A Velocity Decomposition Approach for Three-Dimensional Unsteady Viscous Flow at High Reynolds Number

by

Yang Chen

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Naval Architecture and Marine Engineering)
in The University of Michigan
2017

Doctoral Committee:

Assistant Professor Kevin J. Maki, Chair
Professor Robert F. Beck
Assistant Professor Eric Johnsen
Professor Armin W. Troesch

Yang Chen

chanyang@umich.edu

ORCID iD: 0000-0001-5756-9335

Dedicated to my family

ACKNOWLEDGEMENTS

I am eternally grateful to my advisor Prof. Kevin Maki who has been an incredible mentor for me in both academic and life. I would like to thank my other committee members, Prof. Robert Beck, Prof. Eric Johnsen and Prof. Armin Troesch for their service in aiding the compilation of this thesis.

The previous work done by Dr. William Rosemurgy IV and Dr. Deborah Edmund is greatly appreciated. I am thankful to Dr. Dominic Piro for his help on Aegir. I would also like to thank other members of the CSHL (Computational-Ship-Hydrodynamics Lab) for their friendship and the stimulating discussions. I am indebted to a long list of people that have positive influence on me. To name everyone would be impossible, but I would like them to know that I am eternally thankful.

I would like to express my greatest gratitude to my parents for their unconditional love and support. I would also like to thank Jiaqi for her love and support, and for making me a better person.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

ABSTRACT

A Velocity Decomposition Approach for Three-Dimensional
Unsteady Viscous Flow at High Reynolds Number

by

Yang Chen

Chair: Prof. Kevin J. Maki

A velocity decomposition method is developed for the solution of three-dimensional, unsteady flows. The velocity vector is decomposed into an irrotational component (viscous-potential velocity) and a vortical component (vortical velocity). The vortical velocity is selected so that it is zero outside of the rotational region of the flow field and the flow in the irrotational region can thus be solely described by the viscous-potential velocity. The formulation is devised to employ both the velocity potential and the Navier-Stokes-based numerical methods such that the field discretization required by the Navier-Stokes solver can be reduced to only encompass the rotational region of the flow field and the number of unknowns that are to be solved by the Navier-Stokes solver is greatly reduced. A higher-order boundary-element method is used to solve for the viscous potential by applying a viscous boundary condition to the body surface. The finite-volume method is used to solve for the total velocity on a reduced domain, using the viscous-potential velocity as the boundary condition on the extent of the domain. The viscous-potential velocity and the total velocity are time dependent due to the unsteadiness in the boundary layer and the wake. A two-way coupling algorithm is developed to tightly couple the two solution procedures in time. The velocity-decomposition-coupled solver developed in this work is used to solve three-dimensional, laminar and turbulent unsteady flows. For turbulent flows, the solver is applied for both Unsteady-Reynolds-Averaging-Navier-Stokes and Large-Eddy-Simulation computations. The solver is demonstrated to be capable of solving problems with realistic geometries. Preliminary results for 3D lifting flow are also presented. By using the velocity-decomposition-coupled solver, the solution can be determined on a greatly reduced domain and have the same accuracy as that calculated by a conventional Navier-Stokes solver on a large domain. For some test cases, the number of unknowns in the computational mesh is reduced up to 50%.

