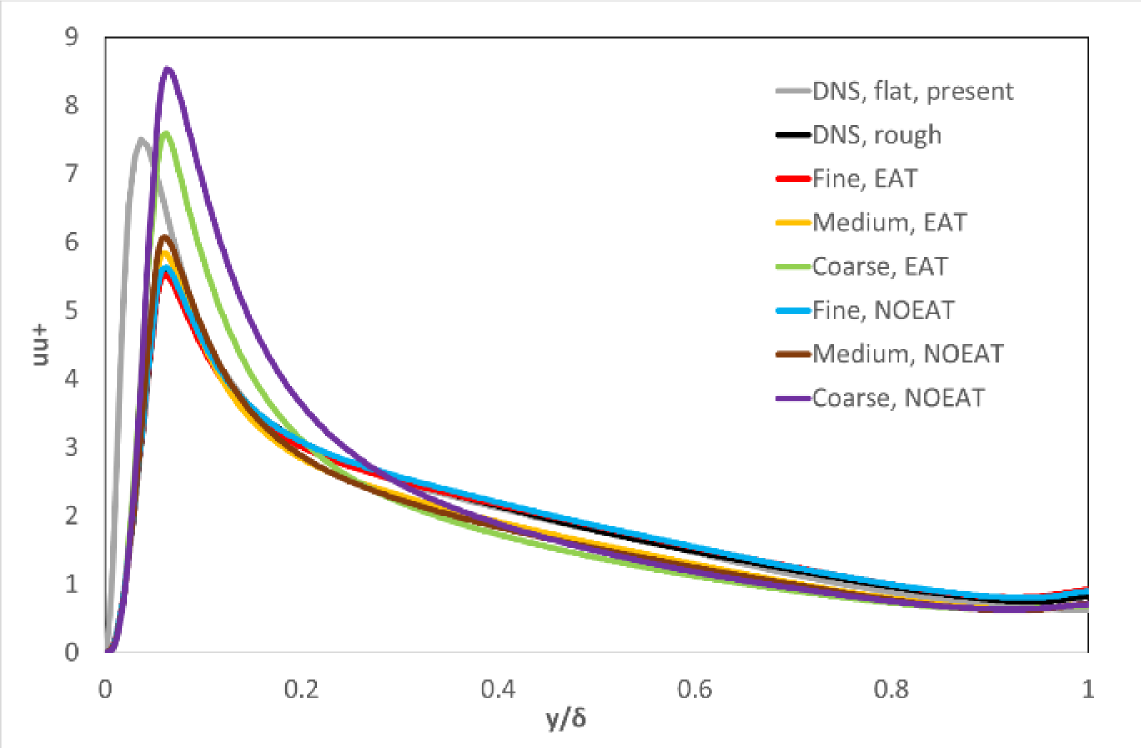
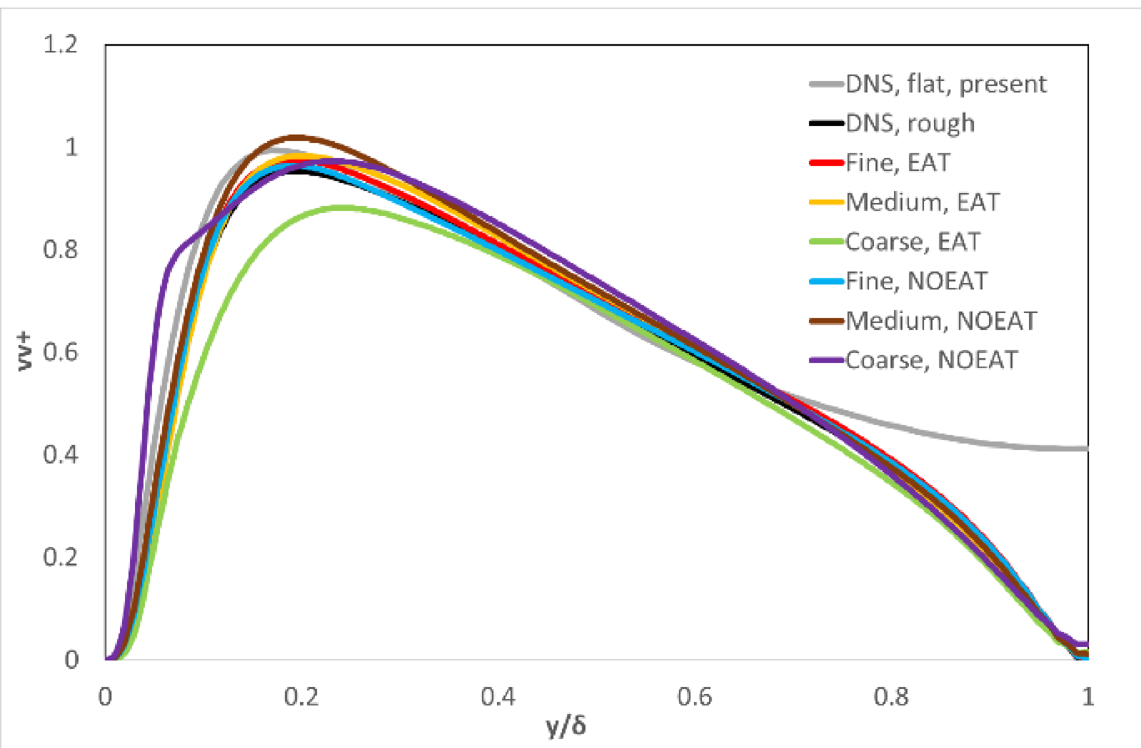


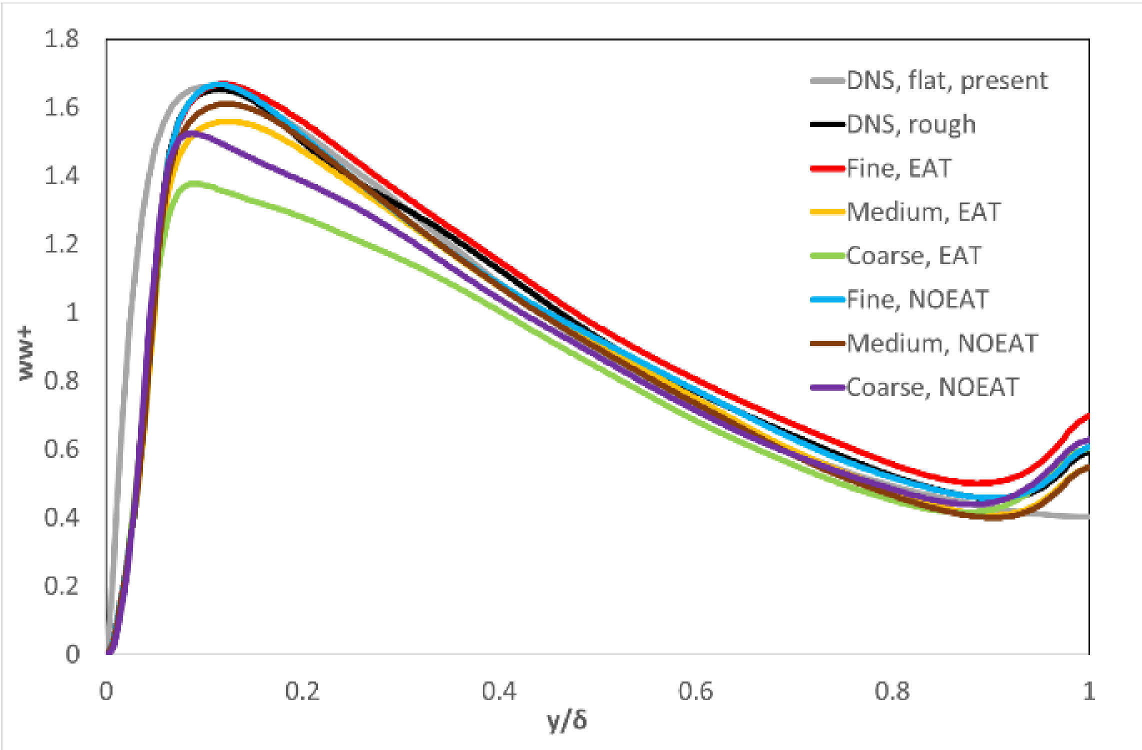
Figure 5.14: Comparison of turbulence energy distributions



9
8
7
6
5
4
3
2
1
0
uu+
0
0.2
0.4
0.6
0.8
1
y/δ
DNS, flat, present
DNS, rough
Fine, EAT
Medium, EAT
Coarse, EAT
Fine, NOEAT
Medium, NOEAT
Coarse, NOEAT



1.2
1
0.8
0.6
0.4
0.2
0
vv+
0
0.2
0.4
0.6
0.8
1
y/δ
DNS, flat, present
DNS, rough
Fine, EAT
Medium, EAT
Coarse, EAT
Fine, NOEAT
Medium, NOEAT
Coarse, NOEAT



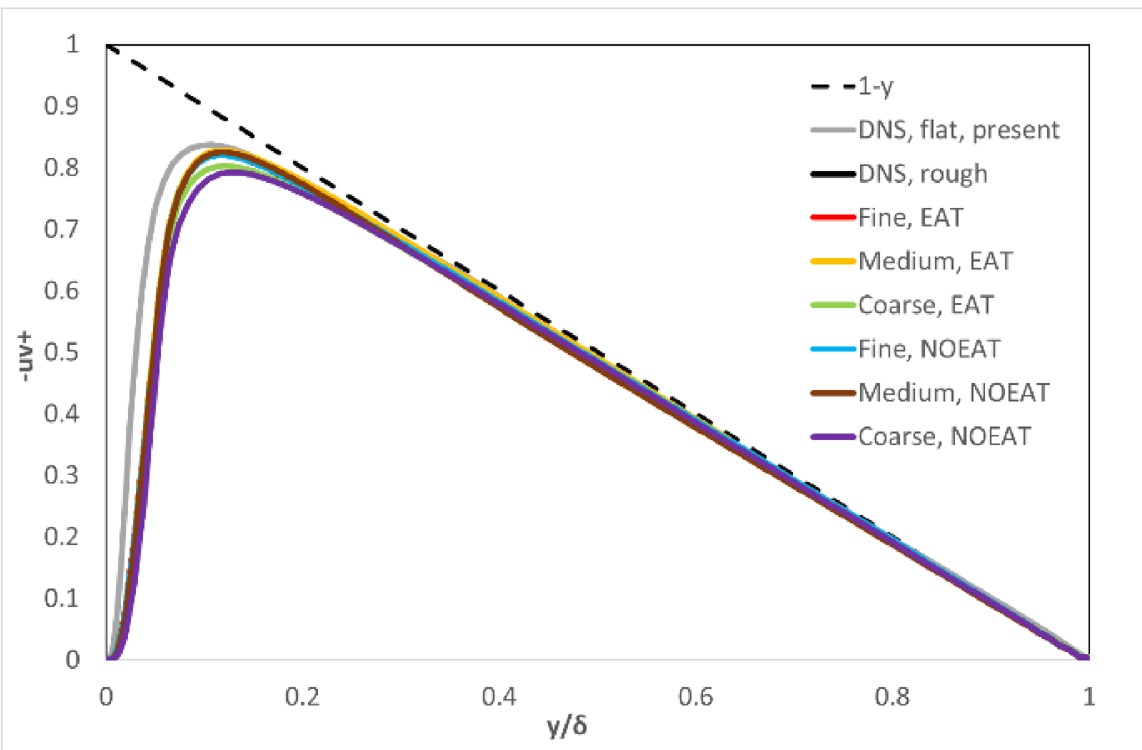


Figure 5.15: Comparison of Reynolds stress for the rough-wall cases and DNS cases (smooth and rough)

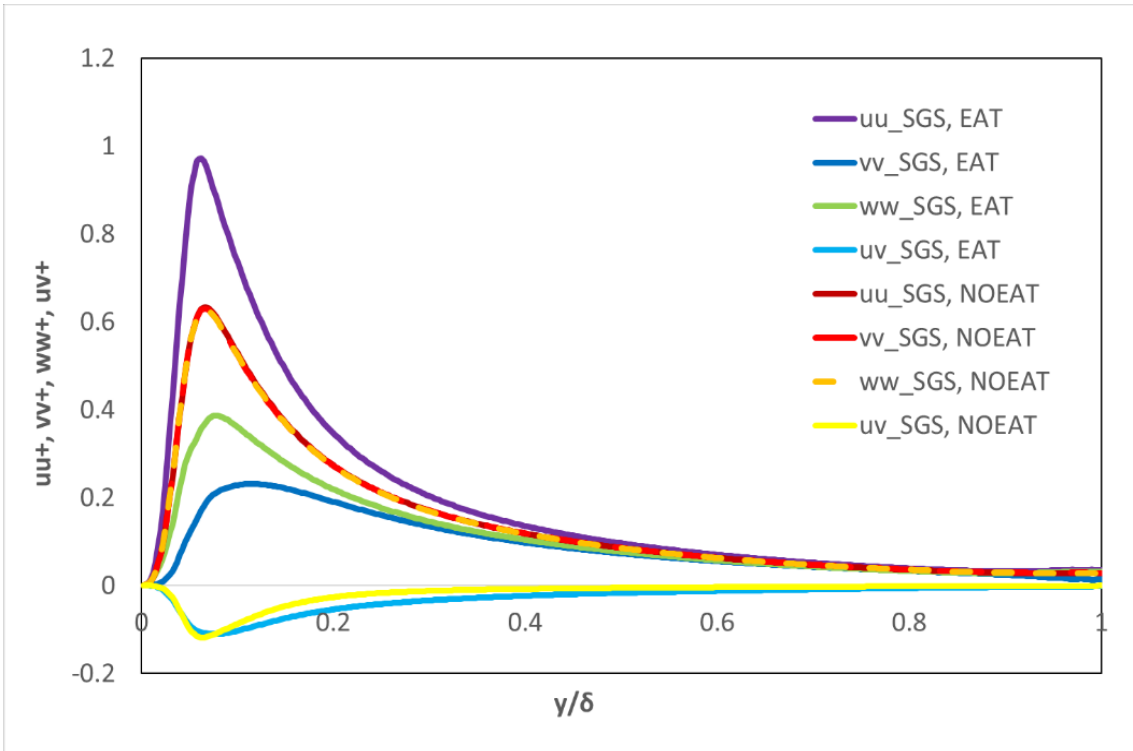
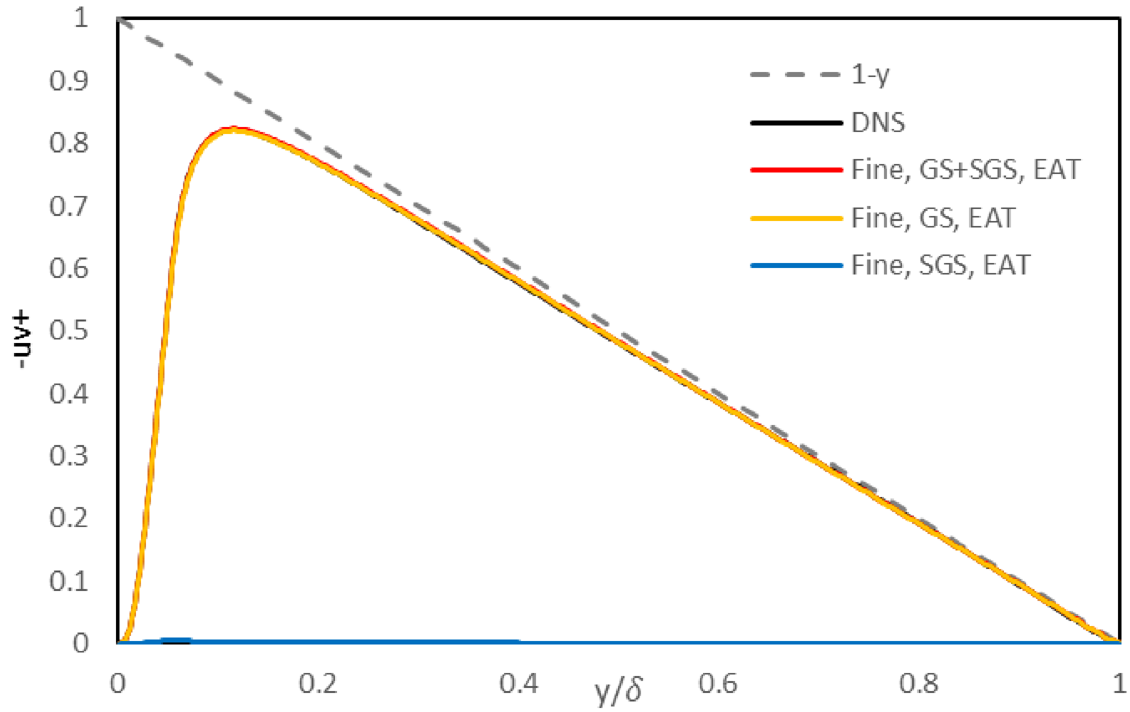
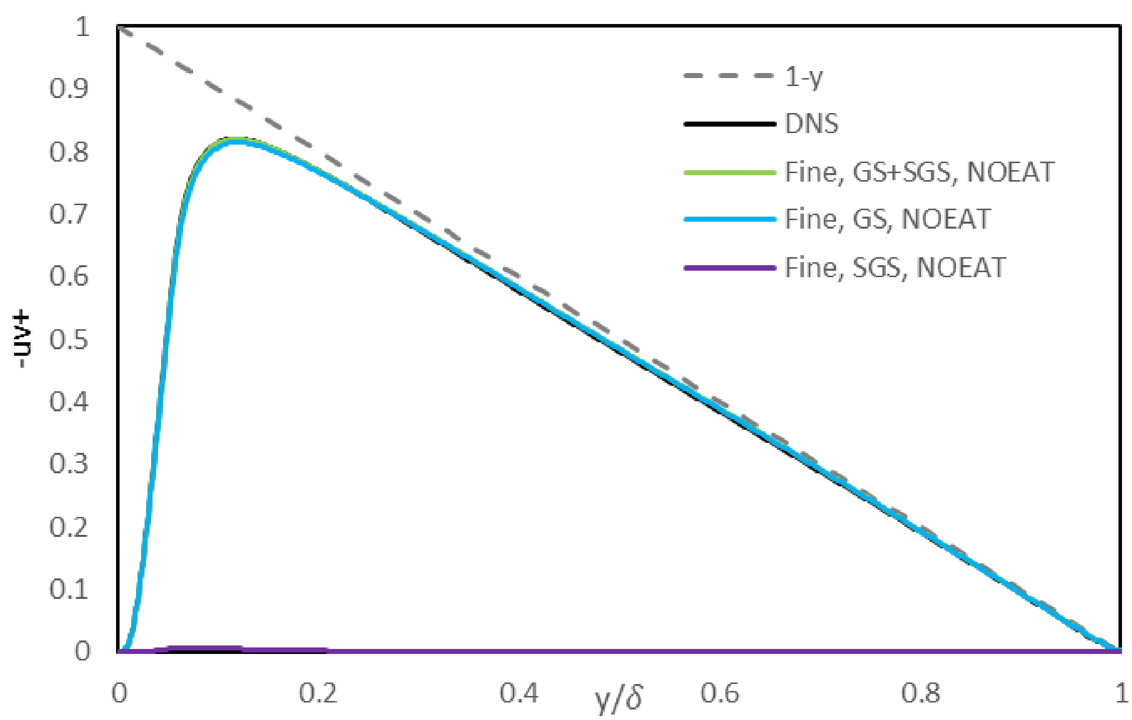
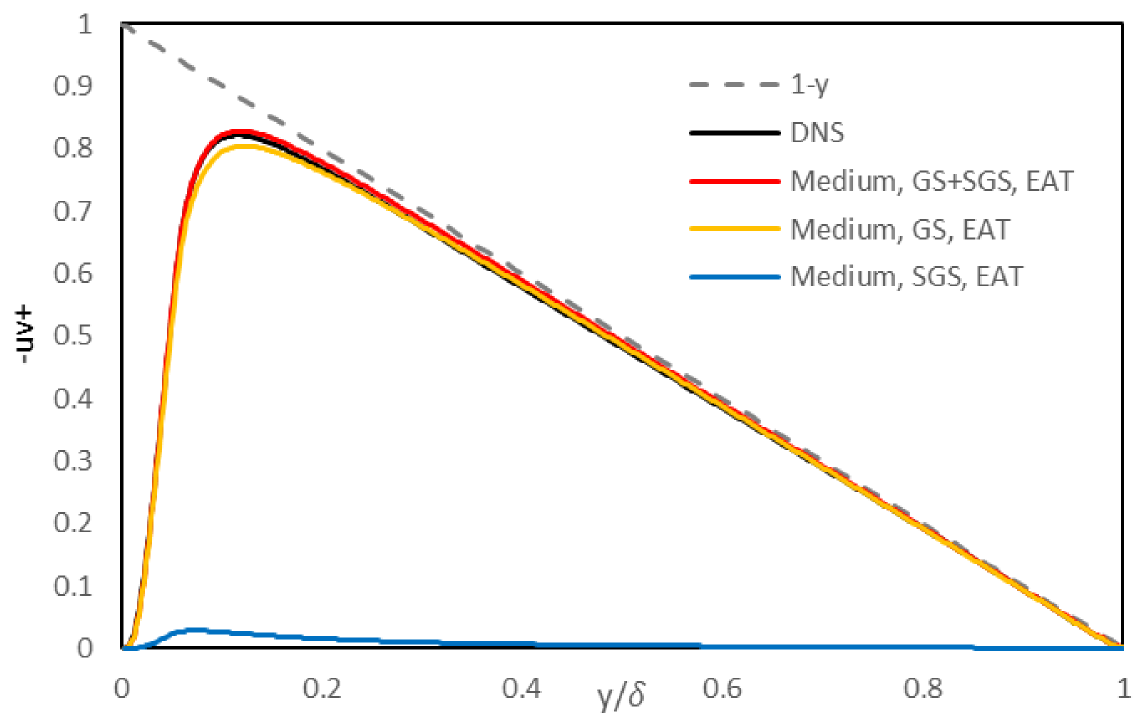
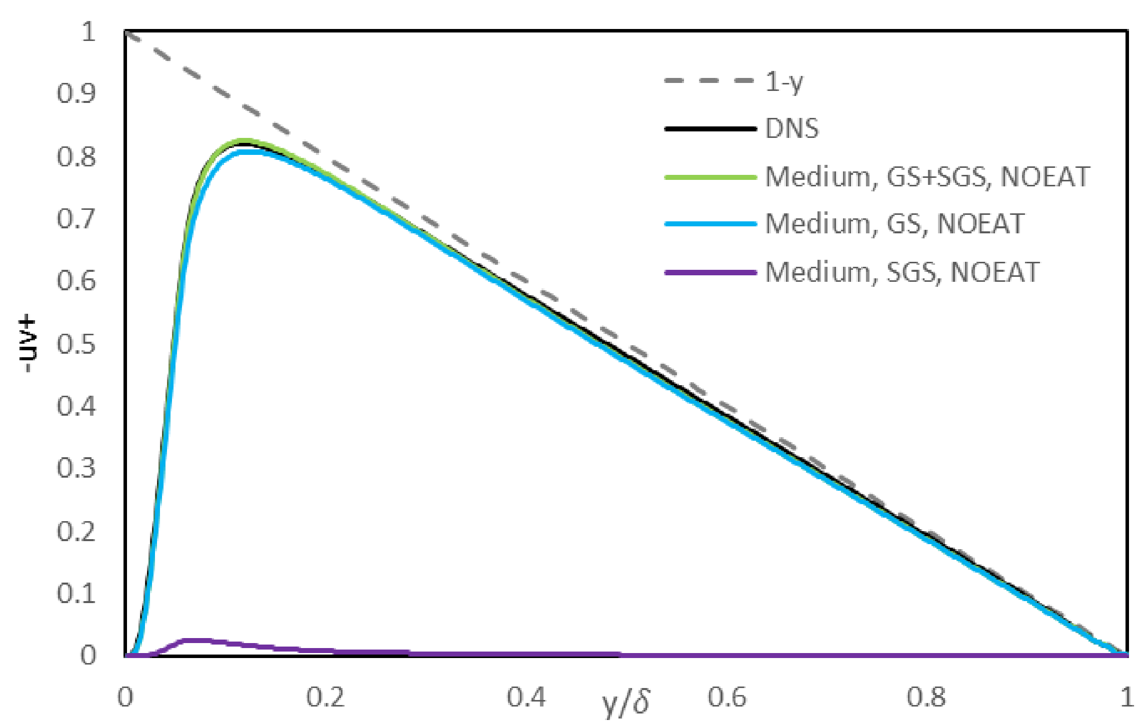


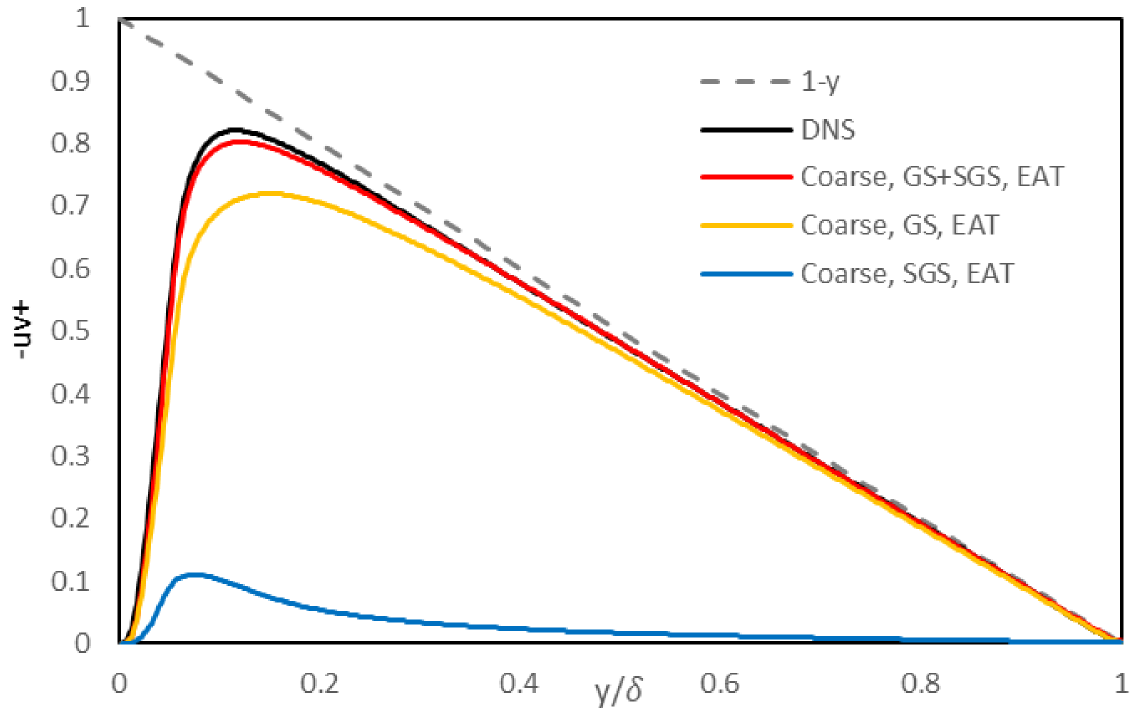
Figure 5.16: Comparison of the SGS stress components for the coarse cases.