CHAPTER I

Introduction

The role of Computational Fluid Dynamics (CFD) has become increasingly important in the engineering analysis for a wide range of engineering applications, such as those in naval engineering, aerospace engineering, and automotive industry. Challenging calculations like those using the Large-Eddy Simulation (LES) technique, are gaining popularity in applications as the availability of computational power increases (*Filip* (2013), *Filip and Maki* (2015)). However, high-fidelity simulations are impractically expensive for naval applications, due to some unique challenges related to marine hydrodynamics. One of them is the "infinite domain" needed to represent the vast open ocean. When using field methods, such as finite-volume, finite-difference and finite element, which are the workhorses of numerical methods for CFD for solving the flow problems, the computational domains need to be large to mimic the open ocean, such that the often-used free-stream far-field boundary condition is valid. Another challenge is that, when simulating free-surface flow around naval vessels, the meshes need to be dense to avoid excessive numerical damping and to capture various wave components in the sea, either generated by the vessels or from the sea environment. This is especially important for seakeeping simulations in irregular seas where the environmental waves contain a wide range of wave components with various wave lengths. Hence, the combination of the large domain and dense meshes demands a prohibitive number of cells which means a high computational cost for CFD. This research is part of the effort for providing computational tools for naval hydrodynamics based on domain-reduction techniques, which provides a solution for these challenges. The main contribution of this thesis is the development of the velocity decomposition solver for three-dimensional, transient flow simulations with realistic ship geometries.

1.1 Background

The popular computational methods for naval hydrodynamic calculations can be roughly categorized as either the inviscid-flow or viscous-flow based formulations. It is worth pointing out that although a large portion of the inviscid-flow solvers are using a potential-flow based solution method, the notion of potential flow only represents the irrotationality of the flow field and does not infer anything about the viscosity of the fluid itself. Given a proper formulation, a viscous potential that satisfies the Navier-Stokes (N-S) problem (*Joseph* (2006)) can be calculated. Since the potential-flow based solvers used in the naval hydrodynamic community are mostly inviscid, they are put in the inviscid-flow method category, although in many cases the inviscid potential closely satisfies the N-S solution outside the rotational-flow region.

The inviscid-flow methods are widely used for engineering analysis for naval vessels and offshore structures to evaluate the maneuvering and seakeeping performance. The flow field around the bodies is assumed irrotational and the velocity field can be represented by the gradient of a scalar potential function. The Boundary-Element Method (BEM) is usually used to solve for the potential, which means only the boundary surfaces need to be discretized. The potential-flow based methods are generally less computationally expensive compared to the viscous-flow methods. However, limited by its irrotational flow assumption, potential-flow based methods generally can not account for important viscous effects, such as turbulent wakes and vortex shedding. The lack of proper treatment of the viscous rotational flows in the regions around the stern, propeller and rudder for inviscid-flow methods, gives rise to the need of viscous-flow methods.

For the viscous-flow methods, the most popular approach is to calculate the primitive variables (velocity vector, pressure) by numerically solving the Navier-Stokes (N-S) equations (such methods are commonly referred to as CFD). Field methods, such as Finite-Element, Finite-Difference and Finite-Volume Methods (FVM), are usually used to numerically solve the governing equations. Viscous-flow methods are able to account for viscous effects, but they can be computationally expensive, because the turbulence structures, in the flow about a ship, have very small time scales and length scales. This means very fine temporal and spatial discretization are required. Moreover, one must discretize the entire flow domain instead of just the flow boundaries. Field-method CFD produces large and sparse matrices. The large sparse matrix can be solve efficiently with iterative methods. For a typical iterative method, such as GMRES (Generalized Minimal Residual Method), the computational effort scales with $\mathcal{O}(N_{\mathrm{CFD}})$, where N_{CFD} is the number of unknowns of the problem. On the other hand, the matrix generated by BEM is much smaller, but it is dense. Solving this small but dense matrix can be computationally expensive. A popular method for solving this matrix is LU decomposition, of which the computational effort scales with $\mathcal{O}(N_{\mathrm{BEM}}^3)$, where N_{BEM} is the unknowns of the problem. However, the sheer scale of the matrices in field-method CFD due to the fine resolution required for solving 3D, high-Reynolds-number, turbulent flow, overwhelms the advantage gained by its sparsity.

Due to the various aspects that result in the prohibitive computational cost in viscous-flow methods, seakeeping and maneuvering problems are far out of reach for the current computation power for most of engineering analyses. To bridge the gap between the inviscid and viscous flow methods, and reduce the cost of the viscous flow methods, a large amount of research has been done to devise a hybrid method to combine the advantage of the inviscid and viscous flow solutions to provide an efficient and accurate computational tool, using the idea of viscous and inviscid interaction.

At the early stage of the research to utilize hybrid methods for numerical solvers, a popular style is to incorporate a boundary layer formulation to help the inviscid flow solver to account for some degree of viscous effects, such as the displacement thickness due to the boundary layer. *Lemmerman and Sonnad* (1979) utilize the equivalent source approach (also known as the transpiration-velocity approach) in *Lighthill* (1958) to model the boundary layer effects. In *Drela and Giles* (1987), the Euler equations are solved together with the boundary layer equation and results for flow with separation have been demonstrated.

As the computational power has grown, the theme of the research for hybrid methods has shifted toward using a viscous-flow solver as the main driver, while using inviscid-flow methods to facilitate a more efficient and accurate solution process.

For most engineering applications, the Reynolds number (Re) of the flow field is usually in the range of 10^6 to 10^9. Also the viscous effects are usually confined within a small region around the body and the wake. A straightforward exploitation of this character of the flow field to improve the accuracy and efficiency of the CFD solver is sometimes referred to as the “far-field correction”, which is to set the potential velocity as the boundary condition of the CFD solver instead of using the free-stream velocity. In *Eça and Hoekstra* (2009), improvements of the Navier-Stokes solution is shown within a smaller domain through using this approach. However, the improvement is limited since the interaction between the viscous and inviscid solution is not fully incorporated.

A large amount of research has been done to devise different ways to achieve the viscous and inviscid interaction of the flow solutions. These methods can generally be put into two categories, domain-decomposition methods and velocity-decomposition methods.

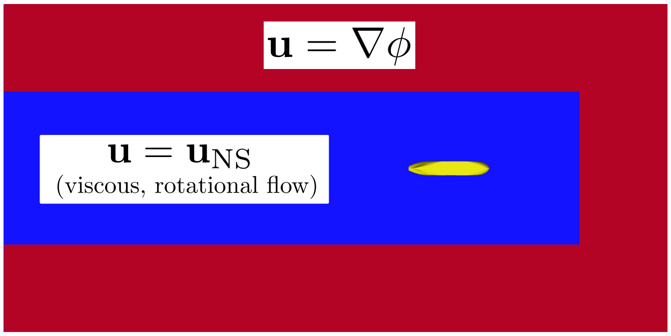

$\mathbf{u} = \nabla\phi$
$\mathbf{u} = \mathbf{u}_{\mathrm{NS}}$
(viscous, rotational flow)

Figure 1.1: The schematic for a conventional domain decomposition method

1.1.1 Domain Decomposition Methods

For high Reynolds-number flows, the viscous effects are usually confined within a small area around the body. The domain decomposition method is introduced to decomposed the flow domain into different regions and suitable solution technique for each specific region can be applied. An example is shown in Fig. 1.1, where the solutions in the inner domain (colored in blue) are solved by a N-S solver. The solutions in the outer domain (colored in red), in which the flow is assumed to be irrotational, are described by a potential-flow model. The two solutions are coupled by a matching boundary condition at the interface of the two sub-domains. Since it is relatively straightforward to implement, the domain decomposition method is a popular choice by researchers who desire to improve the efficiency and accuracy of the solution process. For most of the domain decomposition approaches, the solver that is responsible for the solution of each of the sub-domains does not require modification. Only a matching condition is needed to couple the solutions from each sub-domain.

In *Larsson et al.* (1990) and *Larsson et al.* (1991), the computational domain is divided into three zones. The solution in majority of the domain is calculated by using a panel method with a boundary layer program applied on a surface covering the part of the body where the flow is attached. Then, the solution from these two zones are applied as boundary conditions for the N-S solver to solve for the flow at the region where strong viscous effects are anticipated.