1-y
DNS
Fine, GS+SGS, EAT
Fine, GS, EAT
Fine, SGS, EAT
-uv+
y/δ
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1


1-y
DNS
Fine, GS+SGS, NOEAT
Fine, GS, NOEAT
Fine, SGS, NOEAT
-uv+
y/δ
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.2
0.4
0.6
0.8
1


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-uv+
0
0.2
0.4
y/δ
0.6
0.8
1
1-y
DNS
Medium, GS+SGS, EAT
Medium, GS, EAT
Medium, SGS, EAT


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-uv+
0
0.2
0.4
y/δ
0.6
0.8
1
1-y
DNS
Medium, GS+SGS, NOEAT
Medium, GS, NOEAT
Medium, SGS, NOEAT



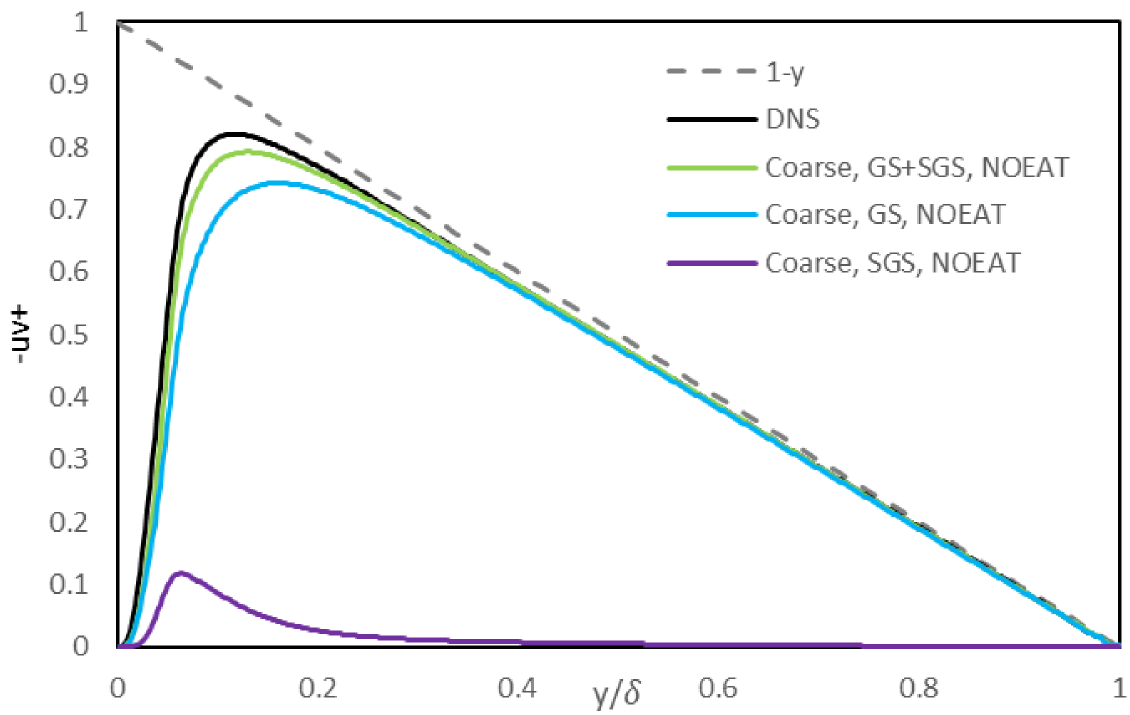
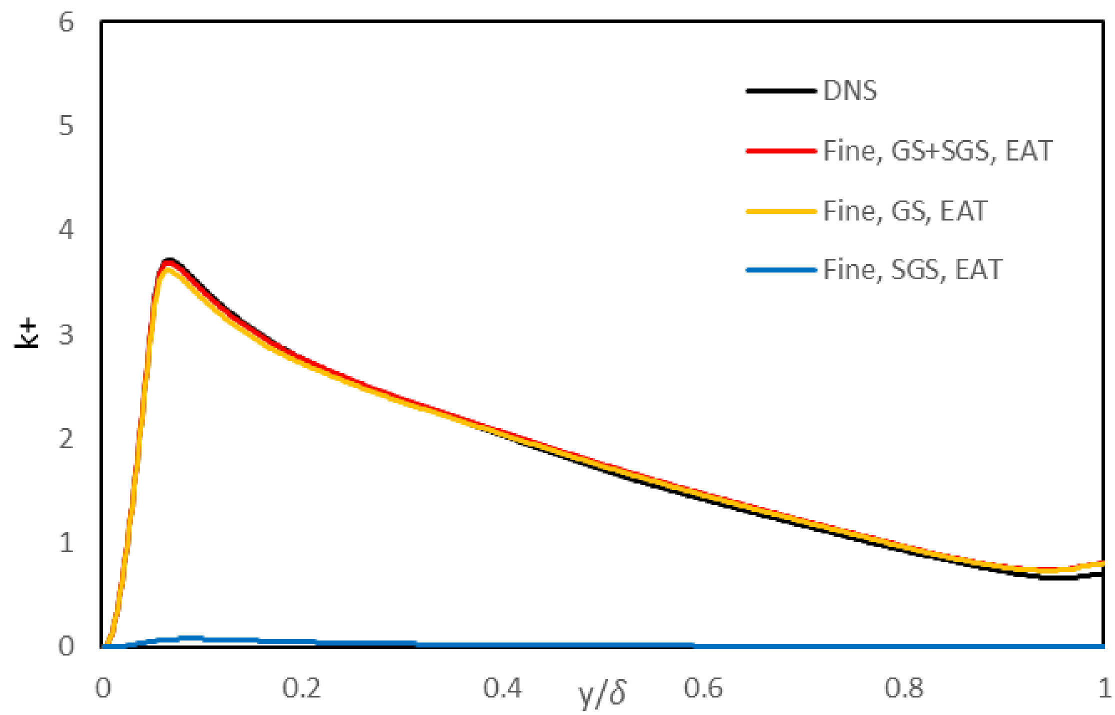
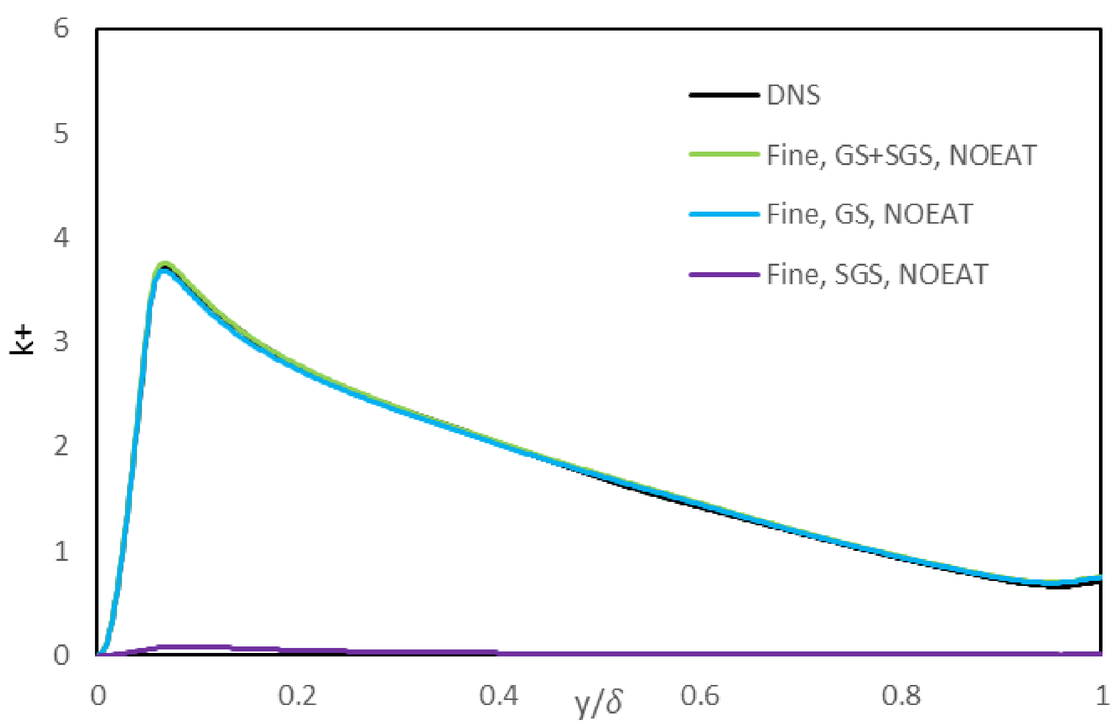
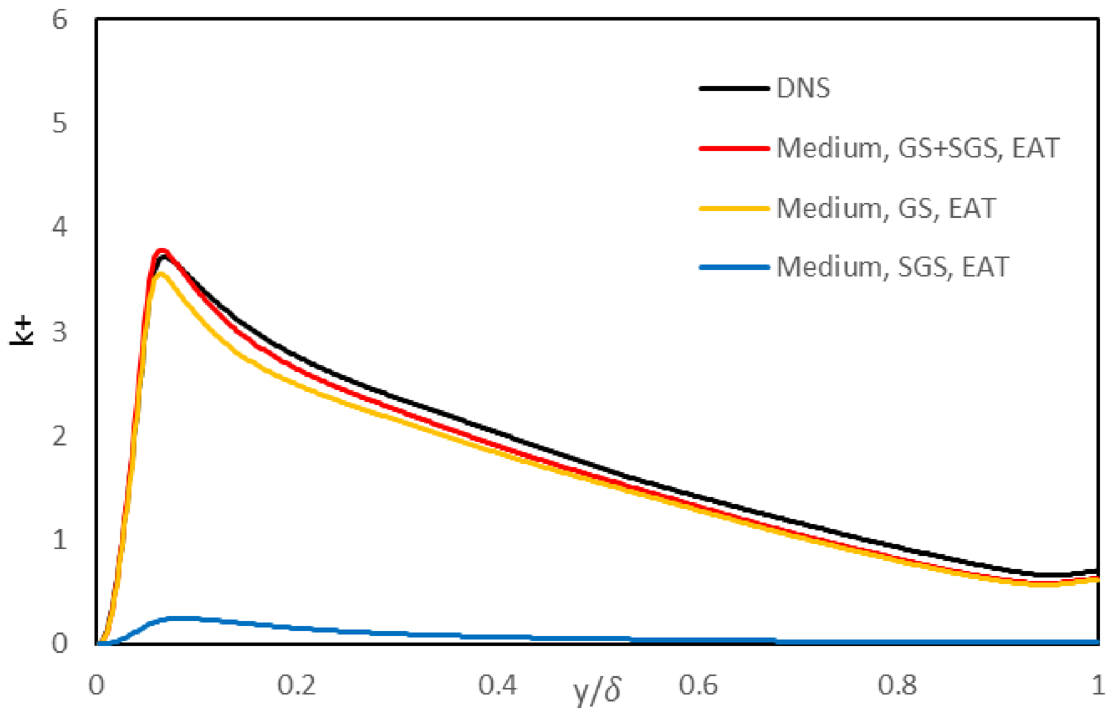
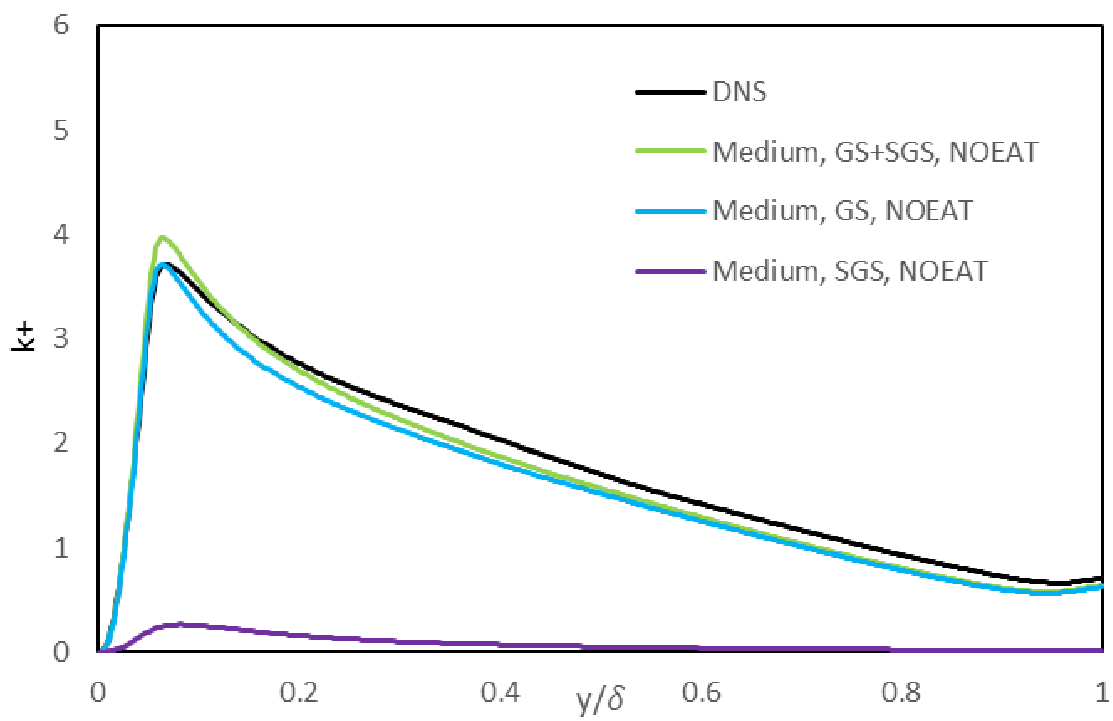


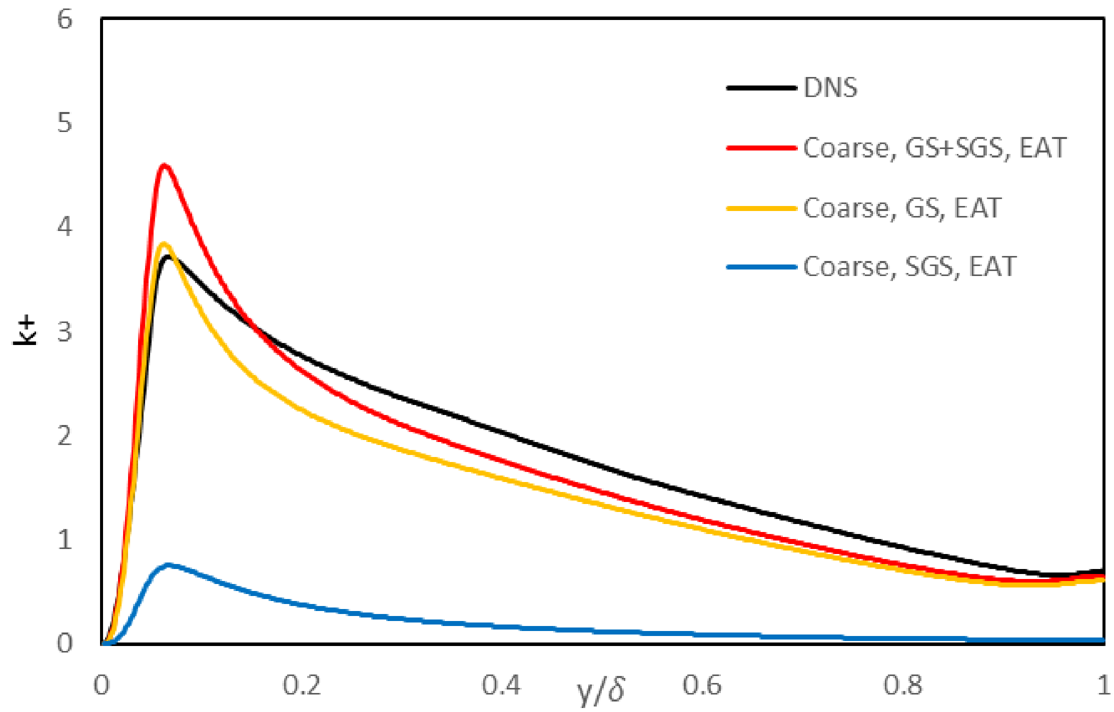
Figure 5.17: Comparison of the Reynolds shear stress.


6
5
4
3
2
1
0
k+
0
0.2
0.4
y/δ
0.6
0.8
1
DNS
Fine, GS+SGS, EAT
Fine, GS, EAT
Fine, SGS, EAT


6
5
4
3
2
1
0
k+
0
0.2
0.4
y/δ
0.6
0.8
1
DNS
Fine, GS+SGS, NOEAT
Fine, GS, NOEAT
Fine, SGS, NOEAT


k+
y/δ
DNS
Medium, GS+SGS, EAT
Medium, GS, EAT
Medium, SGS, EAT


k+
y/δ
DNS
Medium, GS+SGS, NOEAT
Medium, GS, NOEAT
Medium, SGS, NOEAT



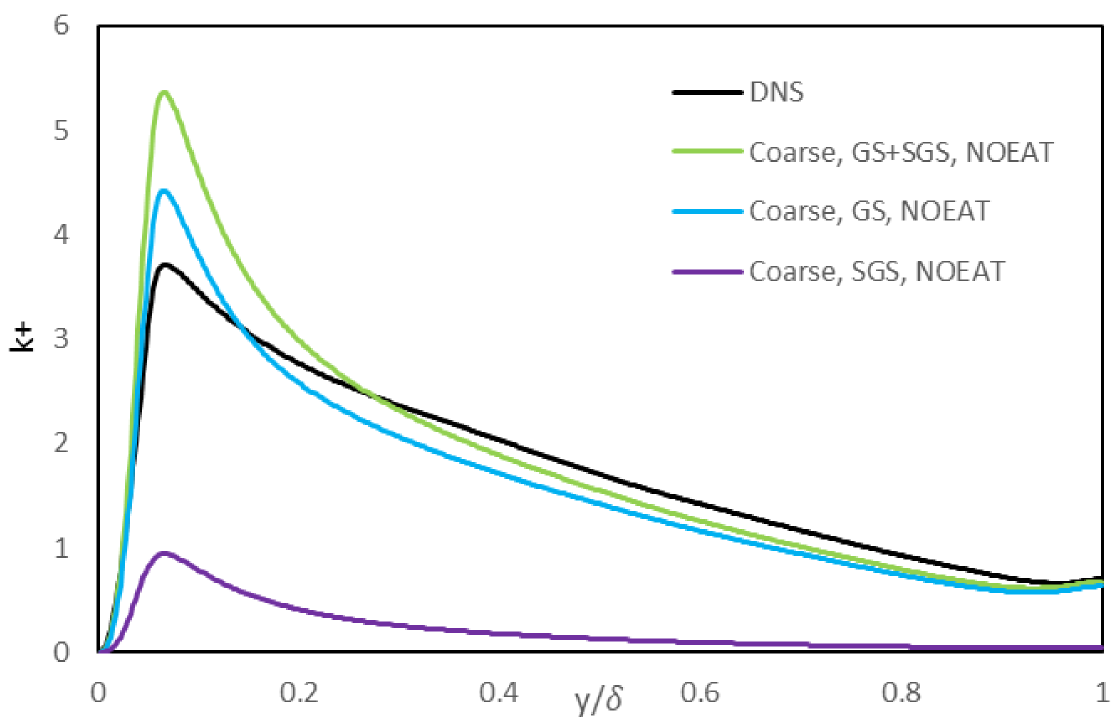


Figure 5.18: Comparison of the turbulence energy.

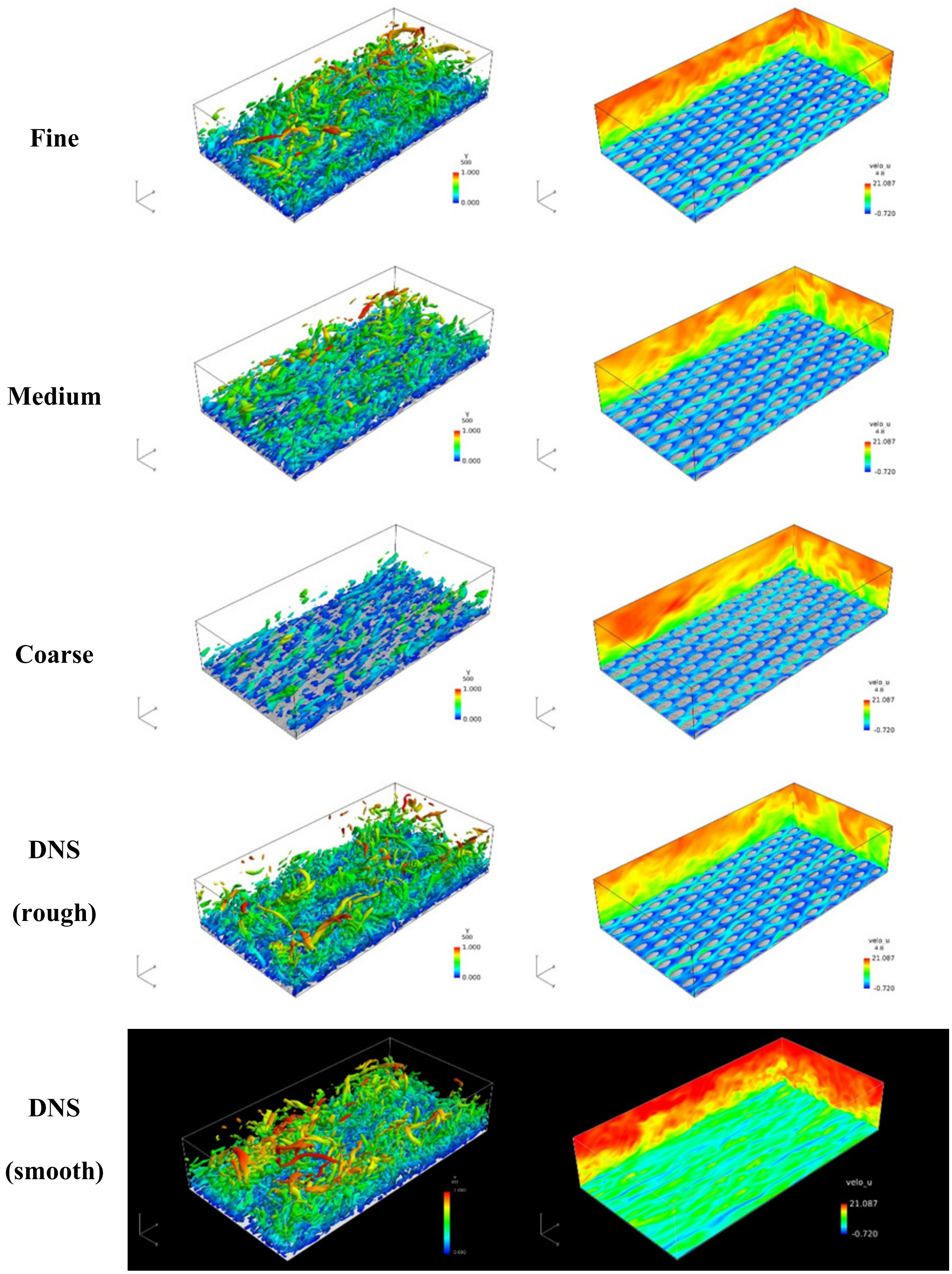
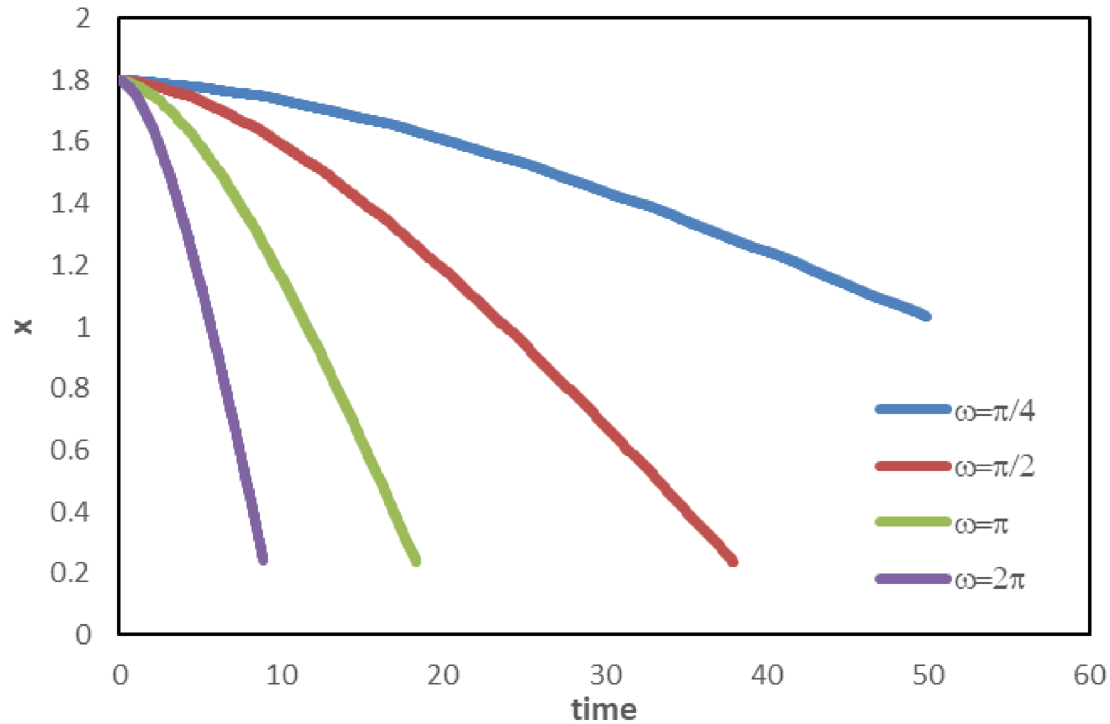
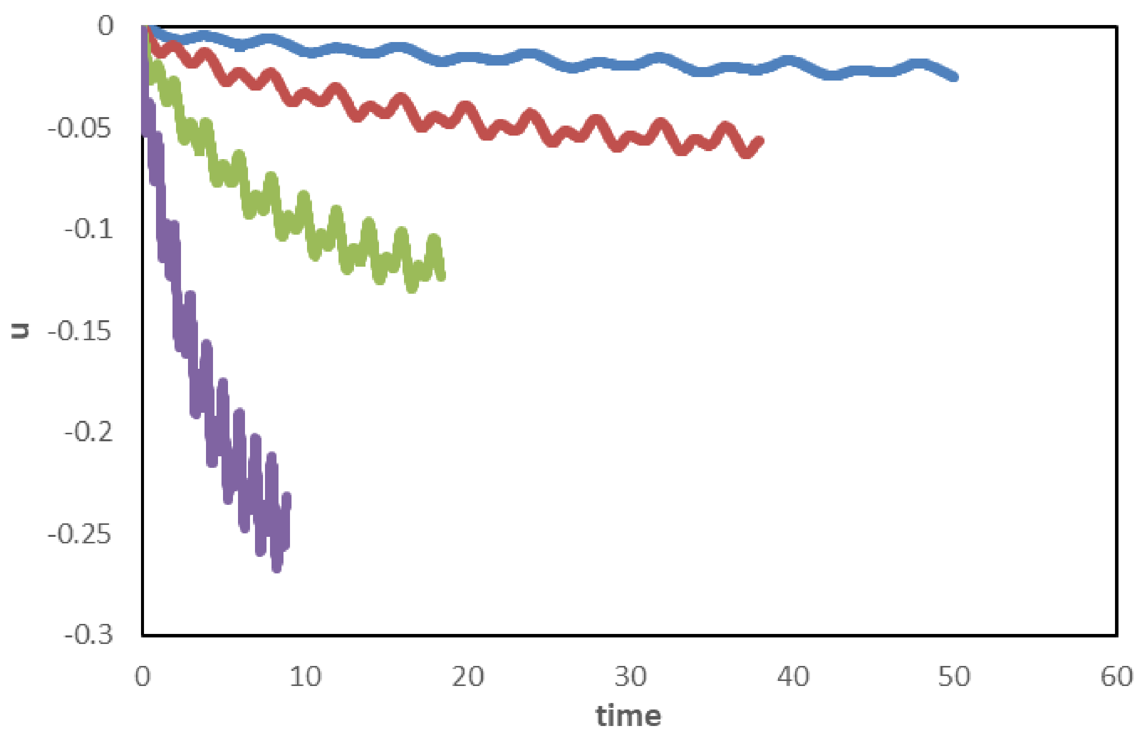