In *Campana et al.* (1995), the computational domain is decomposed into two parts, one contains the body and viscous wake, the other one is the remainder of the flow volume. The conventional RANS (Reynolds-Averaged-Navier-Stokes) method is applied to the former region that has rotational flow. Then the flow field in the other region is described by a potential-flow model, which is solved using a panel method. These two domains are coupled by enforcing a matching condition within their overlap section. The domain decomposition approach in *Campana et al.* (1995) is carried further to solve for unsteady wave-breaking flows in *Iafrati and Campana* (2003). The fluid domain away from the free-surface is modeled using potential flow method. The fluid near the free-surface is solved using the RANS equations while the air-water interface is captured by a level-set technique. In *Chen and Lee* (1996) and *Chen and Lee* (1999), a similar technique was used, except that the potential flow solution is solved using a finite-difference method. In *Guillerm and Alessandrini* (2003), the same type of technique was applied, except that the potential and the N-S solutions are coupled through a Fourier-Kochin method. *Hamilton and Yeung* (2011) presented a generalized two-way coupling method to match the linearized inviscid outer domain and the linear viscous inner domain.

Domain decomposition method are also popular in the coastal engineering community due to potential flow model's efficiency of specifying incident waves. *Zhang et al.* (2013) and *Zhang et al.* (2015) have deployed the domain decomposition method with non-overlapping domains to solve wave-impact problems. In their work, a Mixed-Eulerian-Lagrangian free-surface tracking method is used for the potential flow model. The free-surface elevation between the Finite-Element model and the potential flow model is joined through a spline interpolation. *Paulsen et al.* (2014) has applied a one-way coupled domain decomposition method with a finite-difference nonlinear potential flow model. The free-surface is tracked using Volume-of-Fluid method with the Navier-Stokes solver in the inner domain.

Beyond the scope of viscous inviscid interaction, *Colicchio et al.* (2015) applied a dynamic domain decomposition approach to couple a finite-difference Navier-Stokes solver with a Multiple-Relaxation Time Lattice Boltzmann method. The authors attemp to improve the computational efficiency by dynamically decomposing the domain and allowing the Lattice-Boltzmann solver to handle the fast varying flow region. The method is tested on the impulsively started flow over a two-dimensional (2D) cylinder.

Overall, the domain decomposition approach provides a convenient means for researchers to couple appropriate solvers to increase computation efficiency and accuracy. However, the matching condition can be troublesome, because it usually needs to be far away from the body (outside the vortical region of the flow domain), and this demands high resolution and increases cost.

1.1.2 Velocity Decomposition Methods

The decomposition of the velocity vector has been long exploited for the analysis of fluid flows. The velocity of the fluid can be decomposed into components that respectively depends on the expansion rate, vorticity and boundary conditions of the flow field *Batchelor* (2000); *Serrin* (1959). In *Wehausen* (1973), a velocity decomposition is applied for separation of wave and viscous resistance of ships. *Beck* (1971) models the effect of the ship wake on the total wave resistance by decomposing the total velocity field into components due to the thin ship, the rotational wake and the interaction between the two. In *Dias et al.* (2008), the decomposition of the velocity vector is used to theoretically study the evolution of water waves under the influence of viscosity. *Morino* (1986) generalizes the equivalent source approach in *Lighthill* (1958), so that the method is applicable for flows with separation and no

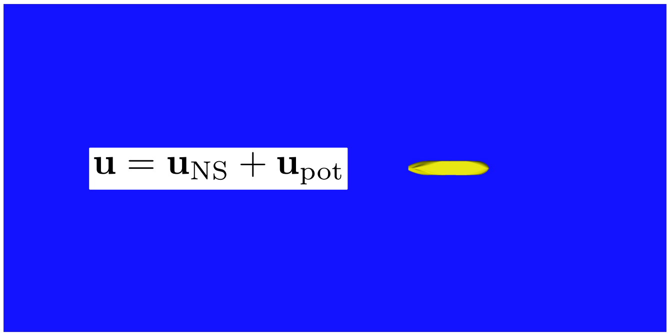

$\mathbf{u} = \mathbf{u}_{\mathrm{NS}} + \mathbf{u}_{\mathrm{pot}}$

Figure 1.2: The schematic for a conventional velocity decomposition method

longer limited to flows with thin boundary layers. In *Morino* (1990), viscous flows are calculated by a Boundary-Element Method (BEM) using a special form of velocity decomposition so that the BEM can retain its efficiency.