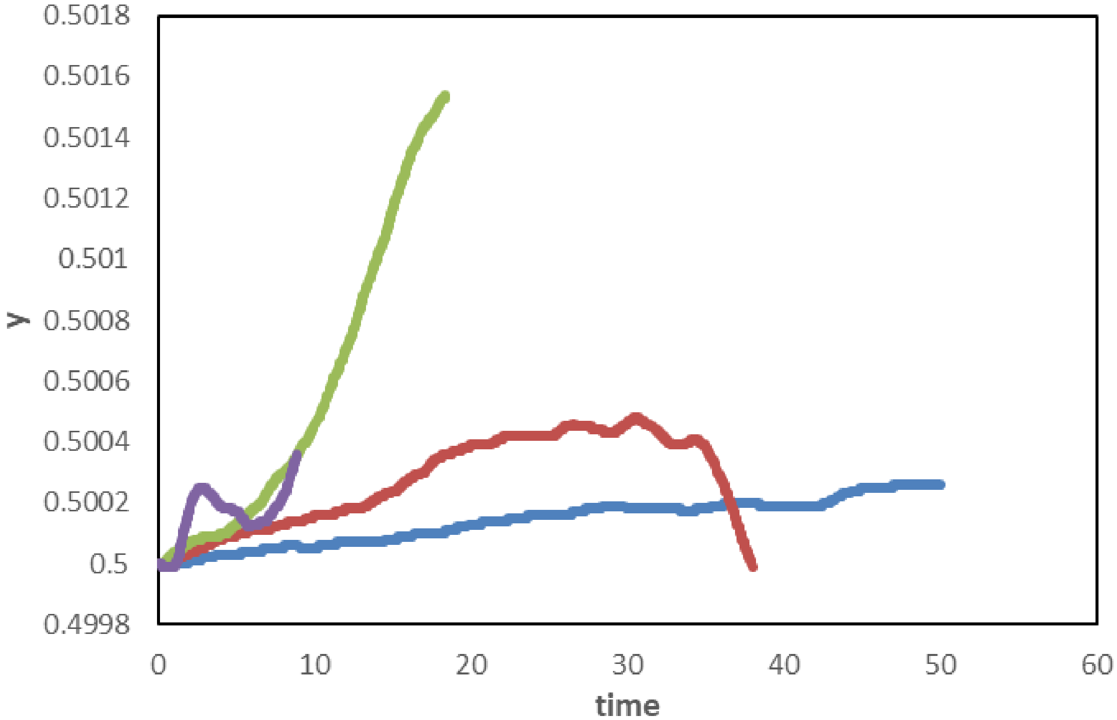
Figure 5.19: Instantaneous vortex structures and streamwise velocity (4.8, 0.02, 0) contours (with EAT)

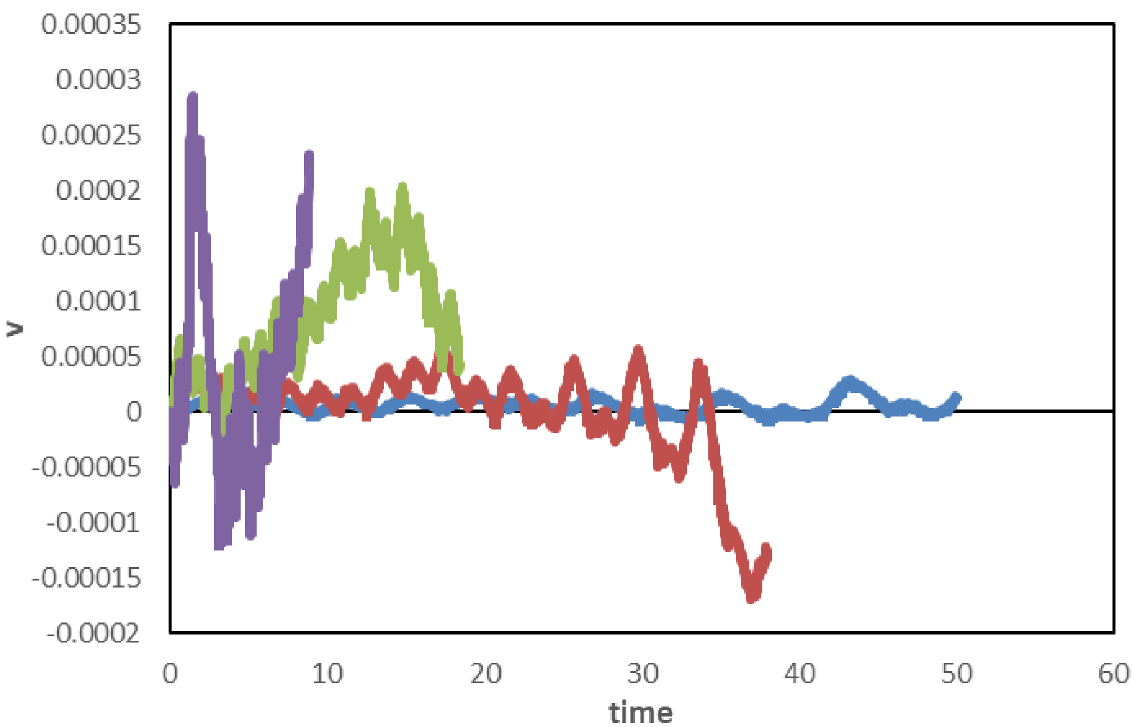


2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
x
0
10
20
30
40
50
60
time
ω=π/4
ω=π/2
ω=π
ω=2π



0
-0.05
-0.1
-0.15
-0.2
-0.25
-0.3
u
0
10
20
30
40
50
60
time


0.5018
0.5016
0.5014
0.5012
0.501
0.5008
0.5006
0.5004
0.5002
0.5
0.4998
y
0
10
20
30
40
50
60
time


0.00035
0.0003
0.00025
0.0002
0.00015
0.0001
0.00005
0
-0.00005
-0.0001
-0.00015
-0.0002
v
0
10
20
30
40
50
60
time

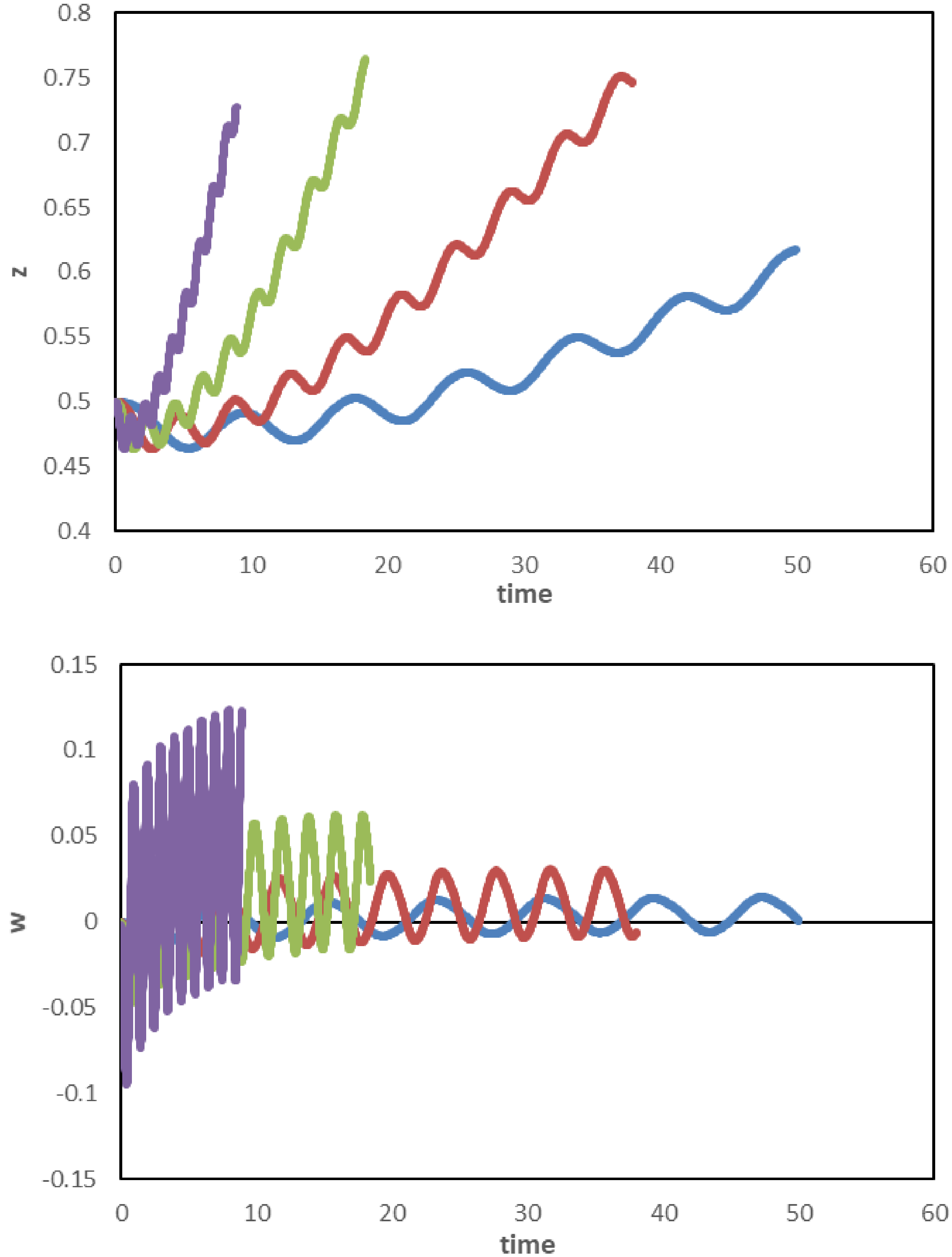
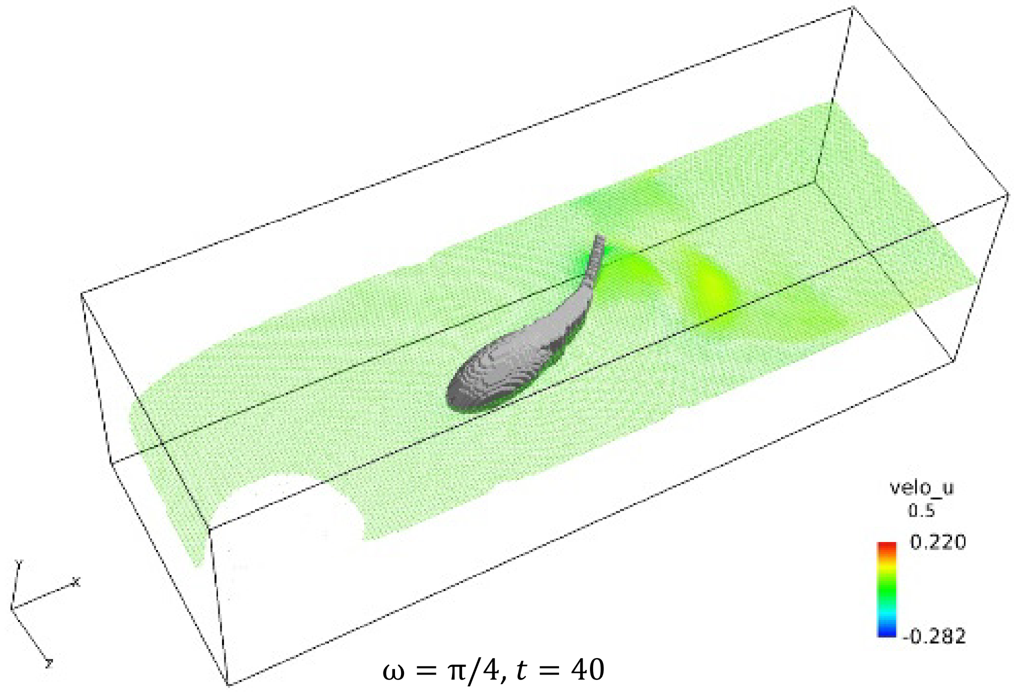
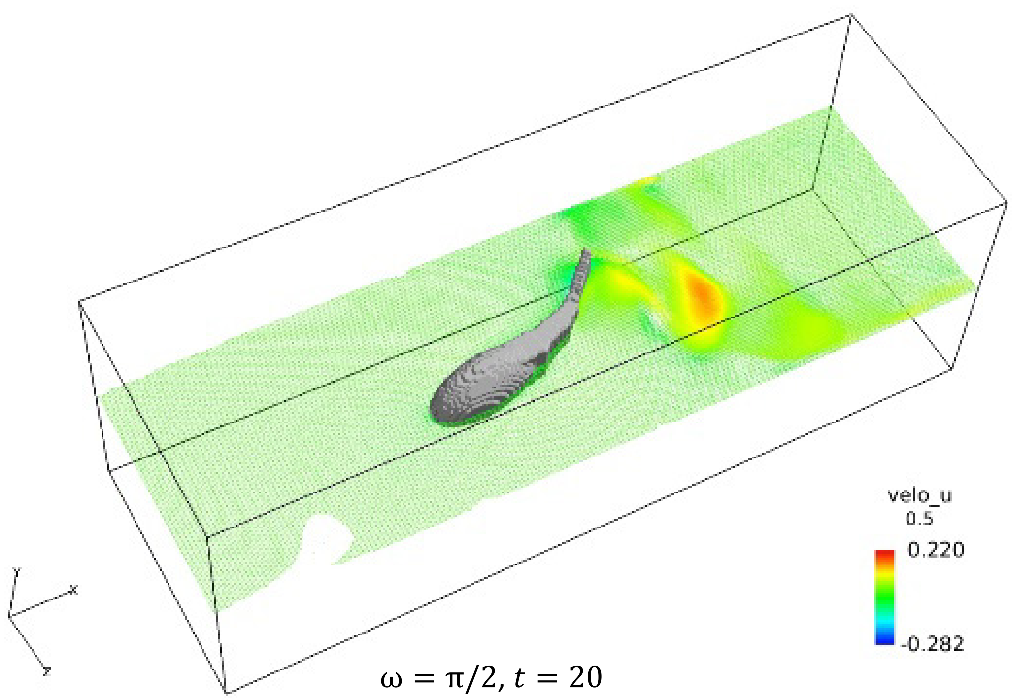


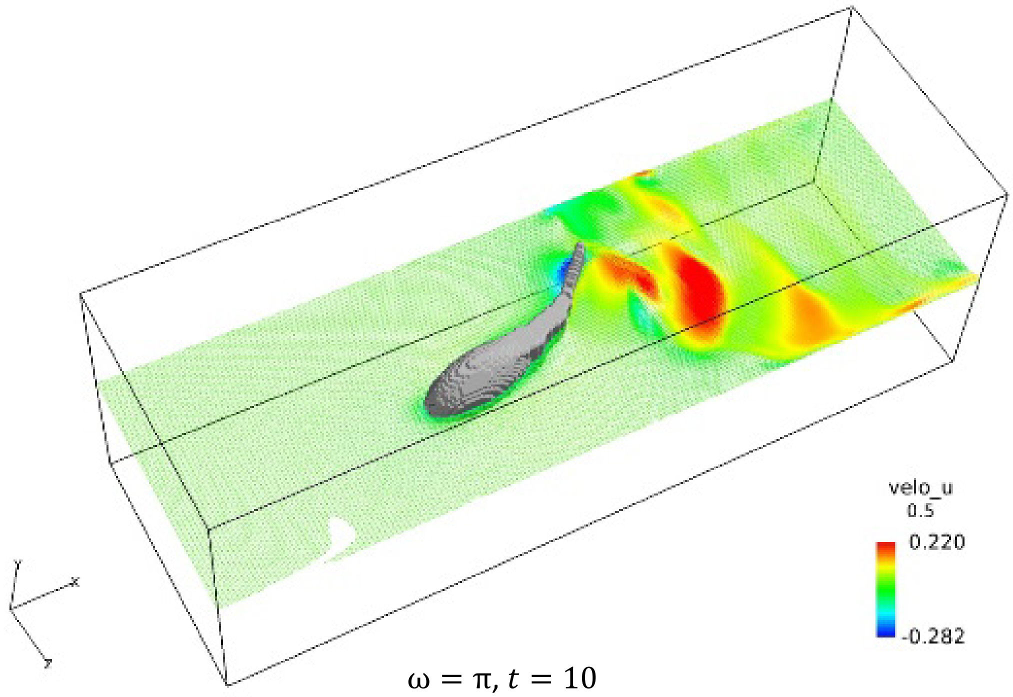
Figure 5.20: Displacement and velocity component of center of mass against time along the three directions for cases with different angular velocities ($\phi = \pi/6$ and A=0.25).



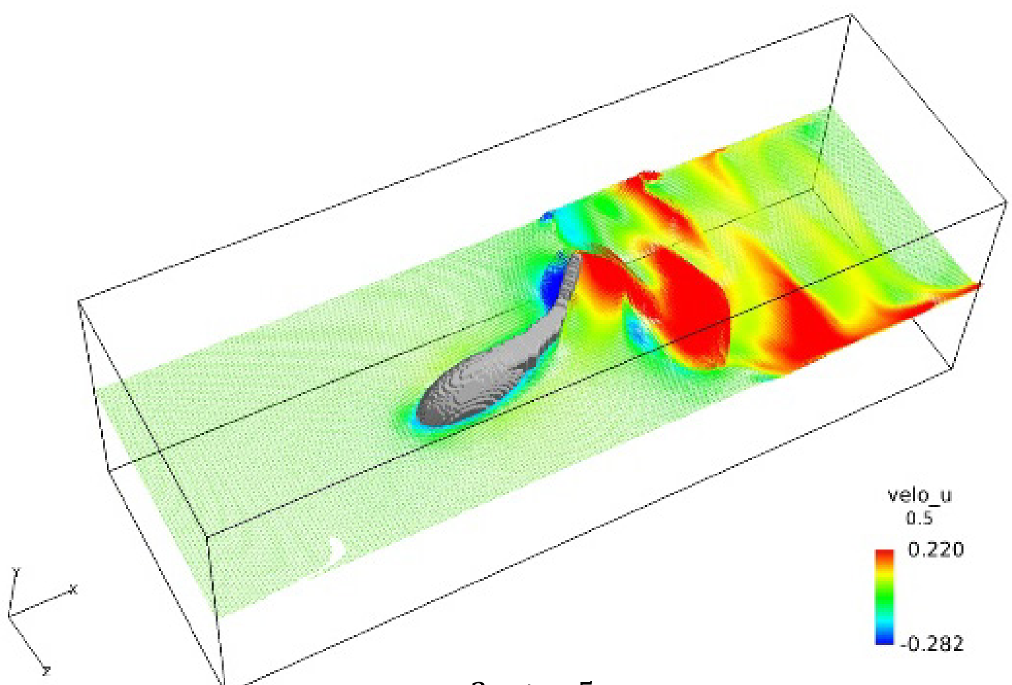
$\omega = \pi/4, t = 40$



$\omega = \pi/2, t = 20$

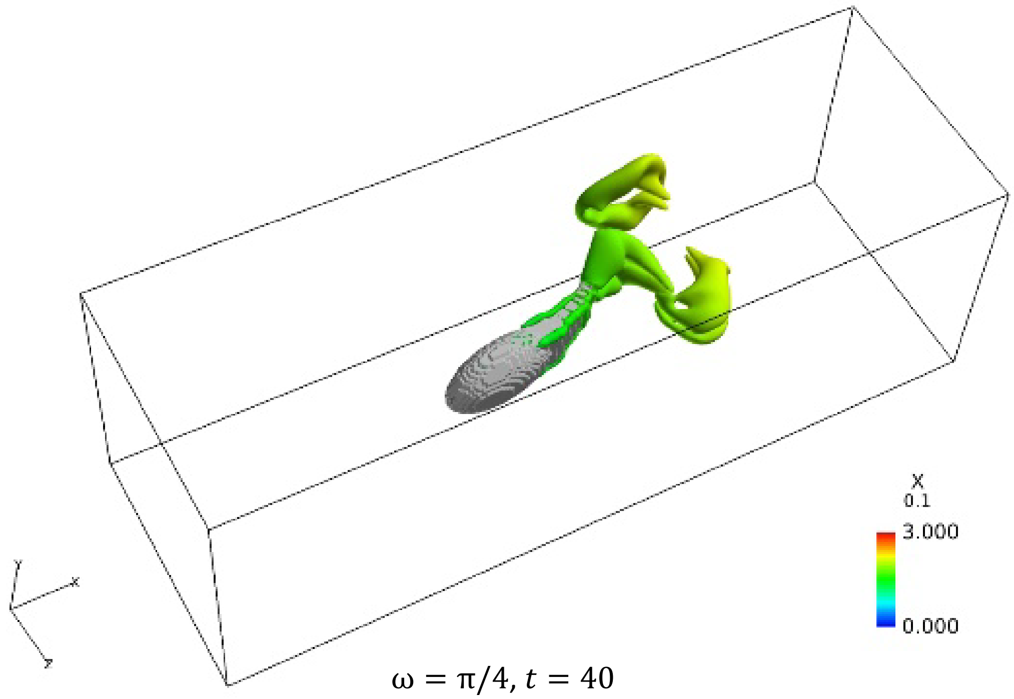


$\omega = \pi, t = 10$

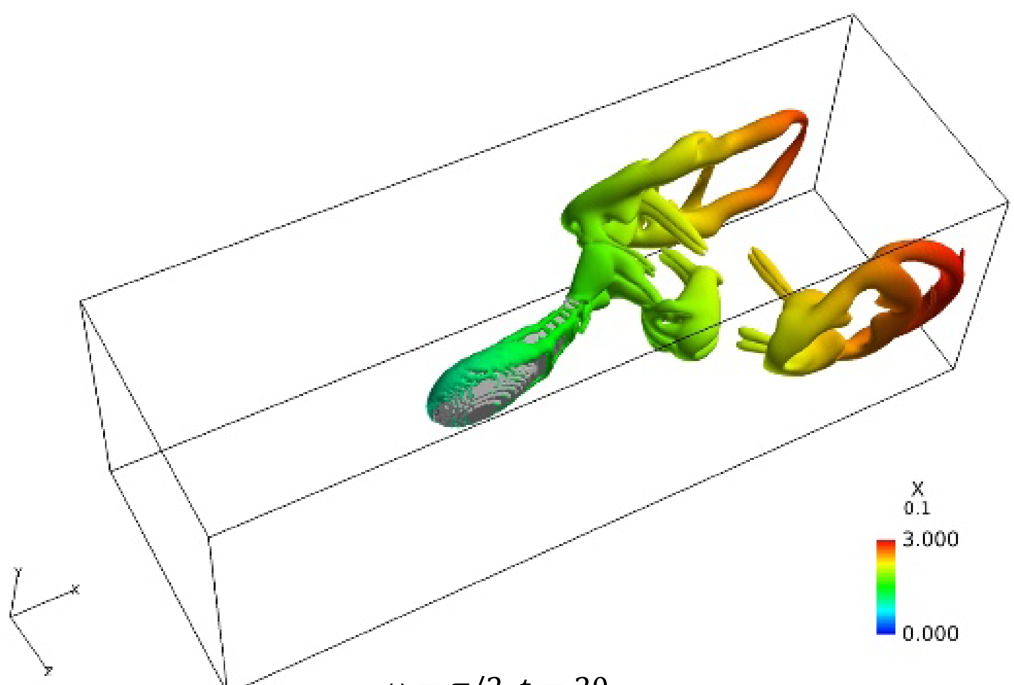


$\omega = 2\pi, t = 5$

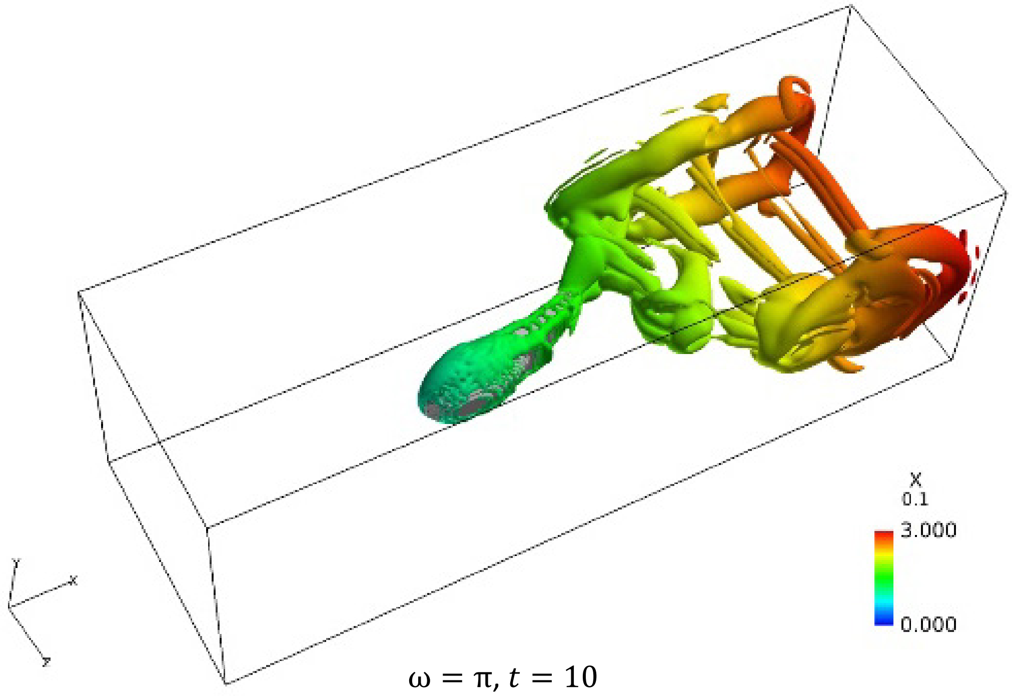
Figure 5.21: Velocity vectors and color contours for cases with different angular velocities ($\phi = \pi/6$ and A=0.25).



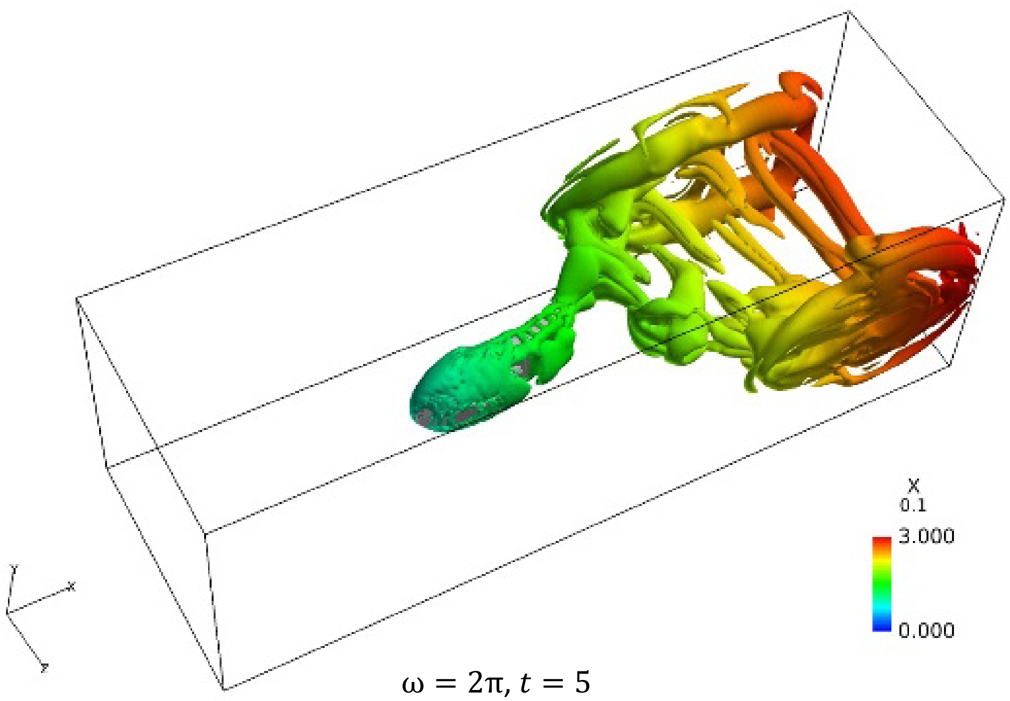
$\omega = \pi/4$, $t = 40$



$\omega = \pi/2$, $t = 20$



$\omega = \pi, t = 10$



$\omega = 2\pi, t = 5$

Figure 5.22: Vortex structures for cases with different angular velocities ($\phi = \pi/6$ and A=0.25).

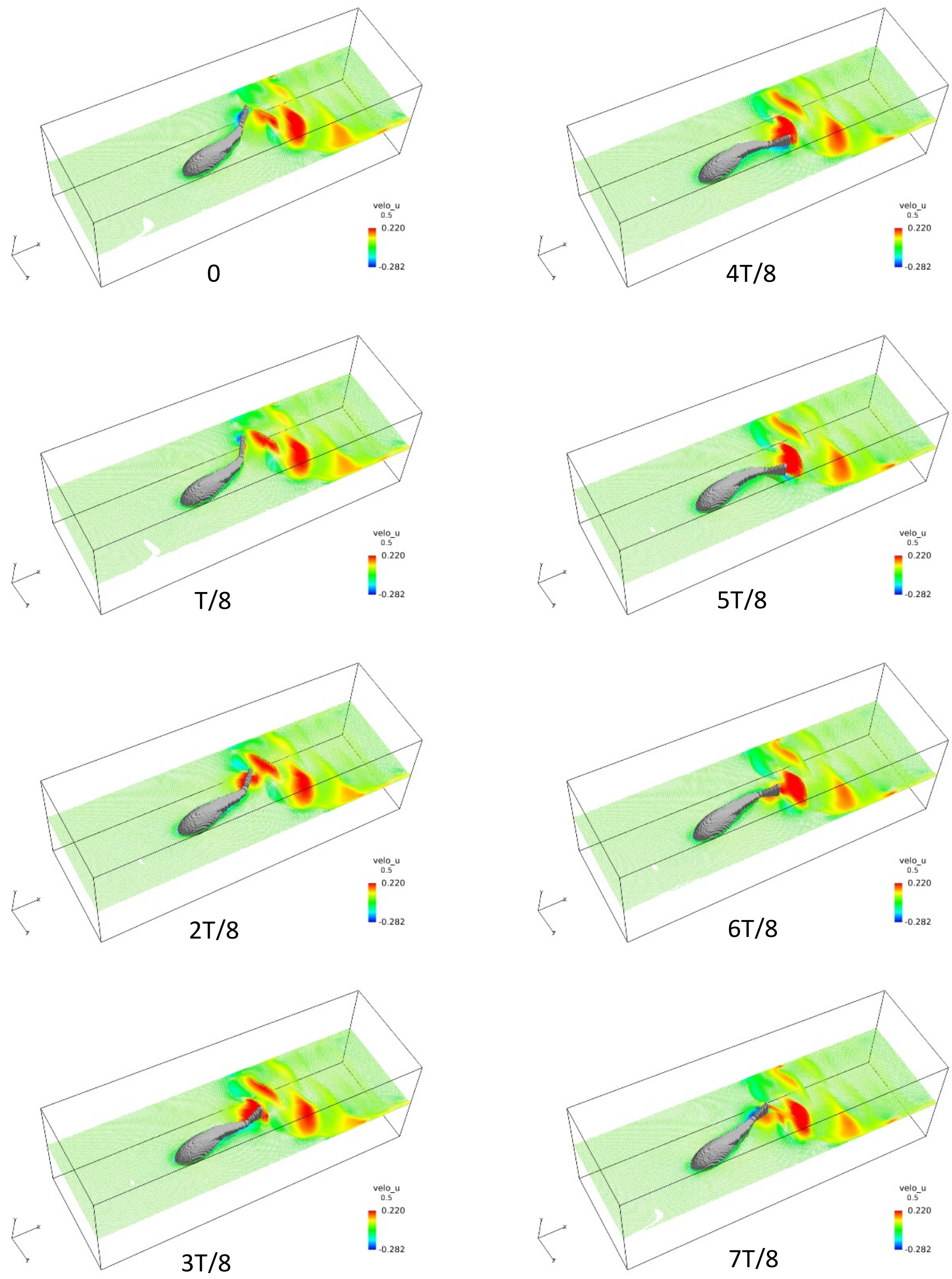


Figure 5.23: Velocity vectors and color contours of velocity in one cycle for the case of $\omega = \pi$ with $\phi = \pi/6$ and A=0.25.

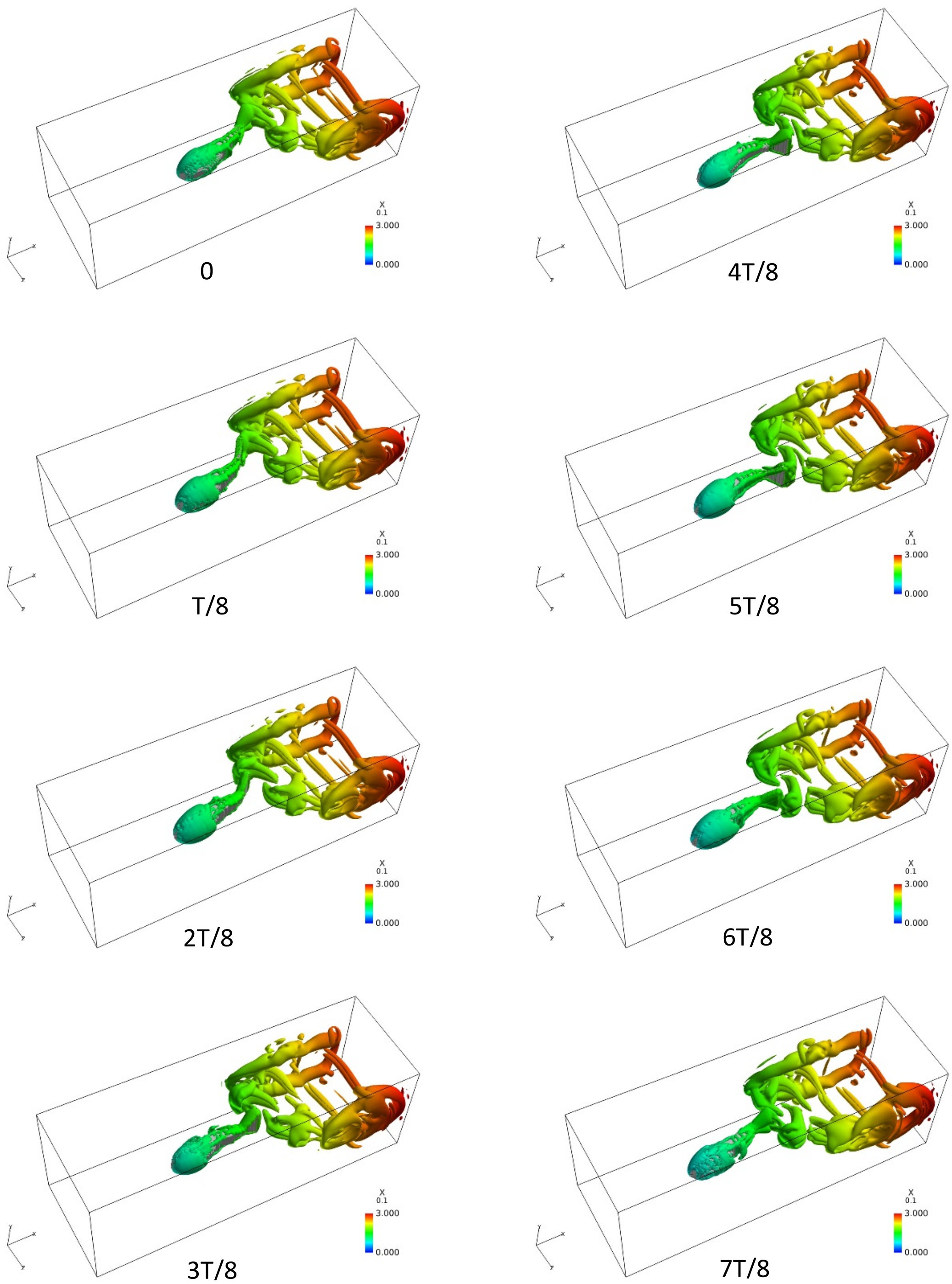


Figure 5.24: Vortex structures visualized by the second invariant of the velocity-gradient tensor in one cycle for $\omega = \pi$ with $\phi = \pi/6$ and A=0.25.

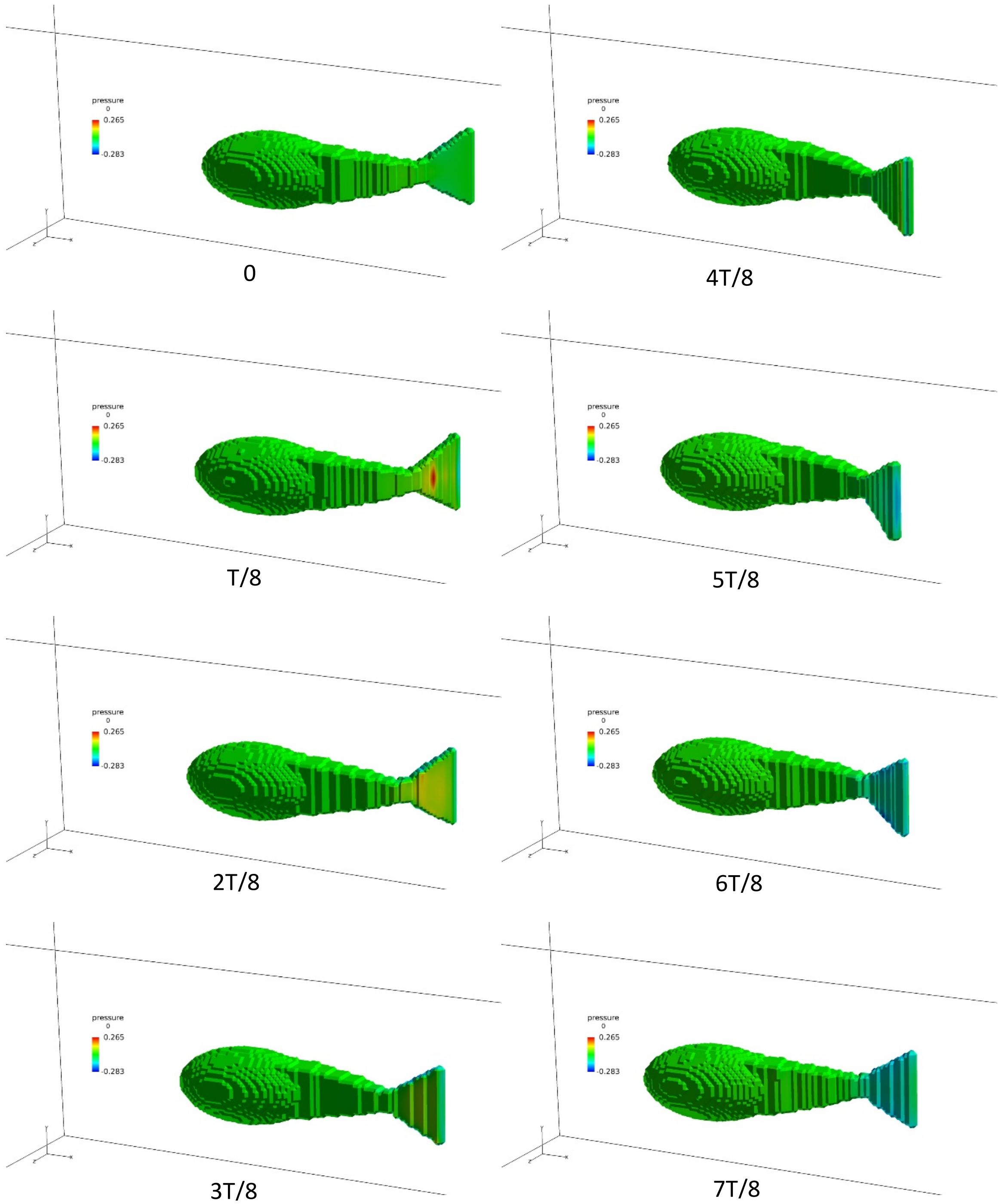
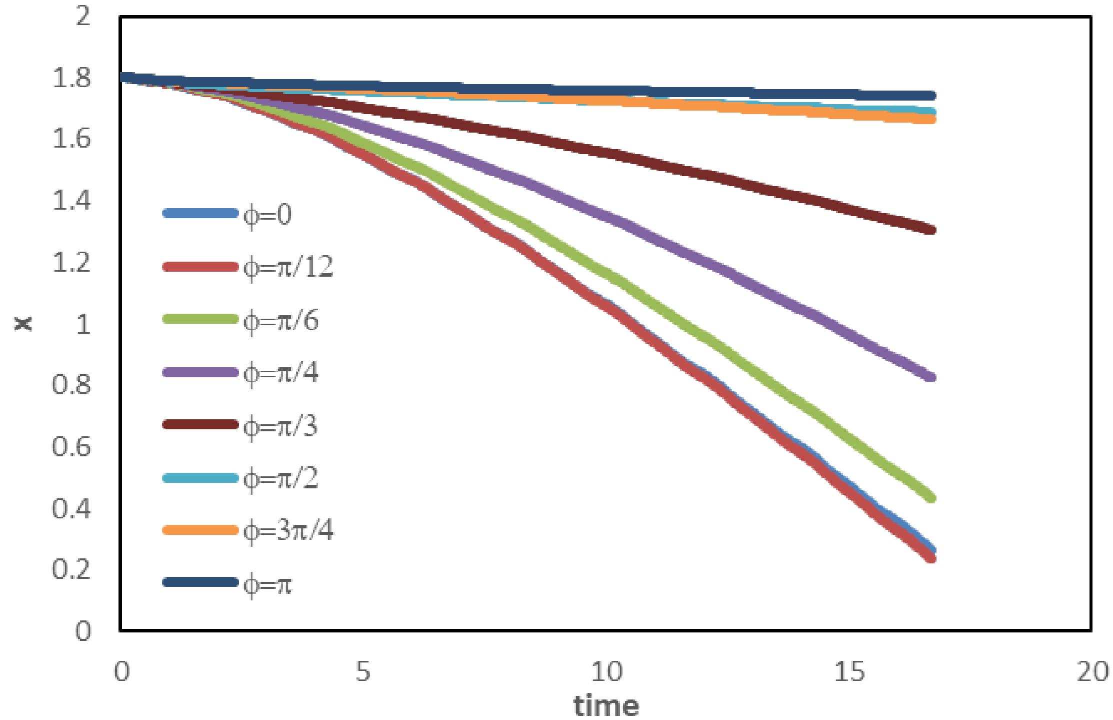
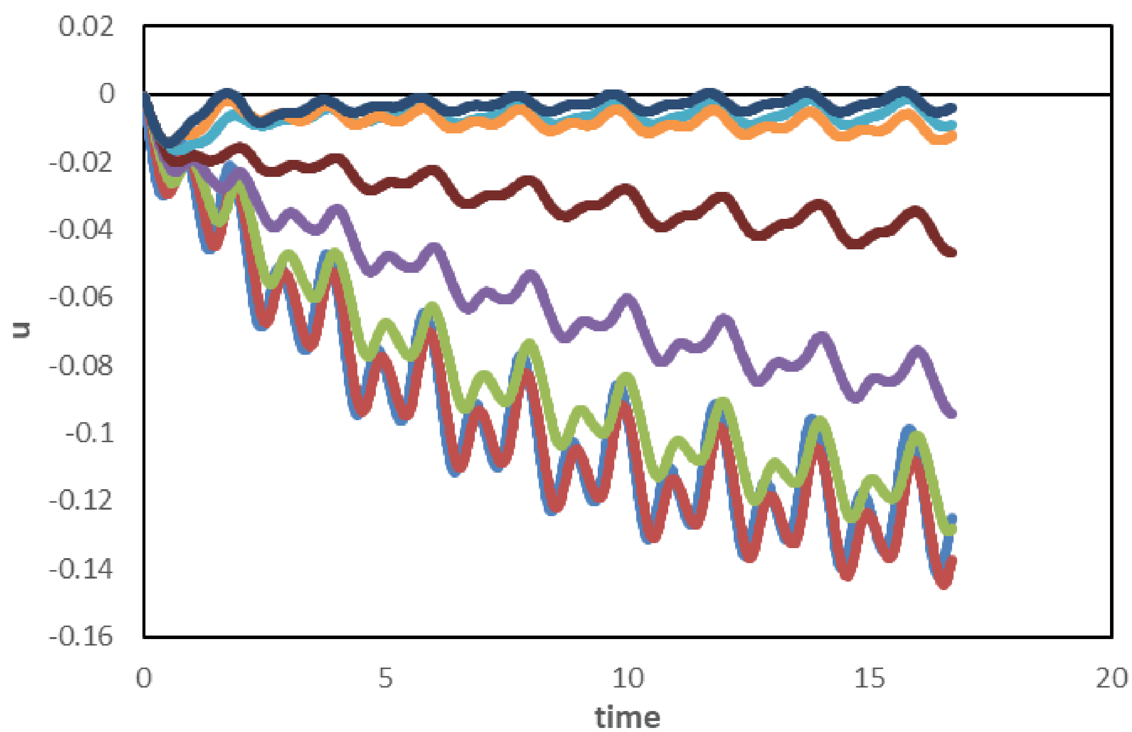
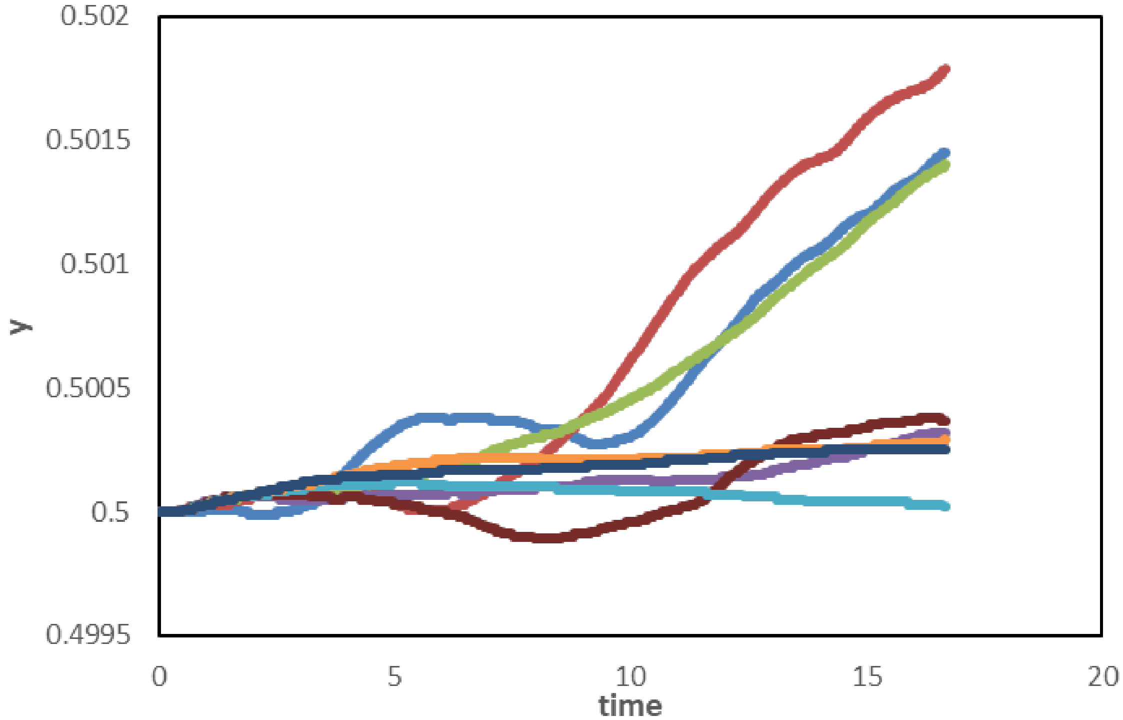
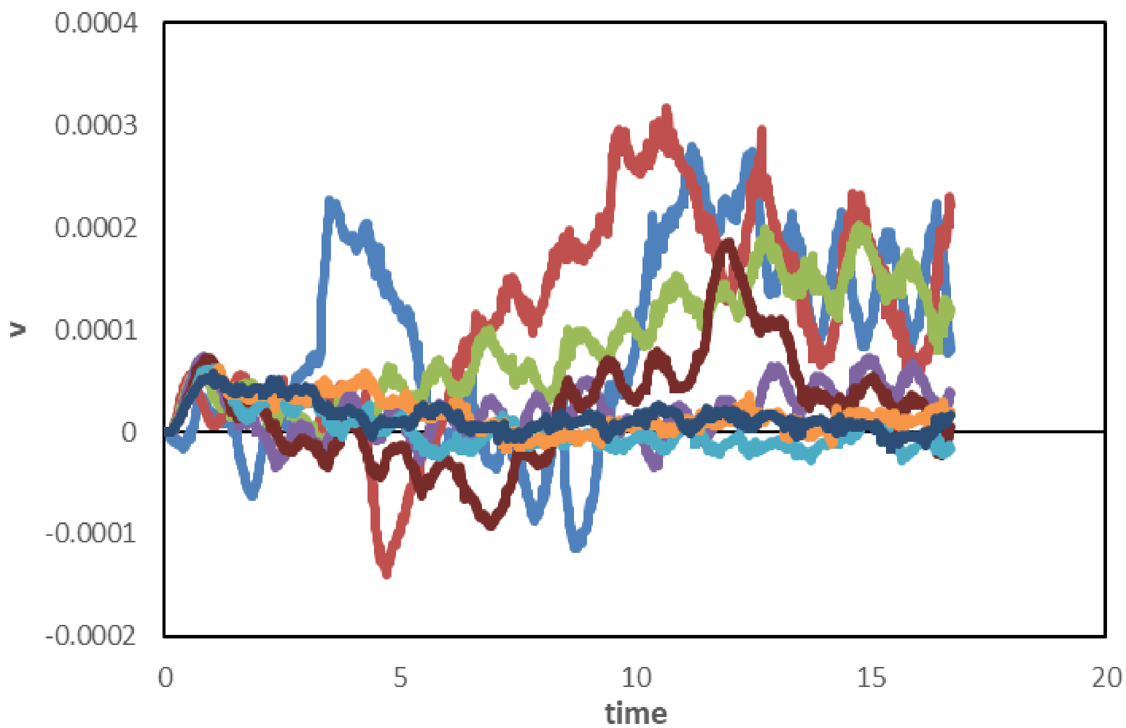


Figure 5.25: Pressure distributions on the body surface in one cycle for case of $\omega = \pi$ with $\phi = \pi/6$ and A=0.25.