Velocity decomposition has also been used to improve the efficiency of field methods for numerically solving the Navier-Stokes equations. In general the velocity vector is split into different components. Each component can then be solved by using a more suitable method and hence provide improvement of accuracy and efficiency of the solution. An example is shown in Fig. 1.2, where the total velocity vector in the whole domain is decomposed into two components. The two components are calculated by a N-S solver and a potential-flow model respectively.

Kendon et al. (2003) decomposed the velocity vector into a invscid potential and a rotational component. The inviscid potential is calculated using BEM and the rotational component is solved by using a finite-element method. *Kim et al.* (2005) utilized the Helmholtz decomposition to derived the complementary RANS equations. The method has been applied to solve for 2D flow over a flat plate and a foil. *Hafez et al.* (2006) and *Hafez et al.* (2007) applied a similar decomposition and devised a modified Bernoulli's law. The method is used to solve for steady and transient flow over a foil section. *Ferrant et al.* (2008) use the SWENSE (Spectral Wave Explicit

Navier-Stokes Equations) approach and the velocity vector is decomposed into an incident and a residual (diffracted) component. Then a high-order spectral method is used to provide the incident wave and the modified Navier-stokes solver is used to solve for the diffracted component. Simulation with irregular waves has been conducted and it shows promising results. *Vukčević et al.* (2016a) and *Vukčević et al.* (2016b) combine the SWENSE approach and a implicit relaxation technique for solving naval hydrodynamic problems.

Generally, the advantage of velocity-decomposition methods over domain-decomposition methods is that the solution is continuous throughout the flow domain, and the difficulty related to the matching condition can be avoided. However, the velocity decomposition methods will usually result in new terms in the momentum equations, that require modification of the original Navier-Stokes solver and existing turbulence models. Moreover, most of the velocity-decomposition methods require all components of the decomposed velocity evaluated throughout the whole computational domain to solve the governing equations. For example, in the SWENSE method described in *Ferrant et al.* (2008), the incident-potential velocity must be evaluated for every cell in the computational domain so that the modified RANS equations can be solved to calculate the residual (diffracted) velocity. The above mentioned challenges diminish the advantage of reducing computational cost that the methods aim to achieve.

1.1.3 Previous Work at the University of Michigan

To circumvent both the difficulty related to the matching condition in domain decomposition methods and the challenges in velocity decomposition methods, *Edmund et al.* (2013) proposed a new formulation of the velocity decomposition method. The main difference that sets this method apart from other hybrid methods is the ability to fully describe the flow field external to the rotational region through a viscous potential which is solved by using a BEM on the physical body boundary.

One of the advantages of this approach over other velocity decomposition methods is that only the boundary condition is modified for each solver. The method is readily applied with current existing fluid flow solvers and turbulence models. Once the viscous potential is determined, it only requires the viscous-potential velocity to be evaluated on the inlet boundary of the flow field, which reduces computational expense.

Another advantage of this velocity decomposition method is that the field discretization required by the Navier-Stokes solver can be reduced to only encompass the rotational part of the flow field. The rotational-flow region only resides within a small volume around the body and the wake downstream of the body for our targeted applications, which are external incompressible flows with high Reynolds number, such as those appearing in naval and automobile engineering. On the other hand, the rest of the flow domain is irrotational and can be described by a scalar function (the viscous potential), which is solved by using a BEM.