2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
x
ϕ=0
ϕ=π/12
ϕ=π/6
ϕ=π/4
ϕ=π/3
ϕ=π/2
ϕ=3π/4
ϕ=π
0
5
10
15
20
time


0.02
0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
-0.14
-0.16
u
0
5
10
15
20
time


0.502
0.5015
0.501
0.5005
0.5
0.4995
y
0
5
10
15
20
time


0.0004
0.0003
0.0002
0.0001
0
-0.0001
-0.0002
v
0
5
10
15
20
time

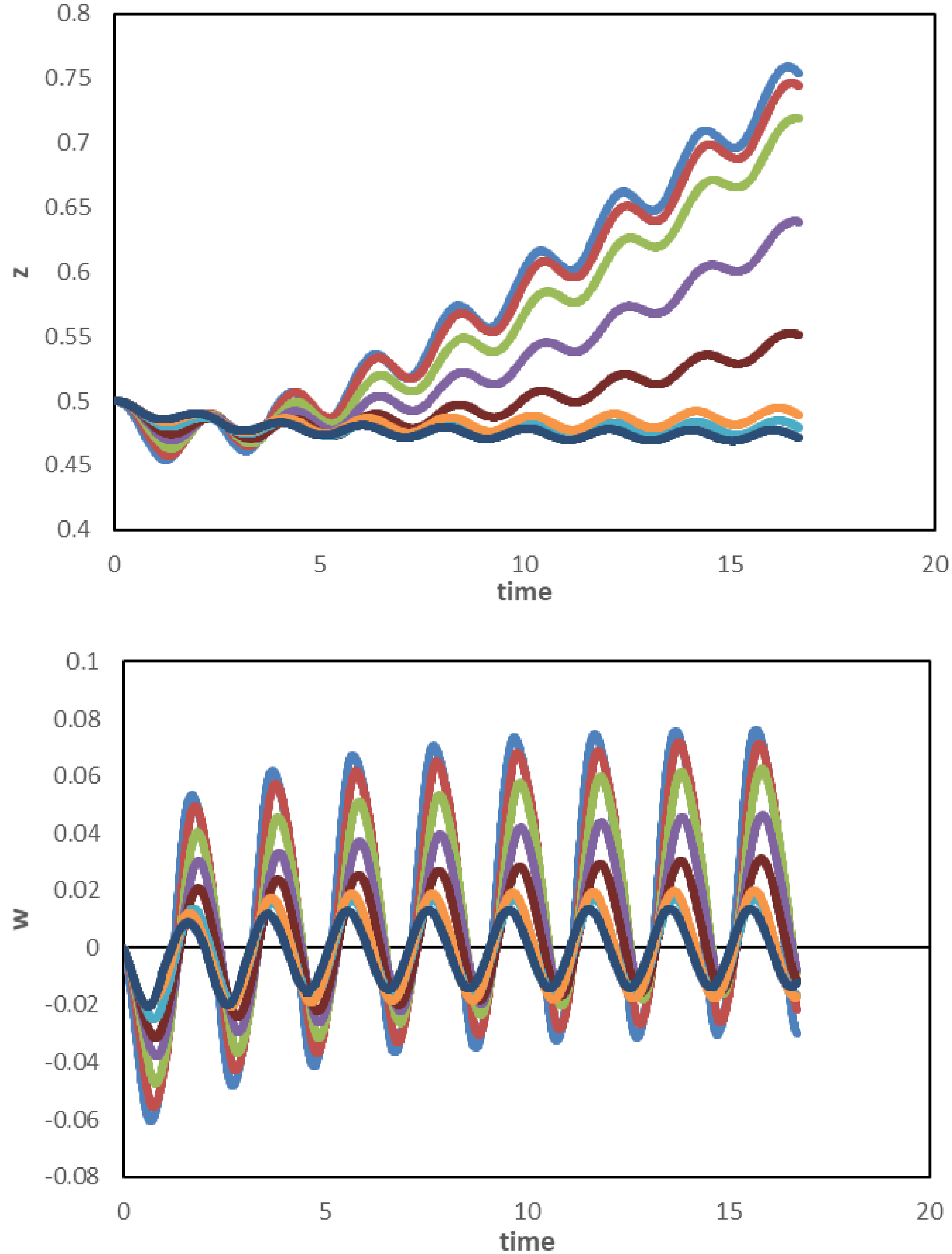
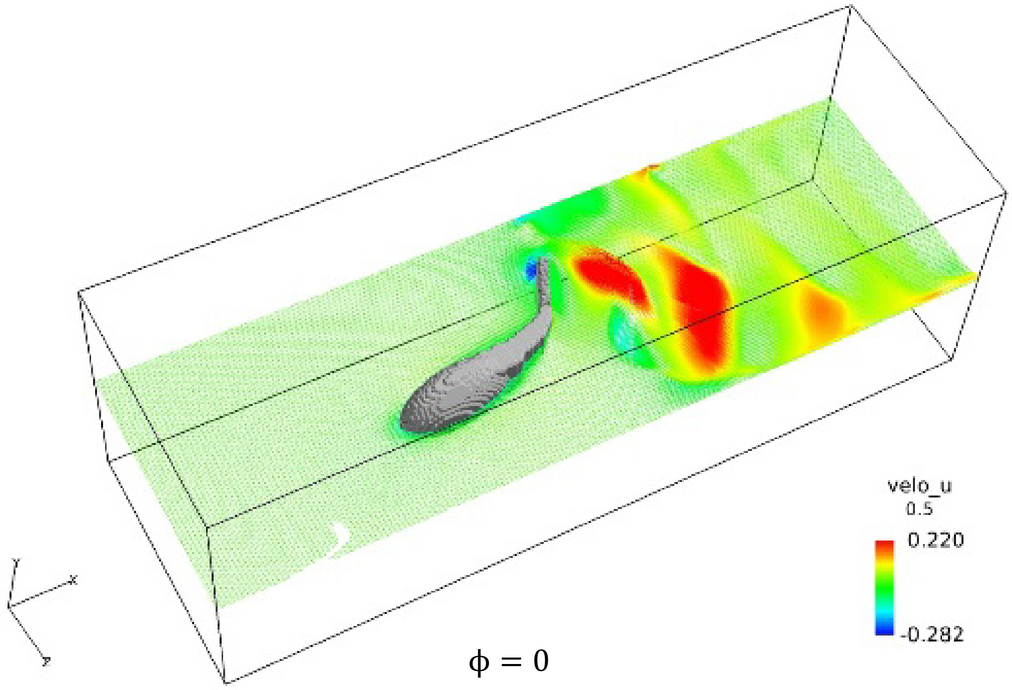
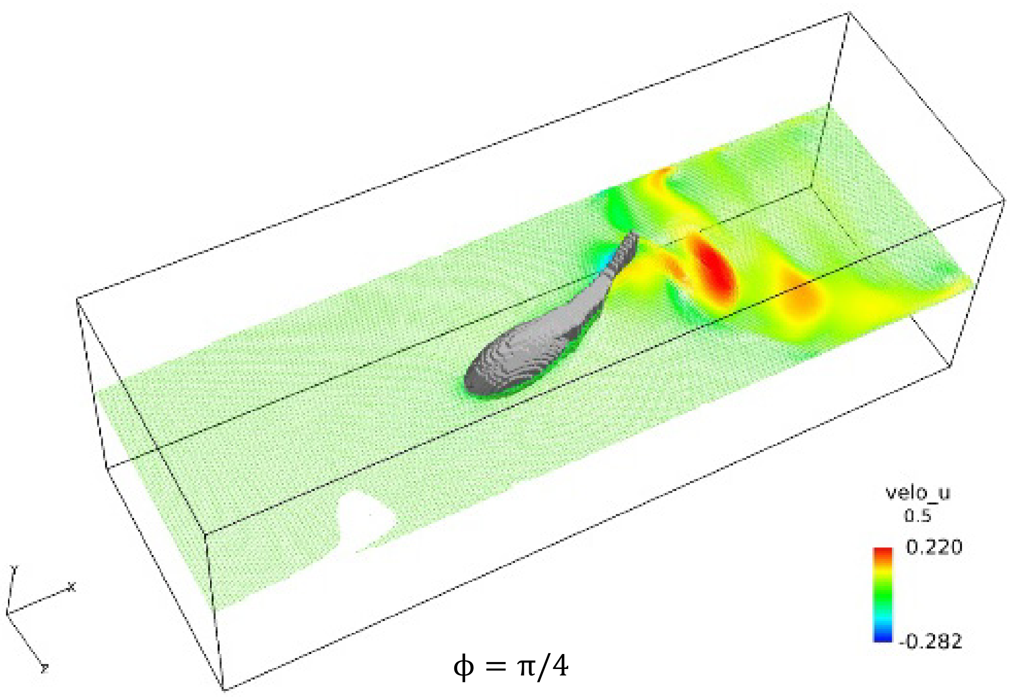


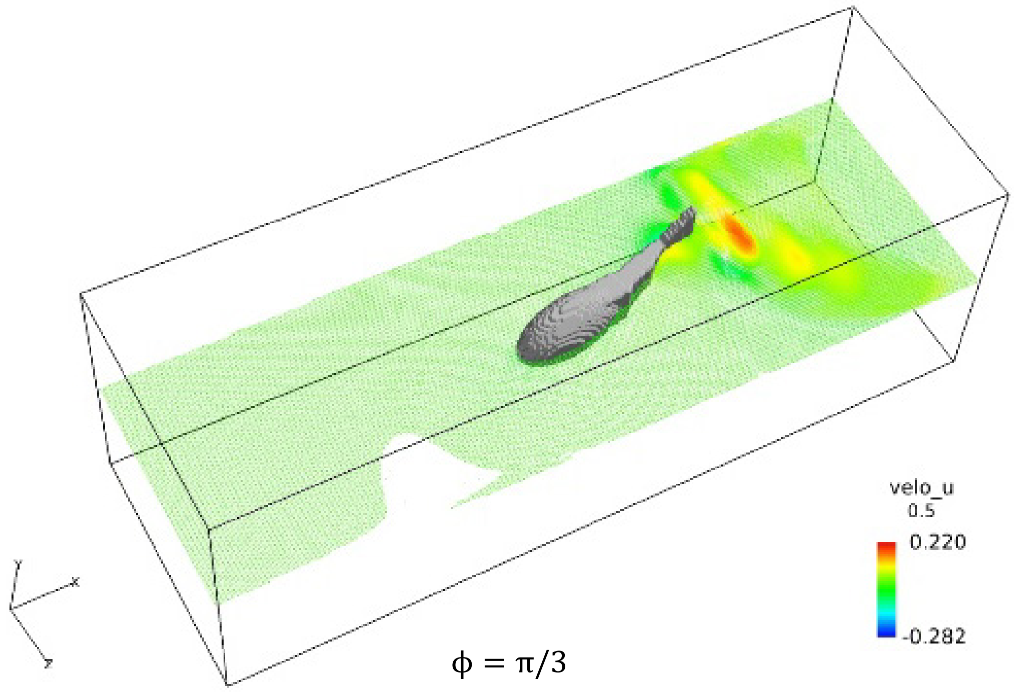
Figure 5.26: Profiles of the displacement and velocity component against time along three directions for phase differences ($\omega = \pi$ and A=0.25).



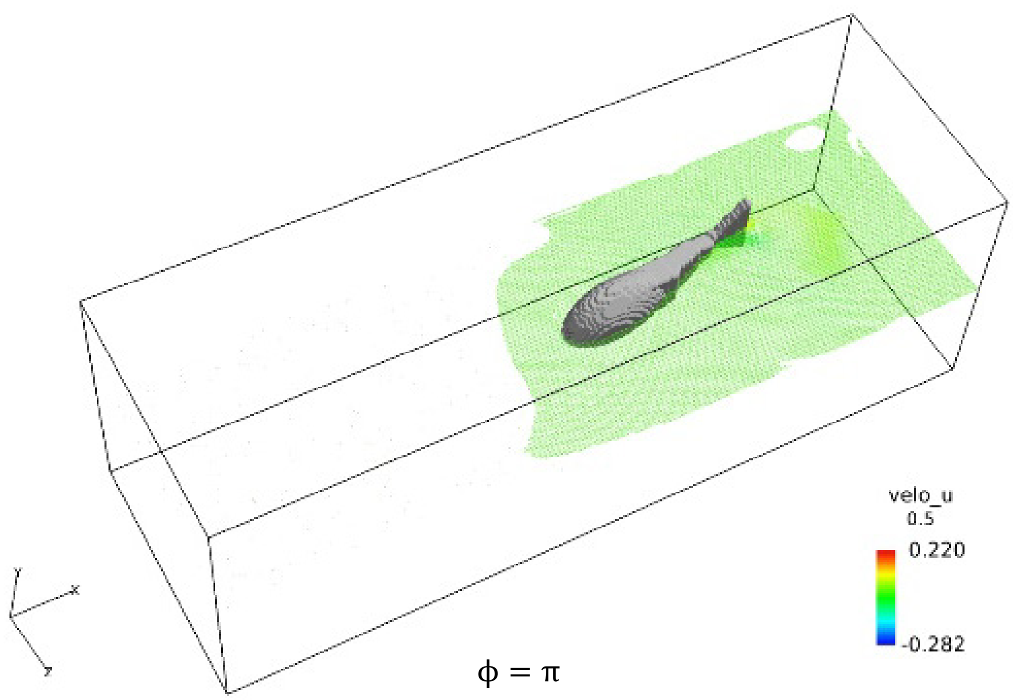
$\phi = 0$



$\phi = \pi/4$

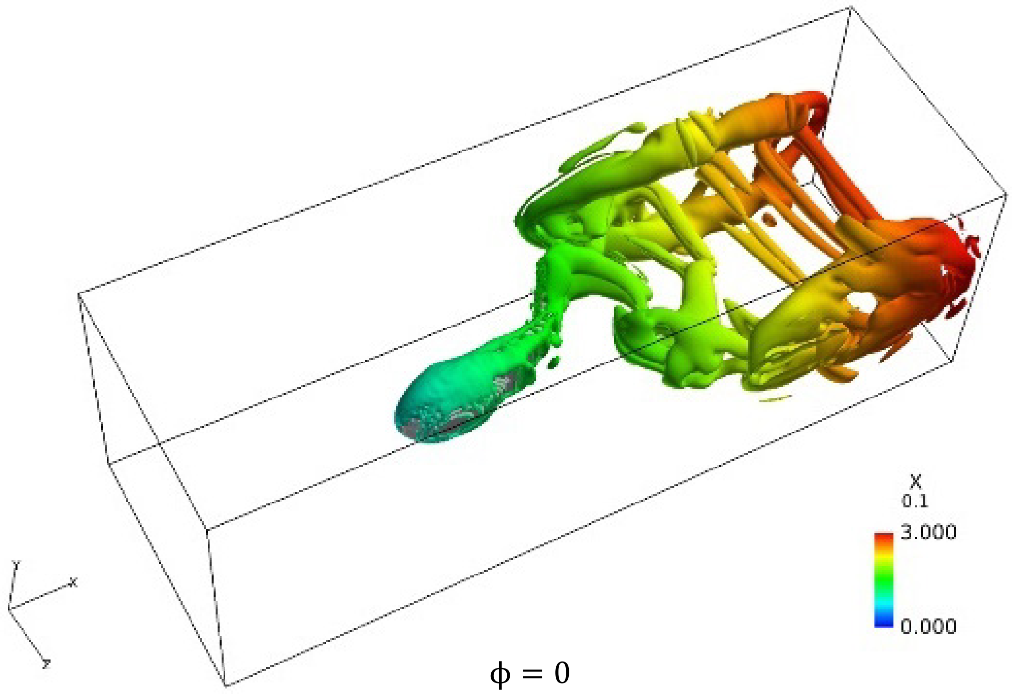


$\phi = \pi/3$

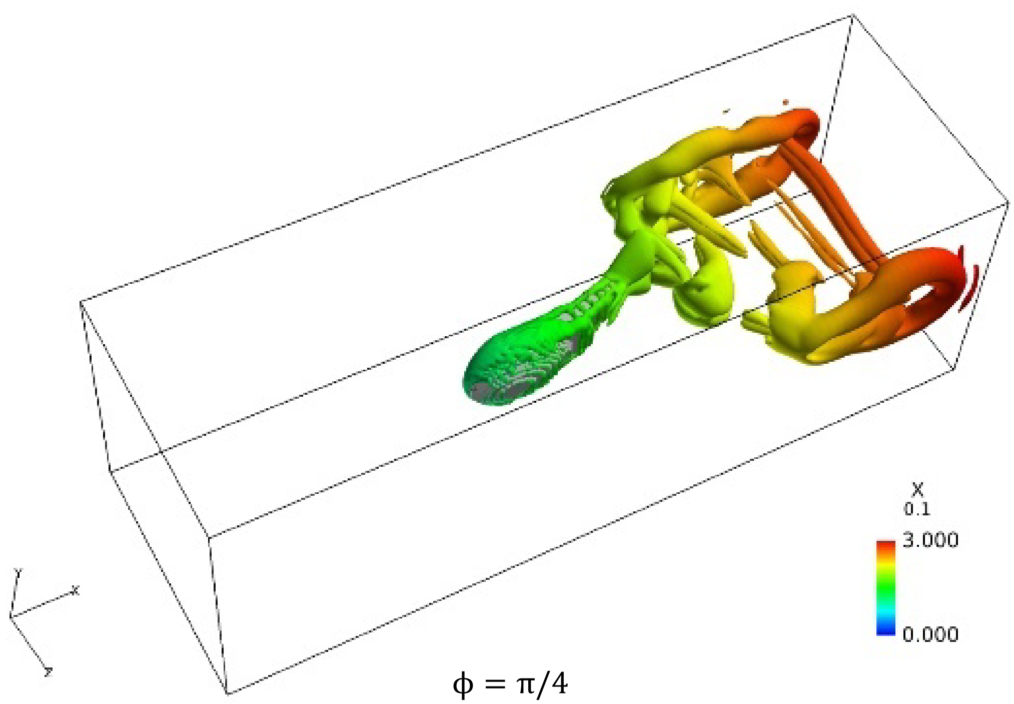


$\phi = \pi$

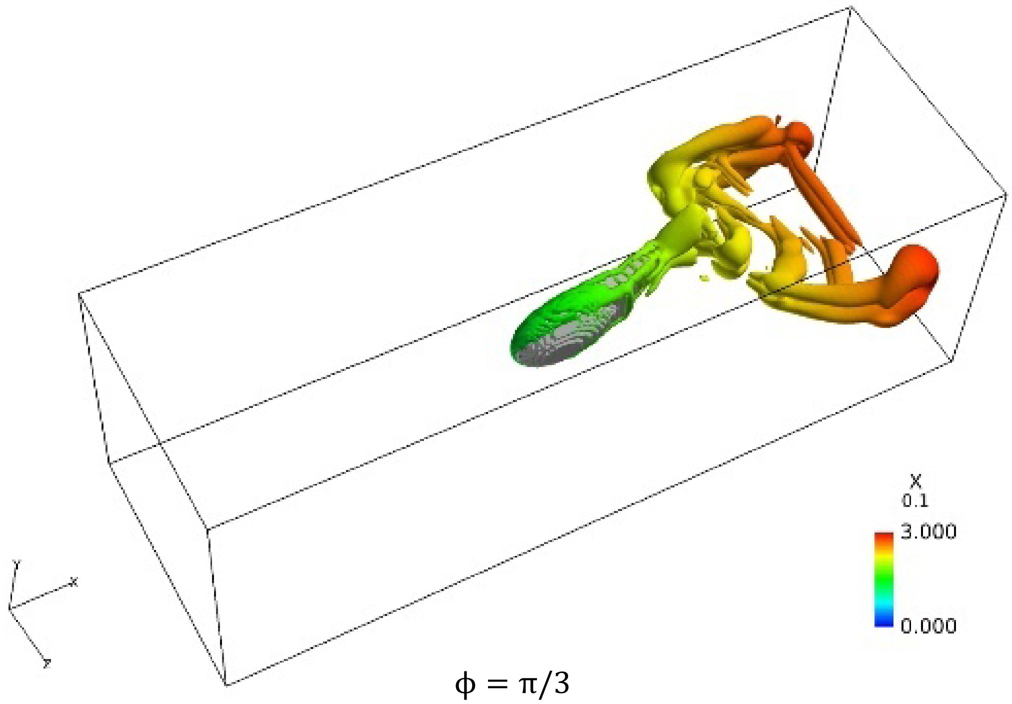
Figure 5.27: Velocity vectors and color contours for cases with different phase difference ($\omega = \pi$ and A=0.25, $t = 10$).



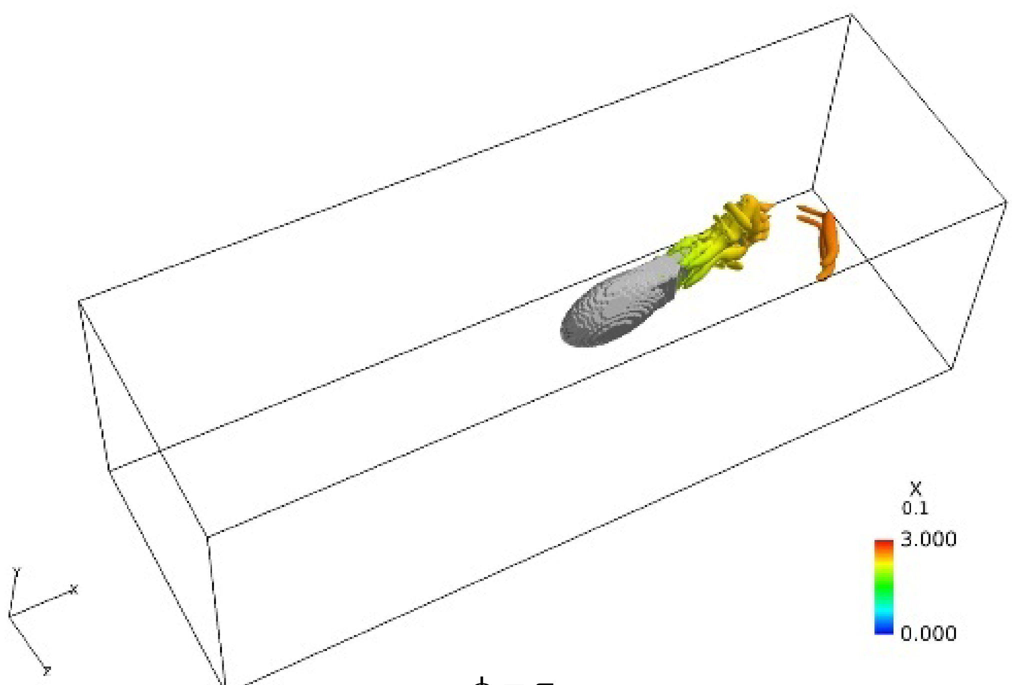
$\phi = 0$



$\phi = \pi/4$

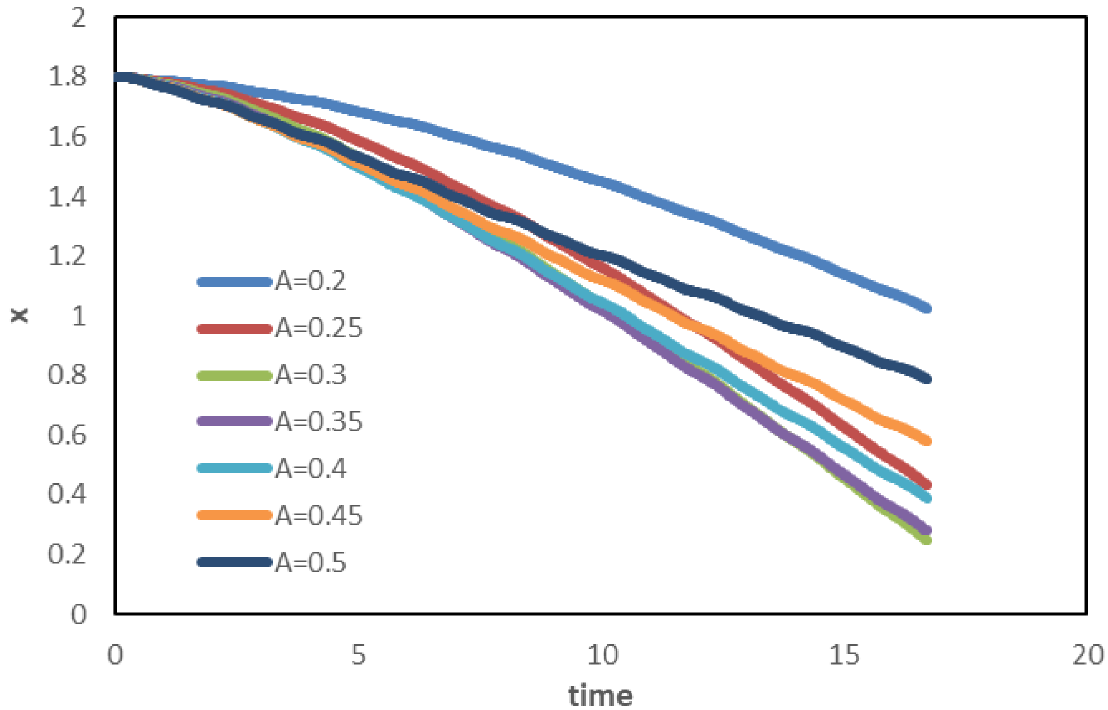


$\phi = \pi/3$

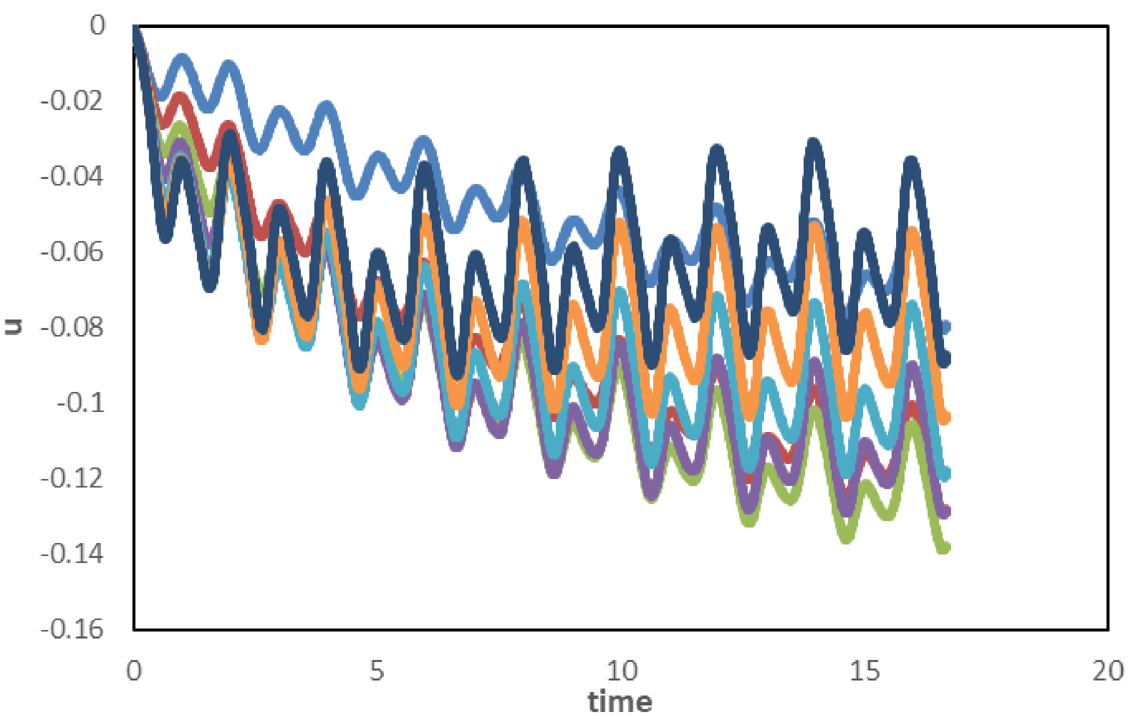


$\phi = \pi$

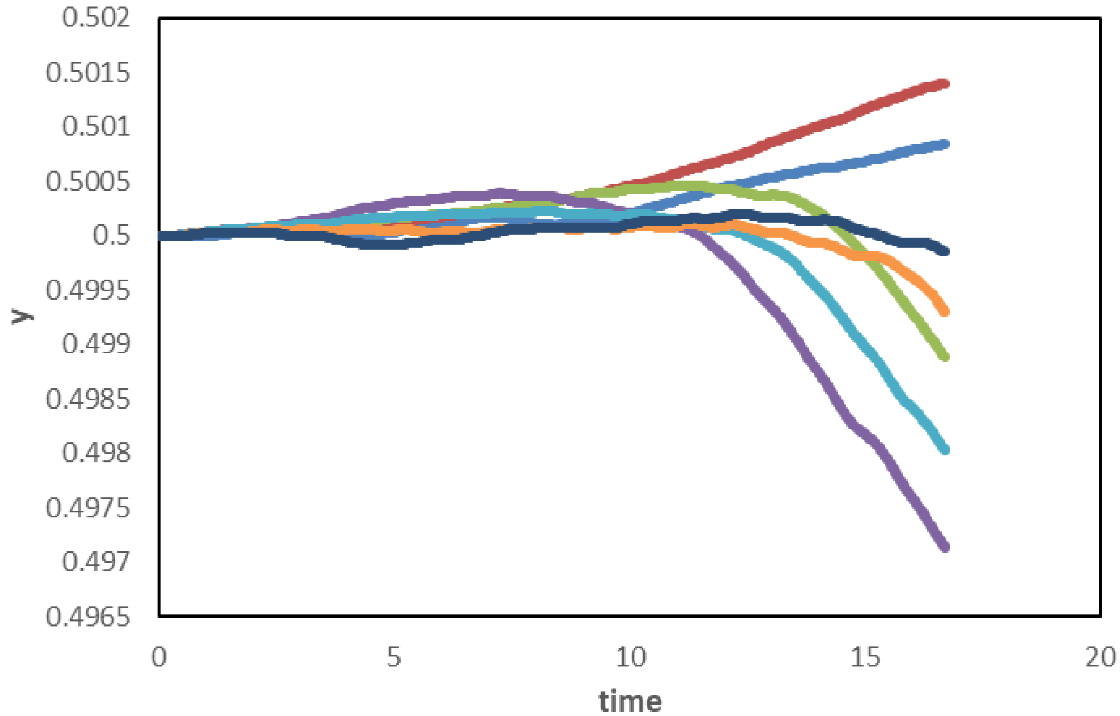
Figure 5.28: Vortex structures for $\omega = \pi$ and A=0.25, $t = 10$.

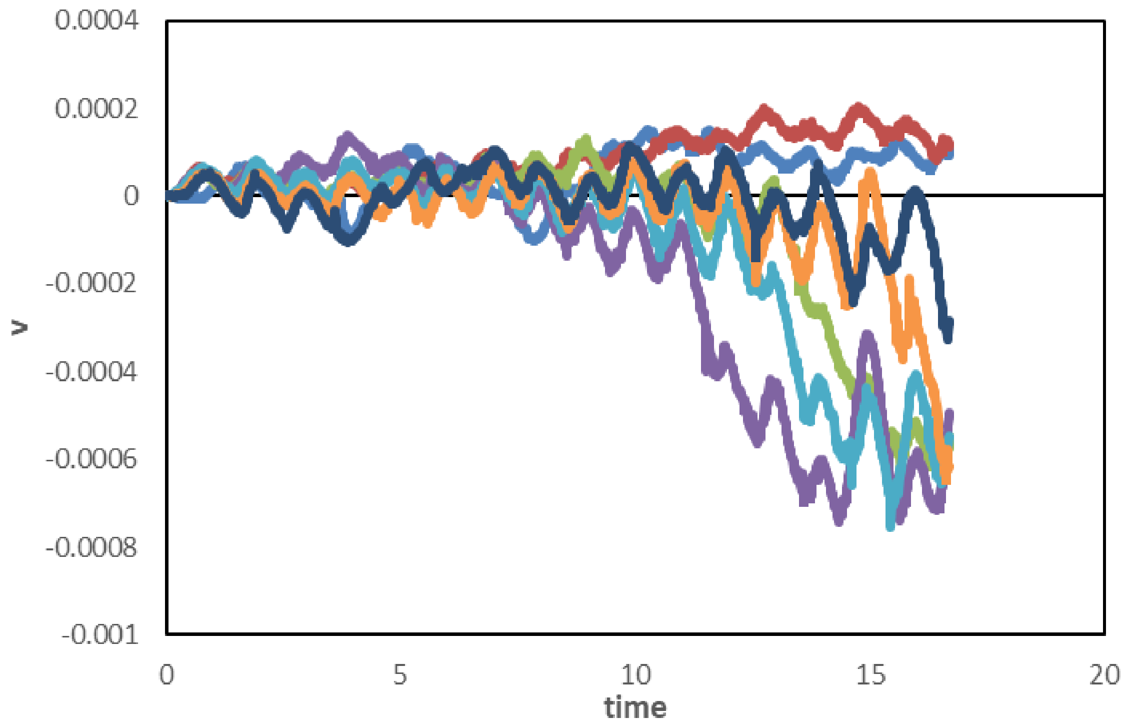


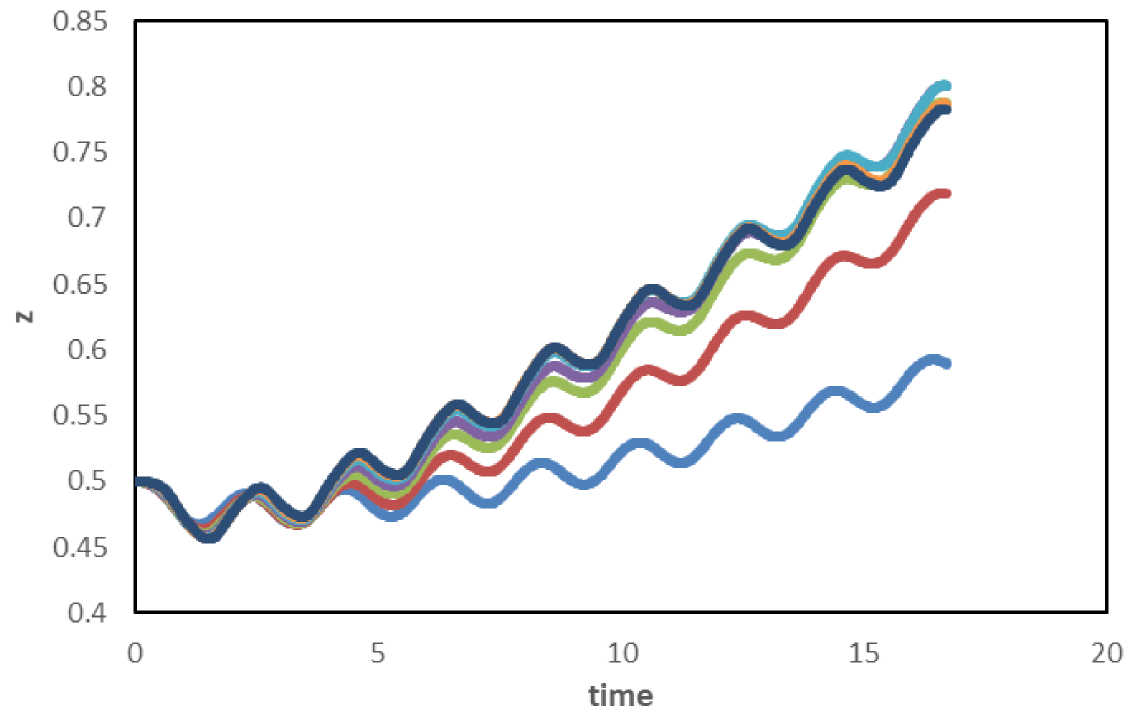
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
x
A=0.2
A=0.25
A=0.3
A=0.35
A=0.4
A=0.45
A=0.5
0
5
10
15
20
time



0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
-0.14
-0.16
u
0
5
10
15
20
time


0.502
0.5015
0.501
0.5005
0.5
0.4995
0.499
0.4985
0.498
0.4975
0.497
0.4965
y
0
5
10
15
20
time


0.0004
0.0002
0
-0.0002
-0.0004
-0.0006
-0.0008
-0.001
v
0
5
10
15
20
time



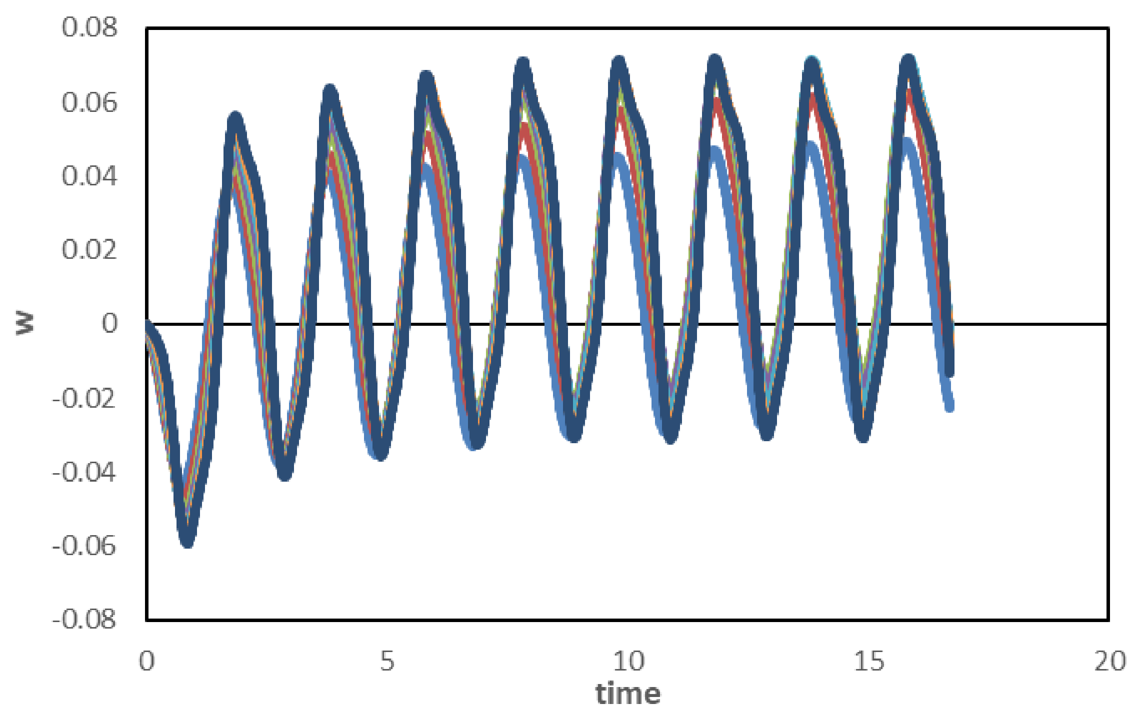
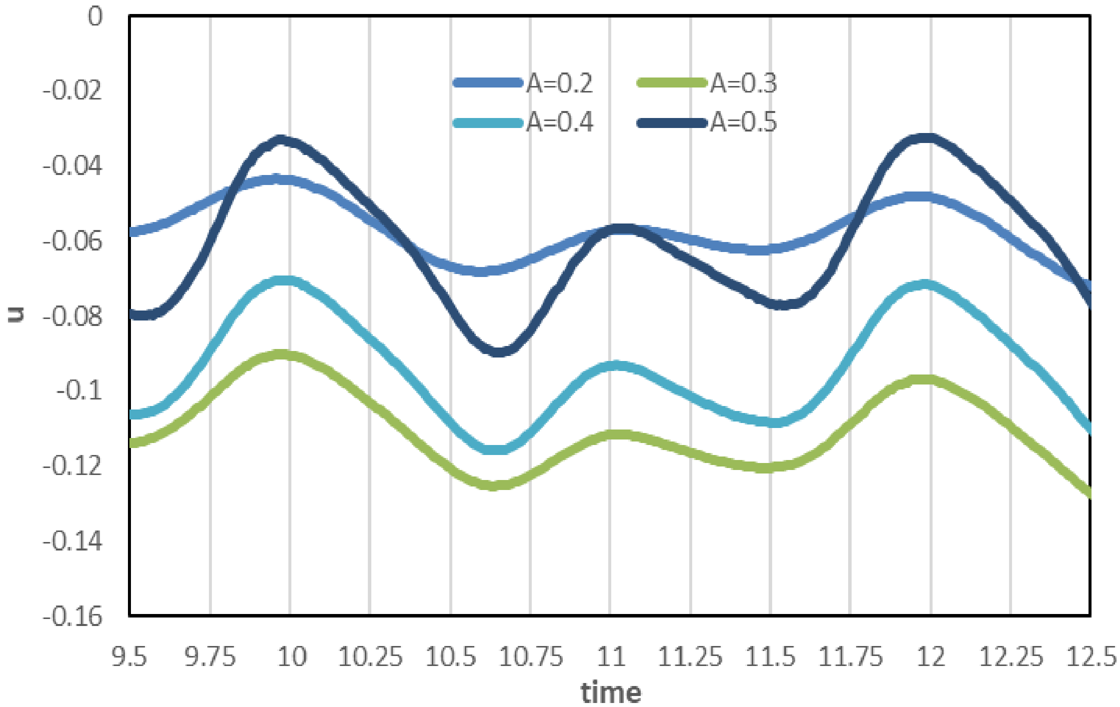


Figure 5.29: Displacements and velocity components against time along three directions for the cases with different body amplitudes ($\omega = \pi$ and $\phi = \pi/6$).

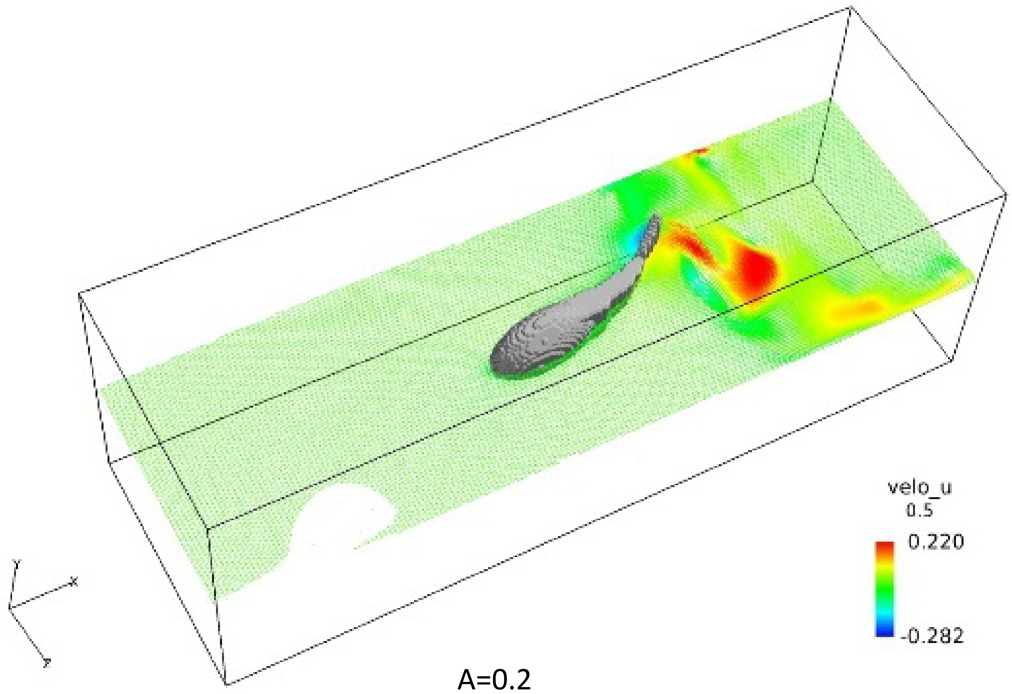


(a)

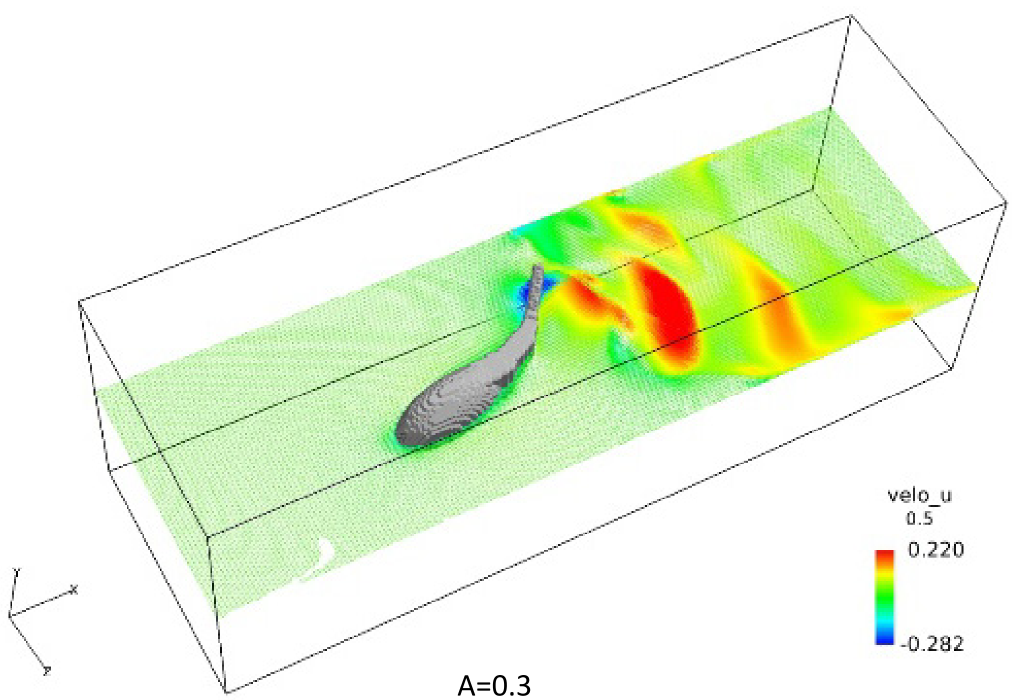
10 10.25 10.5 10.75 11 11.25 11.5 11.75 12

(b)

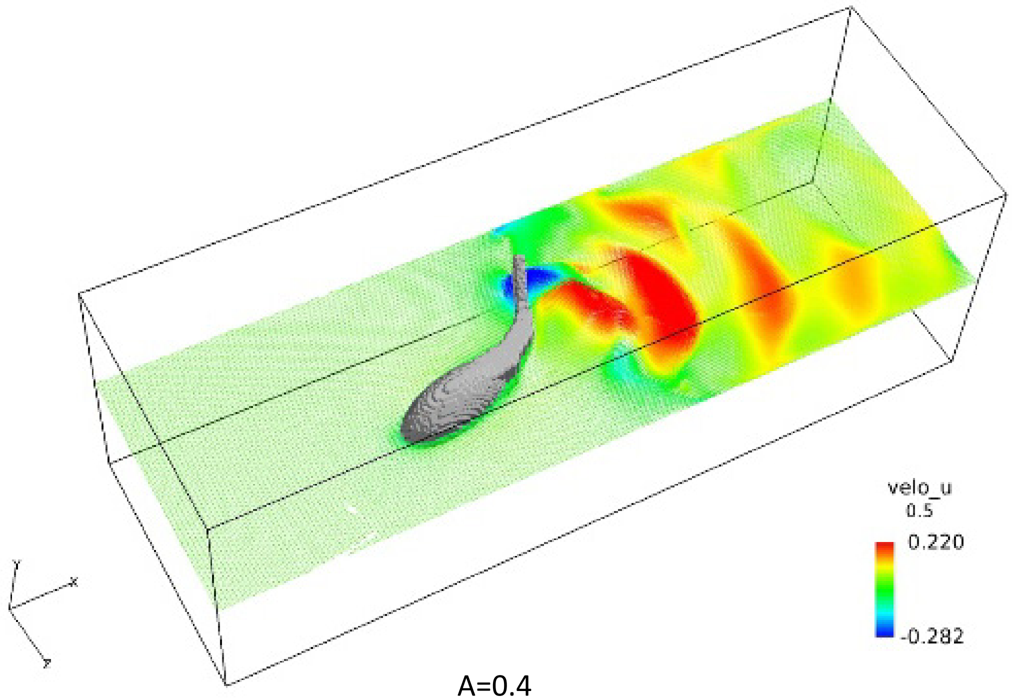
Figure 5.30: (a) Enlarged profiles of the longitudinal velocities for the cases with different amplitudes ($\omega = \pi$ and $\phi = \pi/6$); (b) Postures of the fish model in one tail-beat cycle for the case of A=0.3 ($\omega = \pi$ and $\phi = \pi/6$).



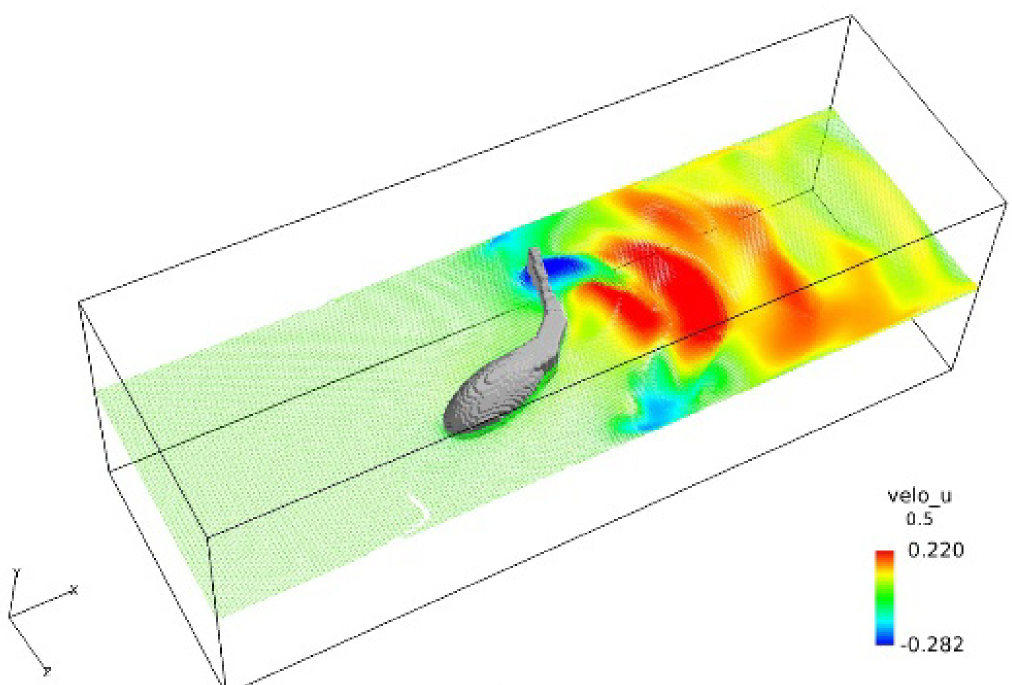
A=0.2



A=0.3

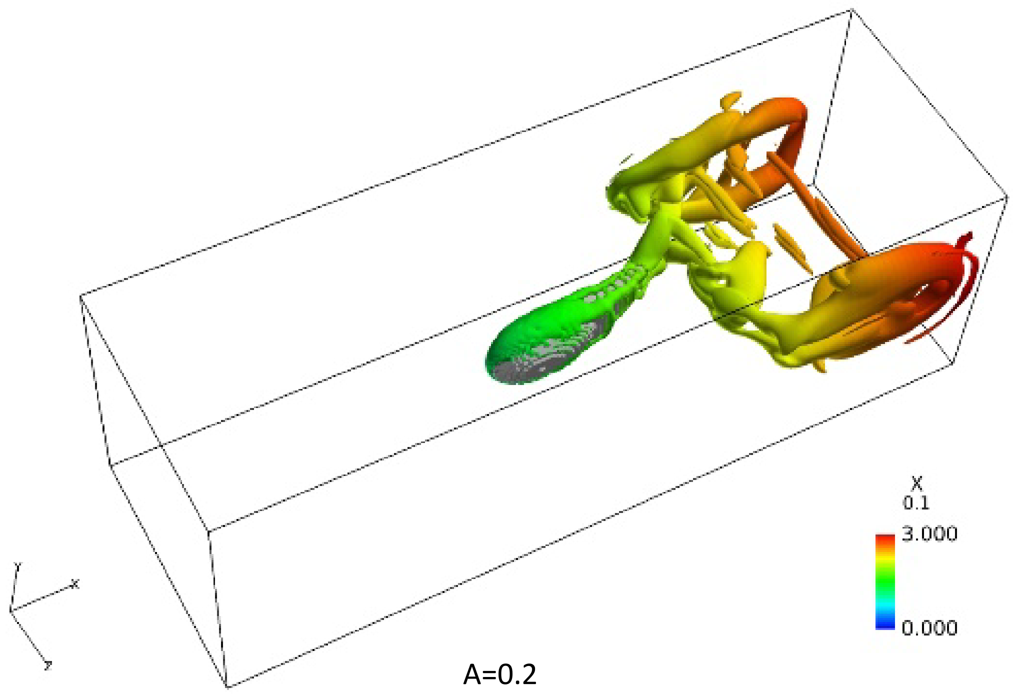


A=0.4

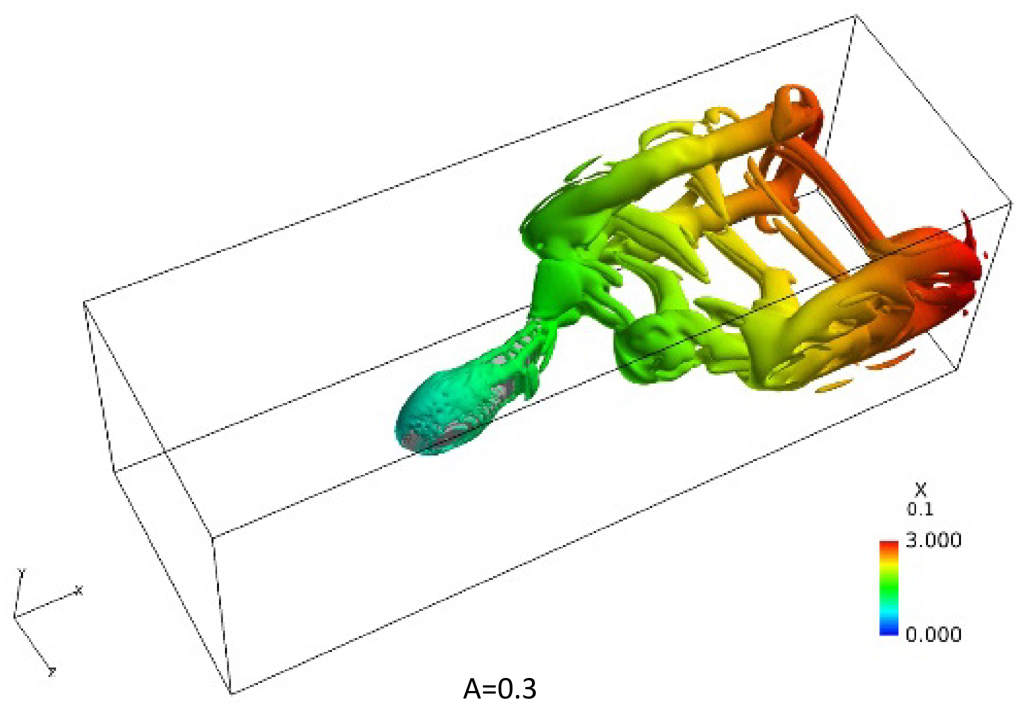


A=0.5

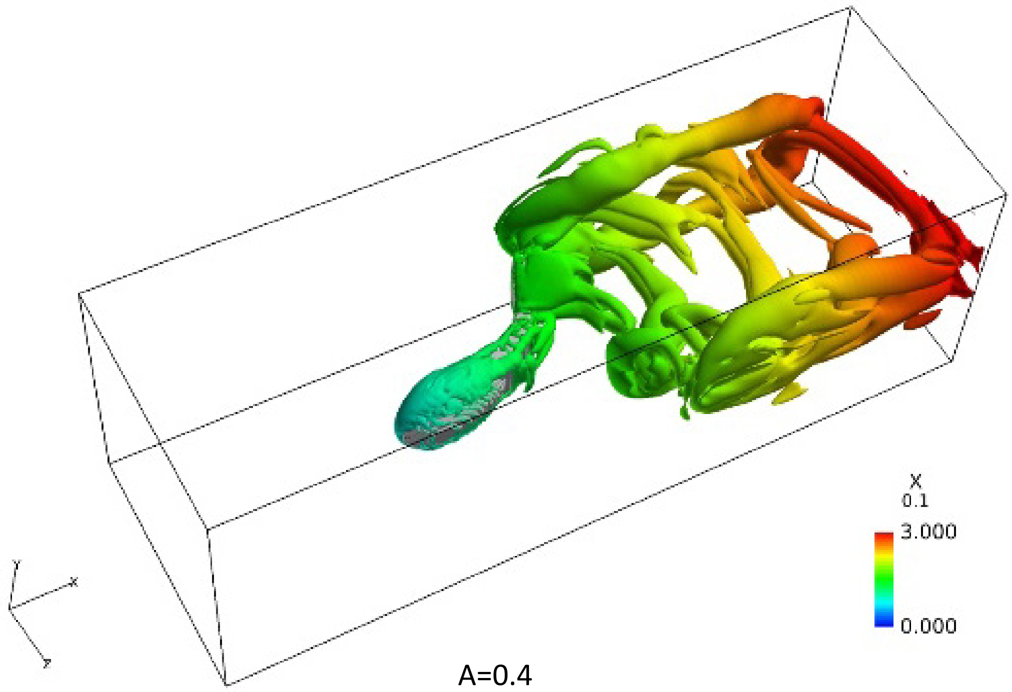
Figure 5.31: Velocity vectors and color contours for cases with different body amplitudes ($\omega = \pi$ and $\phi = \pi/6$, $t = 10$).