In previous work, *Edmund* (2012a); *Edmund et al.* (2013); *Rosemurgy* (2014); *Rosemurgy et al.* (2016), the velocity decomposition method was developed to address steady flows of 2D, axis-symmetric, non-lifting bodies without water waves, or 2D bodies that can have lift and be near a water surface. For a detailed description of the previous development of our velocity decomposition approach, see *Edmund* (2012a); *Rosemurgy* (2014).

1.2 Research Objectives

The ultimate goal for our velocity decomposition method is to provide an accurate and efficient solver for numerical naval hydrodynamic simulations, which means there are still multiple elements of the velocity decomposition approach that need to be developed in addition to the previous work described in Section 1.1.3. The focus of this research is to solve 3D, unsteady flows with realistic geometries. No free-surface effect is considered in this thesis. Some of the important elements include:

- The development of a 3D viscous potential model,
    - Enable different discretization for BEM and FVM solvers
    - A robust algorithm to determine the extent of the vortical-flow region (essential for the solution of the viscous potential), for unstructured mesh and realistic geometries,
    - Stability analysis of the algorithm for solving the viscous potential
- The development of a tightly coupled algorithm for unsteady flow simulations,
- Application of the velocity decomposition approach to LES calculations,
- Appliction of the velocity decomposition approach to 3D, lifting flows.

This thesis addresses the above elements and is organized in the following manner. Chapter II presents the formulation of the Navier-Stokes problem of interest. The velocity decomposition for 3D, unsteady flows and the uniqueness of the viscous potential are discussed. The resultant viscous-potential and Navier-Stokes sub-problems are formulated. The viscous potential for lifting flows is also discussed.

In Chapter III, the numerical methods used to solve the Navier-Stokes and viscous-potential sub-problems are described. The various numerical aspects of the algorithm for the solution of the viscous potential are also discussed. The algorithm for the determination of the extent of the vortical-flow region, which is important for calculating the boundary condition for the viscous potential, is discussed in Section 3.2.2. The stability of the iterative algorithm used to solve the viscous potential is analyzed in Section 3.2.1.1. The coupling algorithm for unsteady flow simulations is presented in Section 3.2.3.

The velocity decomposition approach is tested on a set of canonical problems in Chapter IV. The difference between the inviscid potential and viscous potential is compared. The convergence of the iterative algorithm for solving the viscous potential is demonstrated. In Chapter V, the velocity decomposition approach is applied to perform LES calculations for flows around ship hulls. Preliminary results for the application of the velocity decomposition approach to 3D lifting flows are presented in Chapter VI. For all the test cases, the velocity decomposition solver is used to calculate the flow field within a greatly reduced domain. The reduced domain results are compared with the corresponding Navier-Stokes solution calculated in sufficiently large domains using a conventional solver.

In Chapter VII, the contributions of this thesis are summarized and the suggested directions for future research are highlighted.

CHAPTER II

Problem Formulation

The problems studied in this thesis are governed by the incompressible, 3D, Navier-Stokes equations. The formulation of the problem is first stated in Section 2.1. Then, the velocity decomposition for 3D, unsteady flows is developed in Section 2.2. The viscous potential is shown to be unique for a given velocity field with an appropriate boundary condition. The N-S subproblem and viscous potential subproblem, which result from the velocity decomposition, are formulated respectively in Section 2.3 and Section 2.4. The viscous potential for lifting flows is also discussed in Section 2.4.2.