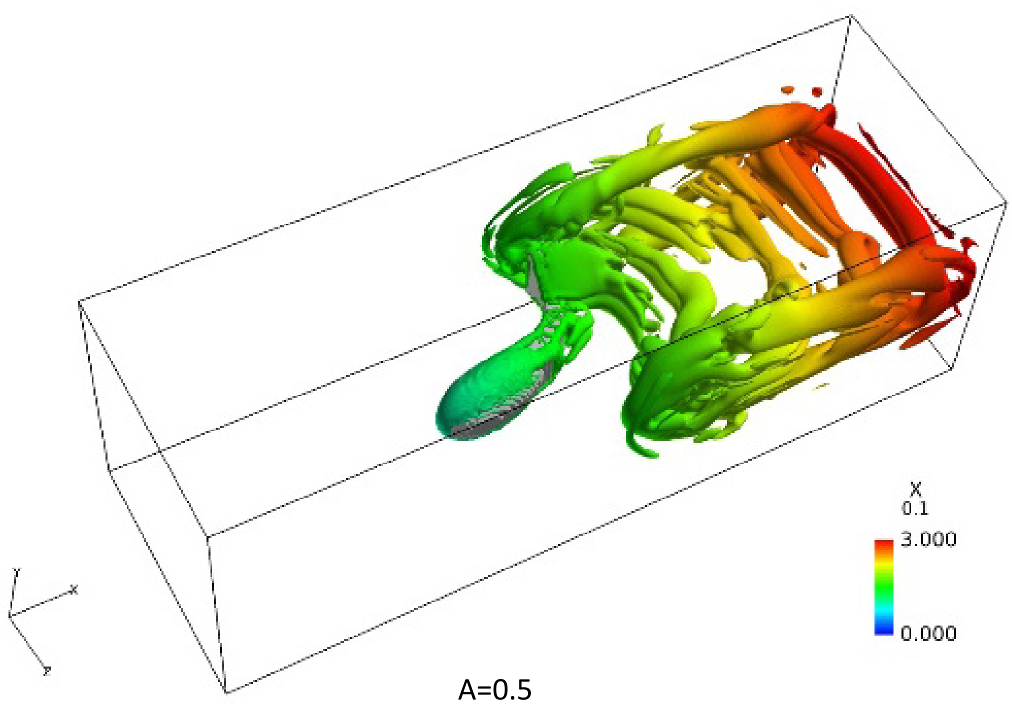
A=0.2



A=0.3

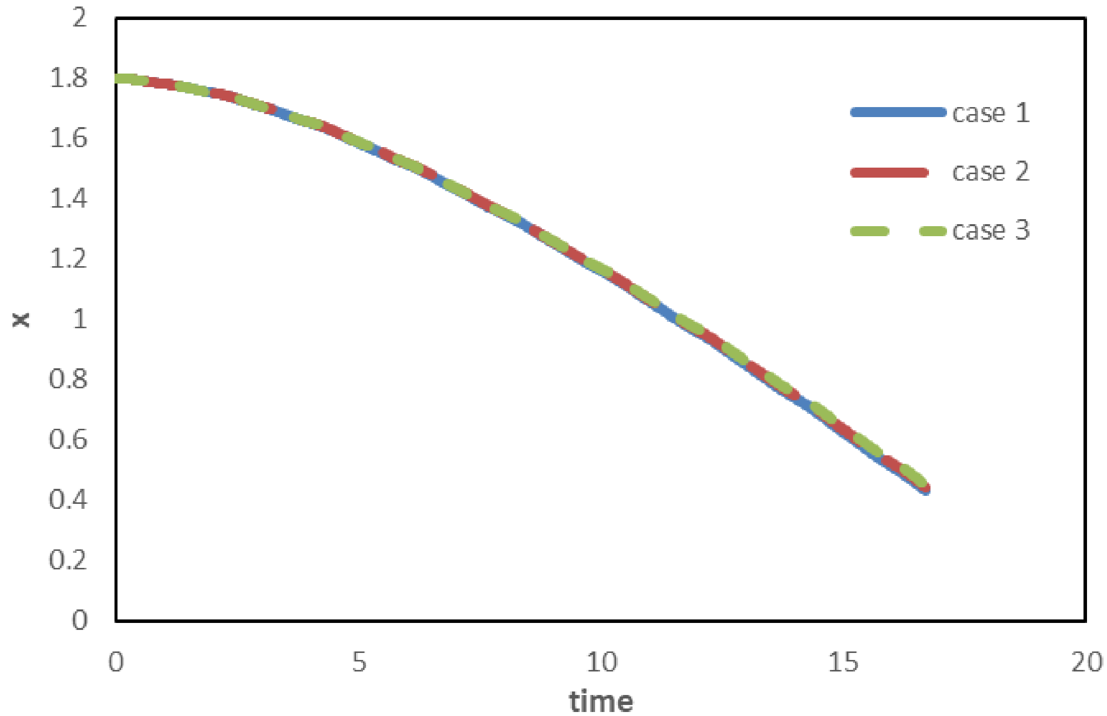


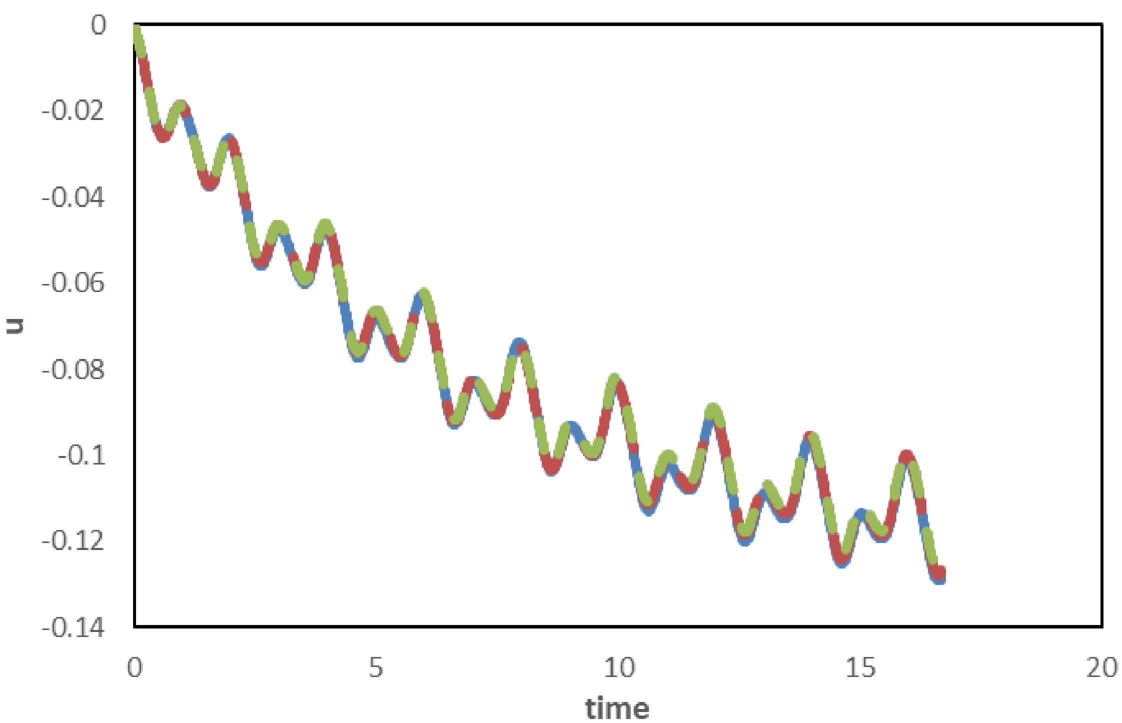
A=0.4

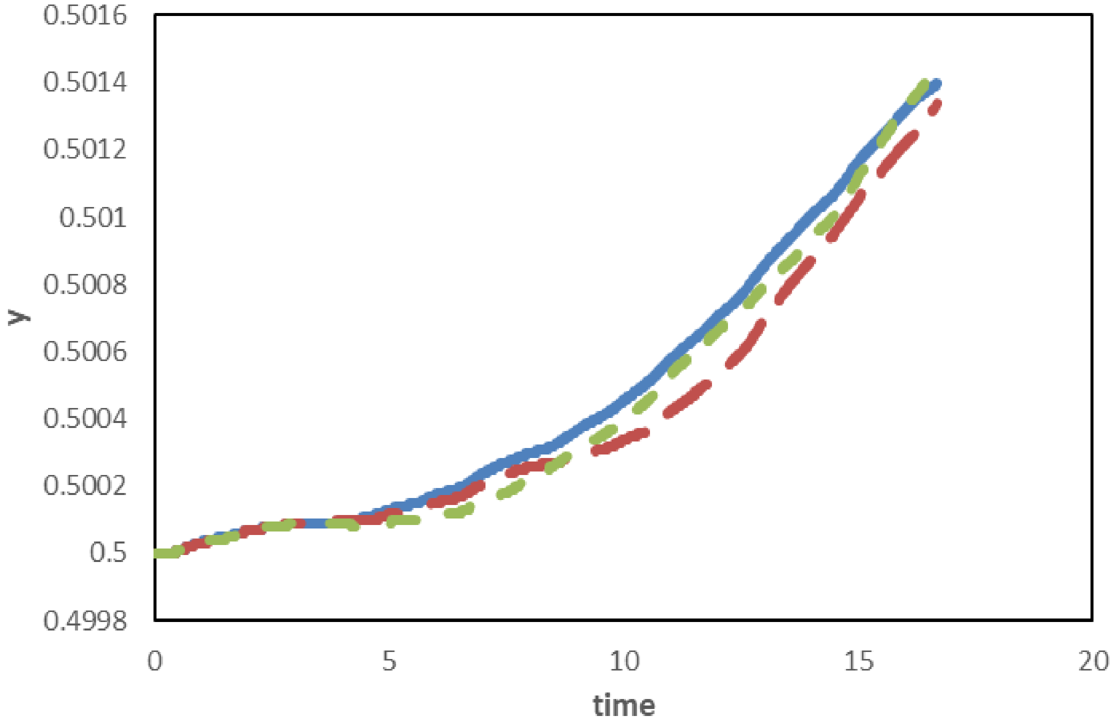


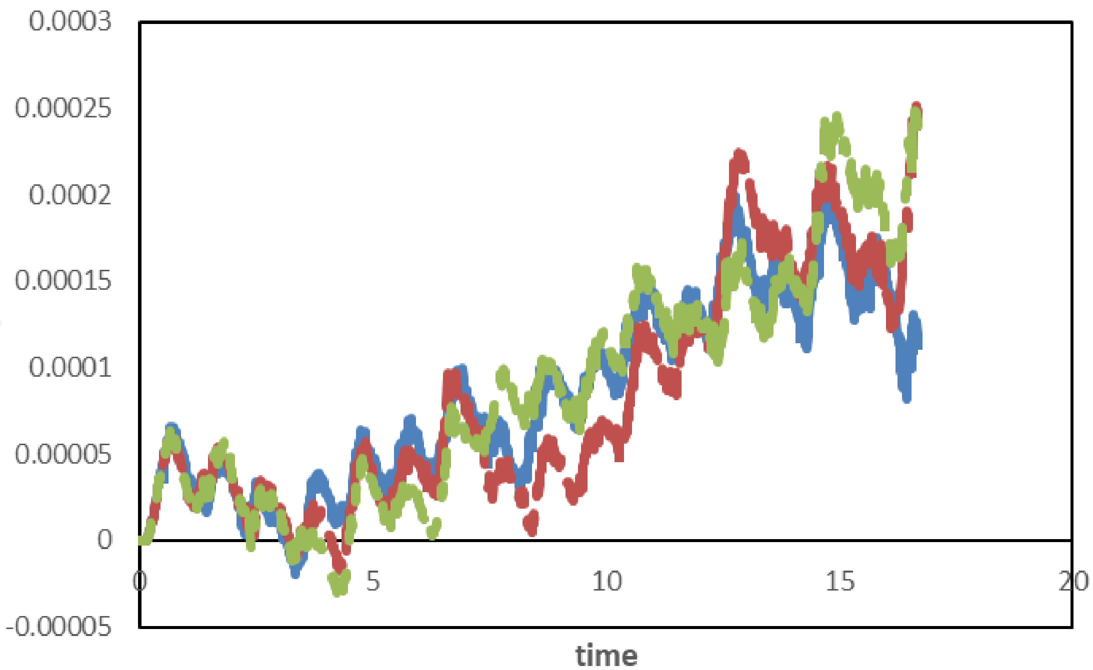
A=0.5

Figure 5.32: Vortex structures for cases with different body amplitudes ($\omega = \pi$ and $\phi = \pi/6$, $t = 10$).


2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
x
0
5
10
15
20
time
case 1
case 2
case 3


0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
-0.14
u
0
5
10
15
20
time


0.5016
0.5014
0.5012
0.501
0.5008
0.5006
0.5004
0.5002
0.5
0.4998
y
0
5
10
15
20
time


0.0003
0.00025
0.0002
0.00015
0.0001
0.00005
0
-0.00005
v
0
5
10
15
20
time

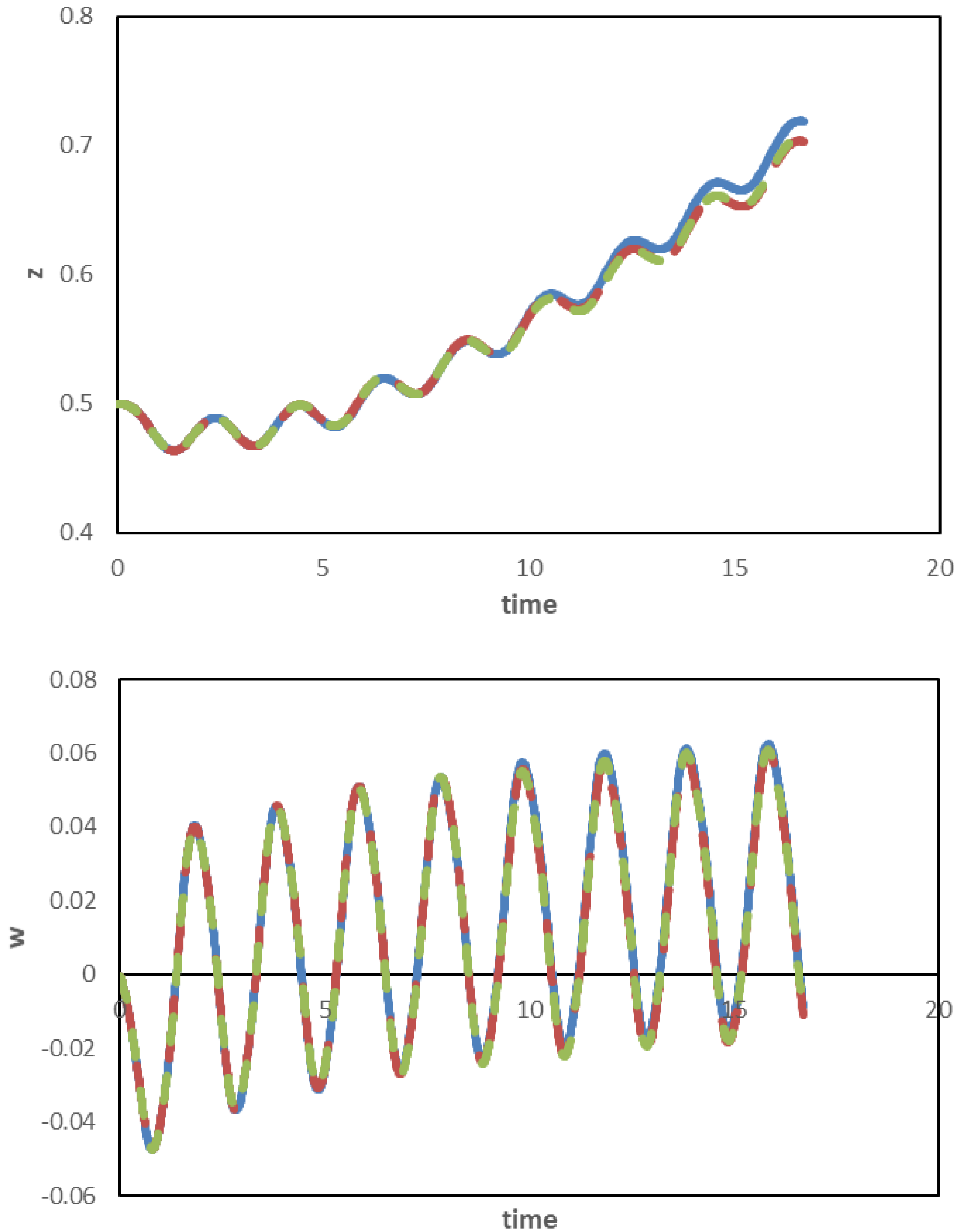
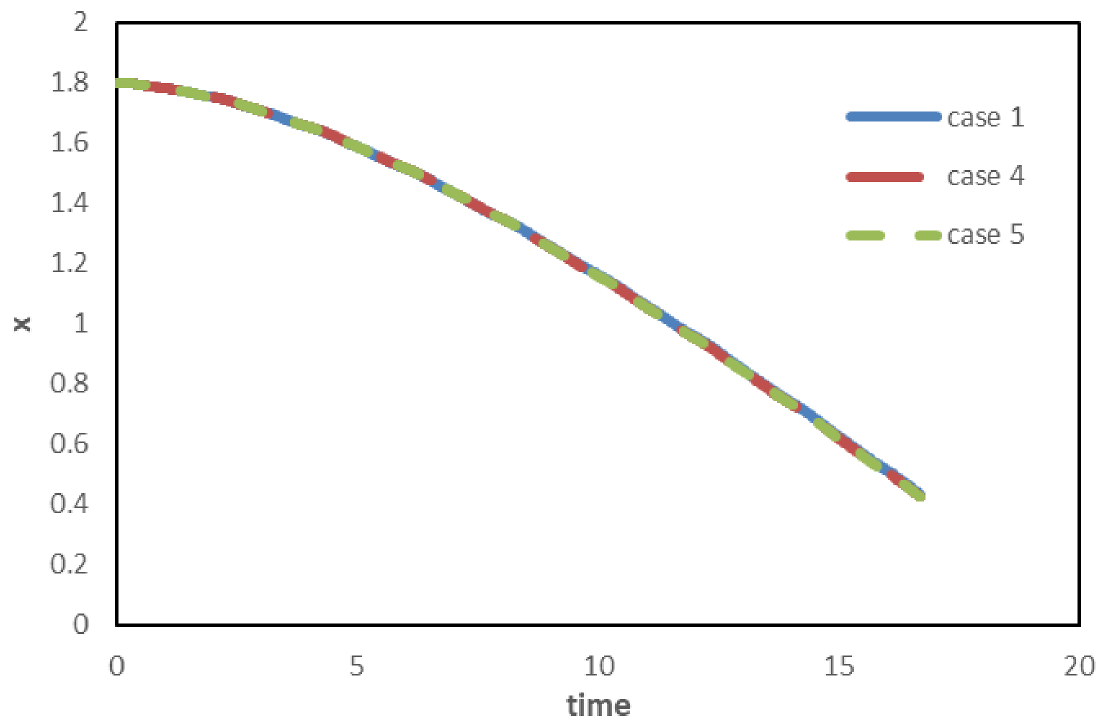
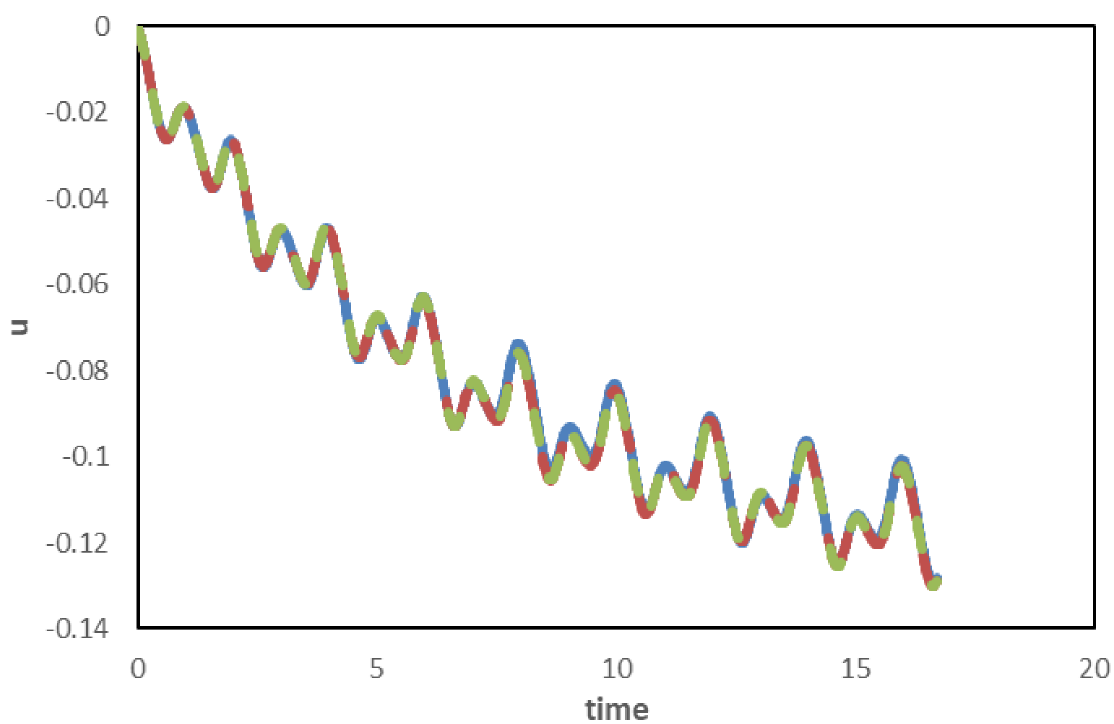
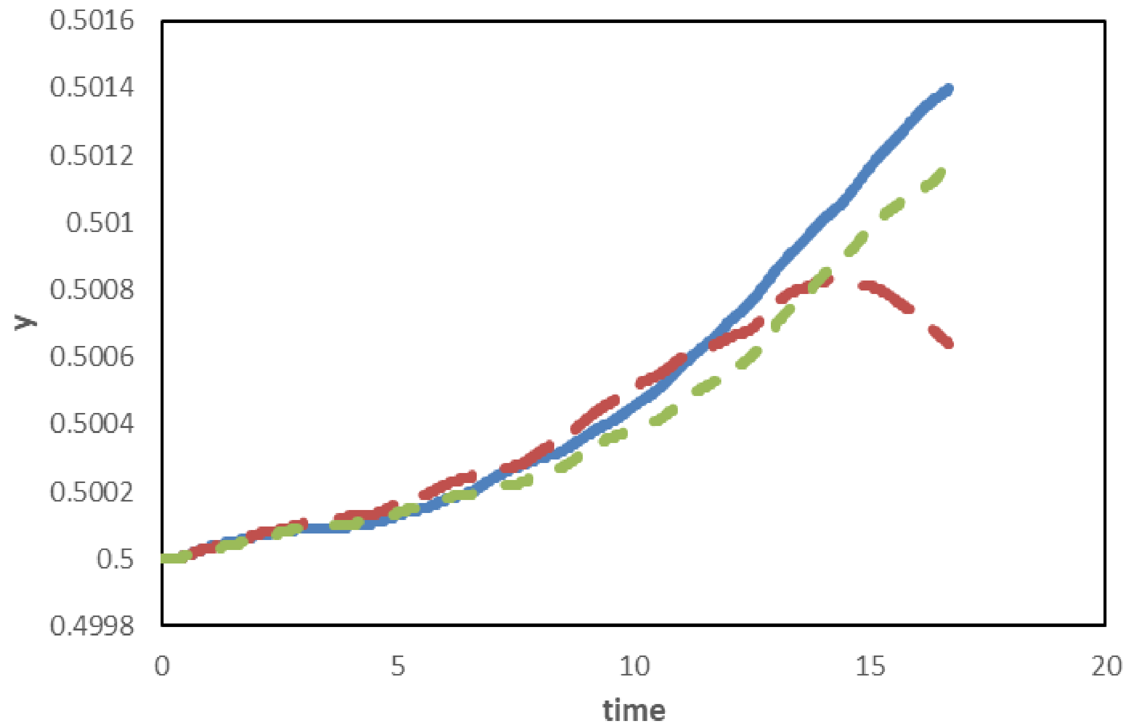
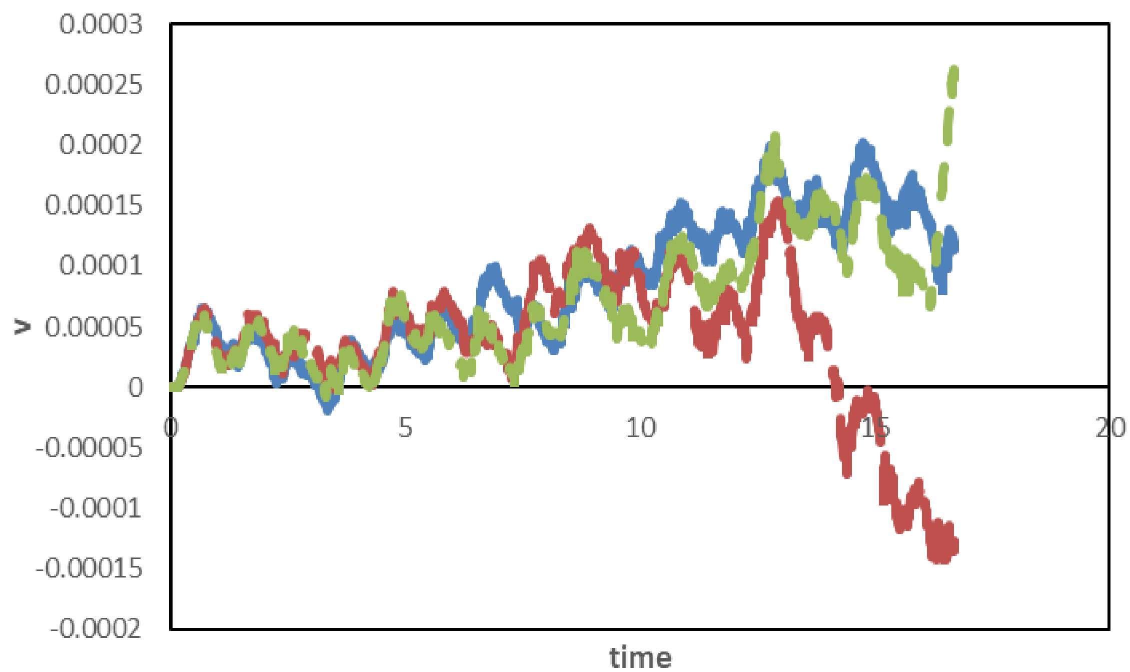