2.1 The Navier-Stokes Problem

For the problem considered in this work, viscous effects, such as the boundary layer and viscous wake, are assumed to be confined within a the region V_{R} (see Fig. 2.1), around the body, and the fluid flow is irrotational for the rest of the domain. This is a suitable assumption for many high Reynolds number applications such as those with automobiles, trains, aerospace and marine vehicles, etc. The flow field of interest is governed by the incompressible Navier-Stokes equations, shown in Eq. (2.1) and

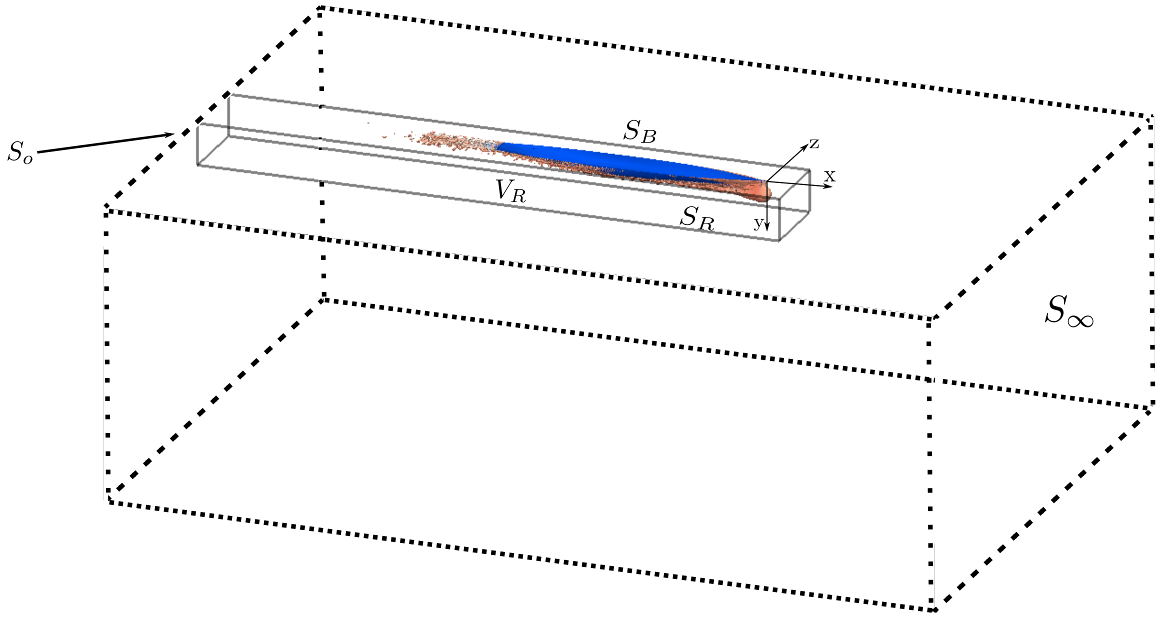


Figure 2.1: The schematic of the computational domain

Eq. (2.2).

$$\nabla \cdot \mathbf{u} = 0 \tag{2.1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} \tag{2.2}$$

where $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ (or $u_i = u_i(x_i, t)$) is the total velocity vector field that varies in space and time (For convenience, both the subscript notation and the vector notations are used interchangeably in this thesis.), $p = p(\mathbf{x}, t)$ is the pressure, ρ is the density of the fluid and ν is the kinematic viscosity. No free-surface effect is considered in this thesis.

The schematic of the computational domain is shown in Fig. 2.1. A body-fixed coordinate system is used and the position vector is $\mathbf{x} = x\ \hat{\mathbf{i}} + y\ \hat{\mathbf{j}} + z\ \hat{\mathbf{k}}$. In this thesis, only problems with a fixed body are considered. On the body boundary S_{B}, the no-slip boundary condition, Eq. (2.3), is enforced.

$$\mathbf{u} = 0 \qquad \text{on } S_{\mathrm{B}} \tag{2.3}$$

The disturbance due to the presence of the body disappears far away from the body, so the far-field boundary condition, Eq. (2.4), is applied on S_∞ (U_∞ is the free-stream velocity).

$$\lim_{|x|\to\infty} \mathbf{u} = U_\infty \, \hat{\mathbf{i}} \tag{2.4}$$

The initial condition is prescribed through an initial velocity field $\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x})$, where $\mathbf{u}_0$ is the initial velocity field.