Figure 5.33: Profiles of displacements and velocity components against time along three directions for the cases using grids of different widths.


case 1
case 4
case 5
x
time
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
5
10
15
20


u
time
0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
-0.14
0
5
10
15
20


0.5016
0.5014
0.5012
0.501
0.5008
0.5006
0.5004
0.5002
0.5
0.4998
y
0
5
10
15
20
time


0.0003
0.00025
0.0002
0.00015
0.0001
0.00005
0
-0.00005
-0.0001
-0.00015
-0.0002
v
0
5
10
15
20
time

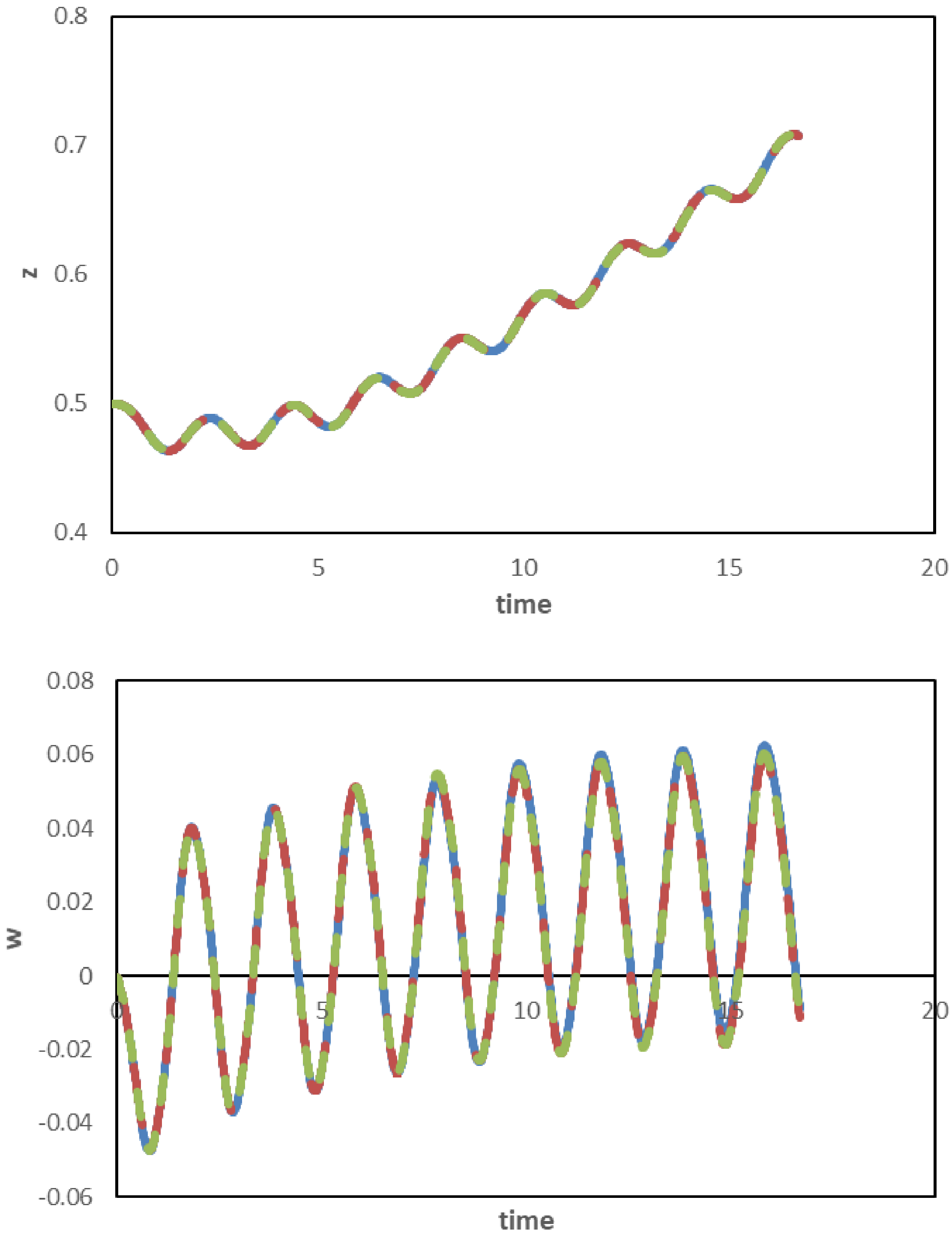


Figure 5.34: Profiles of displacements and velocity components against time along three directions for the cases using grids of different heights.

Chapter 6 Conclusions

In the present study, the immersed boundary method was applied on problems concerning relatively complicated or moving boundaries to investigate the involved flow phenomena. Direct forcing method was adopted considering that there is no need to use specified parameters in the forcing and no extra stability constraints, as well as the capability of building a sharp representation of the immersed boundary, although some disadvantages exist such as the implementation is less straightforward compared to the continuous forcing approach. Some force feedback mechanism was added to the original program code in order to study the hydrodynamic performance of certain movement in fluid such as a self-propelled fish swimming. The performance of the immersed boundary method including the force feedback mechanism was validated firstly through the simulation of flows around a circular cylinder and a single settling sphere particle in fluid. Two application cases of a turbulent open channel flow with wall roughness and a three-dimensional self-propelled fish swimming were then carried out. The conclusions are drawn as follows:

Turbulent open channel flow with wall roughness:

LES of turbulent flow in a channel with a rough wall on one side and a free slip surface on the other was performed by adopting an anisotropy-resolving SGS model. To investigate the grid dependency of the LES results resulting from the SGS model, three grid resolutions were tested under the same definition of a roughness using the immersed boundary method. The obtained results were compared with the corresponding DNS data with and without roughness, as well as the LES results without the extra anisotropic term (EAT).

By comparing the results from smooth and rough DNSs, the presence of roughness caused a downward shift of the mean-velocity profile without any obvious change in slope in the logarithmic law region, which is in line with the general consensus. In addition, from the turbulent stress profiles, there was a shift away from the wall as well as a lower peak value for the streamwise turbulent stress compared with a smooth-wall channel flow.

One main conclusion from the present results is that the grid dependency is not evident for the fine and medium grids in the mean velocity profile as well as turbulence stresses. Some deviation is seen in the mean streamwise velocity profile using the coarse grid. This may be partly a result of reproducing wall roughness using the immersed boundary method, as the rough surface is modified to a height that is slightly lower because the steps were relatively larger. As for the turbulent stresses, the grid dependency seems to bring a non-negligible influence. The vortex structures are also shown to be affected in size, distribution or intensity to some extent, depending on the grid resolution used. Further detailed investigation will be necessary on this issue.

Simulations without the EAT were also performed and their results were compared with those using the anisotropy-resolving SGS model (with EAT). By comparing the results with and without EAT, the mean velocity profile and turbulent stresses agreed relatively well for the fine and medium grid cases, whereas for the coarse grid case, the EAT played a more important role, bringing about a larger difference in prediction accuracy. The present SGS model with EAT successfully provided the predictions of the mean velocity and the Reynolds shear stress better than the model without EAT did. This indicates that the model with EAT can considerably reduce the grid dependency for coarse grid resolutions. Concerning the Reynolds normal stresses, on the other hand, lower values were yielded by the model with EAT compared with those by the model without EAT. Particularly, a remarkable difference was seen in the wall-normal component. In fact, however, this was caused mainly by the effect of the isotropic SGS stress distribution predicted by the model without EAT, where the wrong wall-limiting behavior gave considerably higher wall-normal SGS component. From these results, it has been confirmed that the present SGS model with EAT is useful for application to turbulent flows with wall roughness, although further discussion is necessary to improve the model performance.

3D self-propelled fish swimming:

A three-dimensional numerical simulation of a self-propelled fish-like object swimming in a channel with non-slip wall conditions specified along the lateral and vertical directions was conducted. The moving body was generated by employing the immersed boundary method. A parametric study was performed to investigate the effects of the tail-beat frequency, phase difference and body amplitude on the virtual swimming performance. The obtained results verified the feasibility and effectiveness of the simulation. The details of the flow field can help gain a better understanding of the swimming mechanisms and designs of artificial underwater machines, such as robotic fishes. Differences in swimming performance for various swimming modes highlighted the effect of each parameter and implied the reasonability of the swimming forms in a real fish. The following conclusions can be drawn from the results of the study.

Swimming velocity increases almost linearly with tail-beat frequency. This relationship corresponds well to the knowledge that has been indicated in some previous studies. Wake vortices shed from the tail and formed a reverse Karman vortex street, which played an important role in thrust generation. There were two peak values of longitudinal velocity in the tail-beat cycle corresponding to the period when the tail crossed the vertical central plane of the body.

The results obtained for various phase differences among neighboring rigid body parts show that swimming was considerably more efficient with a small phase difference than with a large one, at least, under the present configuration. However, because the number of segments might not be enough, it should be pointed out that the model used here may need to be modified with more segments for representing undulatory modes in large phase differences more properly.

There exists an optimal body amplitude to obtain a maximum average longitudinal velocity when the other two parameters (tail-beat frequency and phase difference) are fixed. As the amplitude increases, the oscillation of longitudinal velocity increases, while the difference in lateral velocity is not obvious when the amplitude is larger than the optimal value. The discrepancy between the changing tendencies of the oscillation and translational movement in the longitudinal direction as the amplitude increases implies a decrease of swimming efficiency. The fluctuation in vertical velocity implies the importance of pectoral fins for the stability of swimming.

By comparing the cases using grids of different widths or heights, the effect of the lateral and vertical wall distances on the swimming performance was shown to be unobvious, which implies that the aforementioned conclusions may also be applicable to swimming in a wider field that requires greater computational cost.

It is noted that there still remain several limitations to this study, such as the simplification of the fish body and its dynamic modeling. Moreover, in this study, the main concern in swimming performance is the longitudinal velocity, whereas some other aspects such as energy efficiency were not discussed. They should be addressed in future work. In addition, although the test cases involving different simulation fields showed that the effect of lateral or vertical wall distance was not obvious, further detailed investigations of the boundary effects are needed.

Acknowledgements

This research was supported by a Grant-in-Aid for Scientific Research (JP16K05042) from the Japan Society for the Promotion of Science. This work was also supported by the "Advanced Computational Scientific Program" of the Research Institute for Information Technology, Kyushu University, Japan. The present computation was mainly carried out using computer facilities at the Research Institute for Information Technology, Kyushu University, Japan. Y. Zhang was financially supported by the China Scholarship Council.

The author would like to express sincere gratitude to Professor Ken-ichi Abe for the adequate guidance and valuable advice throughout the research, as well as the kind support on the studying and daily life aboard.

The author sincerely appreciates Professor Nobuhiko Yamasaki (Department of Aeronautics and Astronautics, Kyushu University) and Professor Satoshi Watanabe (Department of Mechanical Engineering, Kyushu University) for taking the precious time to review the thesis.

The author is grateful to Assistant Professor Hisashi Kihara for the management and maintenance of the equipment in the lab for a favorable research environment, and the kind guidance on the research and operations. Besides that, various support on daily life is also appreciated.

The author is grateful to all the members in the Fluid Mechanics Laboratory for the various support on the research and daily life, and an enjoyable and memorable experience.

Finally, the author would like to express appreciation to the parents for the financial support and spiritual encouragement during the study.

Reference

Abe, K. (2013) An improved anisotropy-resolving subgrid-scale model with the aid of a scale-similarity modeling concept. *International Journal of Heat and Fluid Flow*. *39*, 42-52.

Abe, K. (2014) An investigation of SGS-stress anisotropy modeling in complex turbulent flow fields. *Flow, Turbulence and Combustion*. *92*, 503-525.

Anderson, J.M., Streitlien, K., Barrett, D.S., & Triantafyllou, M.S. (1998). Oscillating foils of high propulsive efficiency. *J. Fluid Mech*, *360*, 41-72.

Andersson, A.G., Gunnar, J., Hellström, I., Andreasson, P., Lundström, T.S. (2014) Effect of spatial resolution of rough surfaces on numerically computed flow fields with application to hydraulic engineering. *Engineering Applications of Computational Fluid Mechanics*.; *Vol*. *8*, No. 3, *pp*. 373-381.

Arakawa, C. (1994) Computational Fluid Dynamics for Engineering. University of Tokyo Press. (荒川忠一（１９９４）数値流体工学　東京大学出版会.)

Bailon-Cuba, J., Leonardi, S., Castillo, L. (2009) Turbulent channel flow with 2D wedges of random height on one wall. *International Journal of Heat and Fluid Flow*. *30*, 1007-1015.

Balaras, E. (2004). Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations. *Computers & Fluids* 33 375-404.

Bandringa, H. (2010). Immersed boundary methods. Master Thesis, University of Groningen.

Bardina, J., Ferziger, J.H., Reynolds, W.C. (1980 07 14-16) Improved subgrid scale models for large eddy simulation. 13th Fluid and PlasmaDynamics Conference, Fluid Dynamics and Co-located Conference; Snowmass, CO, U.S.A. AIAA Paper, 80-1357.

Bardina, J., Ferziger, J.H., and Reynolds, W.C. (1983) Improved turbulence models based on large eddy simulation of homogeneous, incompressible, turbulent flows. Ph. D. dissertation, Stanford, California: Department of Mechanical Engineering, Stanford University.

Barrett, D.S., Triantafyllou, M.S., Yue, D.K.P., Grosenbaugh, M.A., & Wolfgang, M.J. (1999). Drag reduction in fish-like locomotion. *J. Fluid Mech*, *392*, 183-212.

Beal, D.N., Hover, F.S., Triantafyllou, M.S., Liao, J.C., & Lauder, G.V. (2006). Passive propulsion in vortex wakes. *J. Fluid Mech*, *549*, 385-402.

Bhaganagar, K., Kim, J., Coleman, G. (2004) Effect of Roughness on Wall-Bounded Turbulence. *Flow, Turbulence and Combustion. 72*, 463-492.

Borazjani, I., Ge, L., Sotiropoulos, F. (2008). Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies. *Journal of Computational Physics, 227*, 7587-7620.

Borazjani, I., & Sotiropoulos, F. (2008). Numerical investigation of the hydrodynamics of carangiform swimming in the transitional and inertial flow regimes. *The Journal of Experimental Biology*, *211*, 1541-1558.

Borazjani, I., & Sotiropoulos, F. (2010). On the role of form and kinematics on the hydrodynamics of self-propelled body/caudal fin swimming. *The Journal of Experimental Biology*, *213*, 89-107.

Braza, M., Chassaing, P., & Minh, H.H. (1986). Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder, *J. Fluid Mech. 165*, 79-130.

Chatzikyriakou, D., Buongiorno, J., Caviezel, D., Lakehal, D. (2015) DNS and LES of turbulent flow in a closed channel featuring a pattern of hemispherical roughness elements. *International Journal of Heat and Fluid Flow. 53*, 29-43.

Chung, D., Chan, L., MacDonald, M., Hutchins, N., Ooi, A. (2015) A fast direct numerical simulation method for characterising hydraulic roughness. *J. Fluid Mech*. *vol. 773*, pp. 418-431.

Epps, B.P., Alvarado, P.V.y., Youcef-Toumi, K., & Techet, A.H. (2009). Swimming performance of a biomimetic compliant fish-like robot. *Exp Fluids, 47,* 927-939.

Fadlun, E.A., Verzicco, R., Orlandi, P., and Mohd-Yusof, J. (2000). Combined Immersed-Boundary Finite-Difference Methods for Three-Dimensional Complex Flow Simulations. *Journal of Computational Physics* 161, 35-60.

Fauci, L.J., & Peskin, C.S. (1988). A computational model of aquatic animal locomotion. *Journal of Computational Physics 77*, 85-108.

For fluid analysis / structure analysis, high quality mesh generator Gridgen, (流体解析/構造解析用　高品質メッシュジェネレーター Gridgen,) http://www.vinas.com/jp/seihin/gridgen/index.html

Germano, M., Piomelli, U., Moin, P., Cabot, W.H. (1991) A dynamic subgrid-scale eddy viscosity model. *Phys. Fluids* A 3 (7)*,* 1760-1765.

Gillis, G.B. (1996). Undulatory locomotion in elongate aquatic vertebrates: anguilliform swimming since Sir James Gray[1]. *AMER. ZOOL*., *36*, 656-665.

Goldstein, D., Handler, R., and Sirovich, L. (1993). Modeling a No-Slip Flow Boundary with an External Force Field. *Journal of Computational Physics* 105, 354-366.

Hirose, T. (2013) Fundamental research on the development of analysis technology for complex turbulence combining unstructured grid and Immersed Boundary Method. Master Thesis. (廣瀬高明（２０１３）非構造格子と Immersed Boundary Method を併用した複雑乱流解析技術の高度化に関する基礎研究 修士論文.)

Hirose, T., Kihara, H., & Abe, K. (2012, November). Numerical simulation of flow field around a wing with surface oscillation using an immersed boundary method. *The 90th conference of JSME fluid dynamics division*, CD-ROM, Kyoto, Japan.

http://www.rss21.iis.u-tokyo.ac.jp/

http://www.ciss.iis.u-tokyo.ac.jp/rss21/.

Iaccarino, G., Verzicco, R. (2003). Immersed boundary technique for turbulent flow simulations. *Appl Mech Rev* vol 56, no 3.

Jin, Y., Uth, M.F., Herwig, H. (2015) Structure of a turbulent flow through plane channels with smooth and rough walls: An analysis based on high resolution DNS results. *Computers & Fluids*. *107*, 77-88.

Jordan, S., & Fromm, J. (1972). Oscillatory Drag, Lift, and Torque on a Circular Cylinder in a Uniform Flow. *Physics of Fluids 15*, 371-376.

Kajishima, T. (1999) Numerical Simulation of Turbulent Flows. Yokendo. (梶島岳夫（１９９９）乱流の数値シミュレーション　養賢堂.)

Keirsbulck, L., Mazouz, A., Labraga, L., Tournier, C. (2001) Influence of the surface roughness on the third-order moments of velocity fluctuations. *Experiments in*

Fluids. *30*, 592-594.

Kern, S., & Koumoutsakos, P. (2006). Simulations of optimized anguilliform swimming. *The Journal of Experimental Biology*, *209*, 4841-4857.

Kim, J., Kim, D., & Choi, H. (2001). An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries. *J. Comput. Phys. 171*, 132–150.

Kobayashi, T. (2004) Numerical computation using the next-generation fluid analysis software Advance/FrontFlow/red. AdvanceSoft Corporation. (小林敏雄（２００４）監修：次世代流体解析ソフトウェア　Advance/FrontFlow/red を用いた数値計算　アドバンスソフト株式会社.)

Krogstad, P.-Å., Antonia, R.A. (1999) Surface roughness effects in turbulent boundary layers. *Experiments in Fluids*. *27*, 450-460.

Lauder, G.V., & Drucker, E.G. (2002). Forces, fishes, and fluids: hydrodynamic mechanisms of aquatic locomotion. *News Physiol Sci, 17,* 235-240.

Liao, J.C. (2007). A review of fish swimming mechanics and behavior in altered flows. *Phil. Trans. R. Soc. B*, *362*, 1973-1993.

Lilly, D.K. (1992) A proposed modification of the Germano subgrid-scale closure method. *Phys. Fluids* A 4 (3), 633-635.

Liu, H., & Kawachi, K. (1999). A numerical study of undulatory swimming. *Journal of Computational Physics*, *155*, 223-247.

Liu, J., & Hu, H. (2010). Biological inspiration: from carangiform fish to multi-joint robotic fish. *Journal of Bionic Engineering, 7*, 35-48.

Long, J.H., JR., Mchenry, M.J., Boetticher, N.C. (1994). Undulatory swimming: how traveling waves are produced and modulated in sunfish (*Lepomis gibbosus*). *J. exp. Biol*, *192*, 129-145.

Long, J.H., JR., & Nipper, K.S. (1996). The importance of body stiffness in undulatory propulsion. *Amer. Zool*., *36*, 678-694.

Luo, K., Wang, Z., & Fan, J. (2007). A modified immersed boundary method for simulations of fluid-particle interactions. *Comput. Methods Appl. Mech. Engrg*, *197*, 36-46.

Marchis, M.D., Napoli, E. (2012) Effects of irregular two-dimensional and three-dimensional surface roughness in turbulent channel flows. *International Journal of Heat and Fluid Flow*.; *36*, 7-17.

Mittal, R., Iaccarino, G. (2005). Immersed boundary methods. *Annu. Rev. Fluid Mech*, 37: 239-61.

Mohd-Yusof, J. (1997). Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries. CTR Annual Research Briefs, 161(1): 317-327, NASA Ames/Stanford University.

Mordant, N., & Pinton, J.-F. (2000). Velocity measurement of a settling sphere. *Eur. Phys. J. B 18*, 343-352.

Moser, R.D., Kim, J., Mansour, N.N. (1999). Direct numerical simulation of turbulent channel flow up to $Re_\tau = 590$. *Physics of Fluids* 11, 943-945.

Muto, M., Tsubokura, M., & Oshima, N. (2012). Negative Magnus lift on a rotating sphere at around the critical Reynolds number. *Physics of Fluids, 24*, 1-15.

Muzaferija, P.A. (1994). Adaptive finite volume method for flow predictions using unstructured meshes and multigrid approach. PHD Thesis, Univ. of London.

Nagano, Y., Hattori, H., Houra, T. (2004) DNS of velocity and thermal fields in turbulent channel flow with transverse-rib roughness. *International Journal of Heat and Fluid Flow*. *25*, 393-403.

Naito, H., & Fukagata, K. (2012). Numerical simulation of flow around a circular cylinder having porous surface. *Physics of Fluid, 24*, 117102.

Narasimha, M., Brennan, M., & Holtham, P.N. (2007). A review of CFD modelling for performance predictions of hydrocyclone. *Engineering Applications of Computational Fluid Mechanics Vol. 1, No. 2,* pp.109-125.

Noor, D.Z., Chern, M.J., and Horng, T.L. (2009). An immersed boundary method to solve fluid-solid interaction problems. *Comput Mech* 44: 447-453.

Peskin, C.S. (1972). Flow patterns around heart valves: a numerical method. *Journal of computational physics* 10, 252-271.

Peskin, C.S. (1977). Numerical analysis of blood flow in the heart. *Journal of computational physics* 25, 220-252.

S. Leonardi, I., Castro, P. (2010) Channel flow over large cube roughness: a direct numerical simulation study. *J. Fluid Mech. vol. 651*, *pp.* 519-539. Cambridge University Press 2010.

Schultz, M.P., Flack, K.A. (2007) The rough-wall turbulent boundary layer from the hydraulically smooth to the fully rough regime. *J. Fluid Mech. vol. 580, pp. 381-405.* Cambridge University Press, 2007.

Sfakiotakis, M., Lane, D.M., & Davies, J.B.C. (1999). Review of fish swimming modes for aquatic locomotion. *IEEE Journal of Oceanic Engineering*, *24*, 237-252.

Shamloo, H., Pirzadeh, B. (2015) Analysis of roughness density and flow submergence effects on turbulence flow characteristics in open channels using a large eddy simulation. *Applied Mathematical Modelling*. *39*, 1074-1086.

Smagorinsky, J. (1963) General circulation experiments with the primitive equations. I. The basic experiment. Mon. *Weather Rev*. 91, 99.

Stamou, A.I., Katsiris, I., Politis, M., & Schaelin, A. (2008). Applying a CFD model to evaluate thermal comfort in the MPC amphitheatre of the Olympic Games "Athens 2004". *Engineering Applications of Computational Fluid Mechanics Vol. 2, No. 1*, pp. 41-50.

Taira, K., Colonius, T. (2007). The immersed boundary method: A projection approach. *Journal of Computational Physics* 255 2118-2137.

Tosaka, N., Miyata, H., Shimazaki, Y., Nomura, T., Shimura, M., and Hatanaka, K. (1995) Analysis of Moving Boundary Flows (Computational Fluid Dynamics Series 4; Ed. By M. KAWAHARA). University of Tokyo Press. (数値流体力学編集委員会（１９９５）編：数値流体力学シリーズ４　移動境界流れ解析　東京大学出版会.)

Triantafyllou, M.S., Techet, A.H., Zhu, Q., Beal, D.N., Hover, F.S., & Yue, D.K.P. (2002). Vorticity control in fish-like propulsion and maneuvering[1]. *Integr. Comp. Biol.*, *42*, 1026-1031.

Tritton, D.J. (1959). Experiments on the flow past a circular cylinder at low Reynolds numbers. *J. Fluid Mech. 6*, 547.

Tseng, Y., & Ferziger, J.H. (2003). A ghost-cell immersed boundary method for flow in complex geometry. *Journal of Computational Physics 192*, 593-623.

Tsikata, J.M., Tachie, M.F. (2013) Adverse pressure gradient turbulent flows over rough walls. *International Journal of Heat and Fluid Flow*. *39*, 127-145.

Unemura, T. (2003) IT program by Ministry of Education, Culture, Sports, Science and Technology (MEXT). The exploration of the next-generation fluid analysis software FrontFlow/red. The 17th Symposium on Computational Fluid Dynamics. (畝村毅（２００３）文部省 IT プログラム　次世代流体解析ソフトウェア FrontFlow/red の開発　第１７回数値流体力学シンポジウム.)

Vreman, B., Geurts, B., Kuerten, H. (1994) On the formulation of the dynamic mixed subgrid-scale model. *Phys. Fluids 6* (12), 4057-4059.

Wang, F., Reitz, R., Pera, C., Wang, Z., & Wang, J. (2013). Application of generalized RNG turbulence model to flow in motored single-cylinder PFI engine. *Engineering Applications of Computational Fluid Mechanics Vol. 7, No. 4,* pp. 486-495.

Webb, P.W. (1993). The effect of solid and porous channel walls on steady swimming of steelhead trout *oncorhynchus mykiss*. *J. exp. Biol*, *178*, 97-108.

Webber, D.M., Boutilier, R.G., Kerr, S.R., & Smale, M.J. (2001). Caudal differential pressure as a predictor of swimming speed of cod (*Gadus morhua*). *The Journal of Experimental Biology*, *204,* 3561-3570.

Wu, T.Y. (2001). Mathematical biofluiddynamics and mechanophysiology of fish locomotion. *Math. Meth. Appl. Sci*, *24*, 1541-1564.

Zang, Y., Street, L.R., and Koseff, J.R. (1993) A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows. *Phys. Fluids* A 5 (12), 3186-3196.

Zhu, Q., Wolfgang, M.J., Yue, D.K.P., & Triantafyllou, M.S. (2002). Three-dimensional flow structures and vorticity control in fish-like swimming. *J. Fluid Mech*, *468*, 1-28.