Throughout this paper, the problem described above is designated as the *Navier-Stokes (N-S) problem*, which consists of the equations, domain and the boundary conditions. The *Navier-Stokes problem* is also used to refer to the equations, domain, and boundary conditions for the varied form of governing equations the turbulence is modeled through either the Unsteady-Reynolds-Averaged-Navier-Stokes (URANS) equations and Large-Eddy-Simulation (LES) Navier-Stokes equations.

2.1.1 URANS Equations

The Reynolds number is defined as $Re = U_\infty L/\nu$, where L is the length of the body. When Reynolds number of the flow is sufficiently high, the flow is turbulent. Turbulent flow is typically modeled through two different ways which lead to two different variations of the N-S equations, *i.e.* URANS equations and LES equations.

To derive the URANS equations, the flow variables are expressed as the sum of a mean component due to the ensemble average, Eq. (2.5), of the flow and a fluctuating component due to the turbulence denoted by the superscript $'$, Eq. (2.6).

$$\overline{f}(\mathbf{x}, t) = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f_n(\mathbf{x}, t) \tag{2.5}$$

$$\begin{aligned} \mathbf{u} &= \overline{\mathbf{u}} + \mathbf{u}' \qquad (2.6) \\ p &= \overline{p} + p' \end{aligned}$$

After applying Eq. (2.6) to Eq. (2.1) and Eq. (2.2), the URANS can be expressed as

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0 \tag{2.7}$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\bar{u}_i \bar{u}_j) = -\frac{\nabla \overline{p}}{\rho} + \frac{\partial}{\partial x_j}\left(\nu\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right) + \tau_{ij}\right) \tag{2.8}$$

$$\tau_{ij} = -\overline{u_i' u_j'} \tag{2.9}$$

The quadratic nonlinearity of the convective term (the root of all "evil" *Doering and Gibbon* (1995)) in the N-S equations leads to the extra term on the right hand side, which often referred to as the Reynolds-stress tensor τ_{ji}. It represents the loss of momentum in the fluid due to the turbulent fluctuations.

The components of the Reynolds-stress tensor are the additional unknowns and require some means to determine them. Using the Boussinesq eddy viscosity approximation, the deviatoric part of the Reynolds stress term is assumed to be proportional to the mean rate of strain Eq. (2.10),

$$-\overline{u_i' u_j'} + \frac{2}{3}k\delta_{ij} = \nu_t\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) \tag{2.10}$$

where ν_t is the kinematic turbulent viscosity (also referred as kinematic eddy viscosity), $k = \frac{1}{2}\overline{u_k' u_k'}$ is the turbulent kinetic energy, δ_{ij} is the Kronecker delta. The term with the turbulent kinetic energy k is absorbed into the pressure term in Eq. (2.8) and the pressure becomes a modified pressure that incorporates the turbulent fluctuation. Then URANS equations can be expressed as:

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0 \tag{2.11}$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\bar{u}_i \bar{u}_j) = -\frac{\nabla \overline{p}}{\rho} + \frac{\partial}{\partial x_j}\left((\nu + \nu_t)\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right)\right) \tag{2.12}$$

The k-ω SST turbulence model is used for the URANS equations in this research. More details are presented in Chapter III. For a detailed discussion of turbulence modeling, the readers are referred to *Wilcox* (1998).

2.1.2 LES Equations

For LES calculations, the underlying premise is that in high-Re number flows, the large-scale of the turbulent structures contain most of the energy and must be directly computed. The small-scale turbulence has a smaller contribution. Moreover, the small-scale turbulence is assumed to be statistically isotopic and has universal characteristics, based on Kolmogorov's hypothesis, *Pope* (2001). So it is acceptable and more amenable to model the small-scale turbulence.

A spatial filter is defined in Eq. (2.13), where the flow variables can be separated into a resolved large-scale component, indicated with "$^-$", and a subgrid-scale component, indicated with "$'$", Eq. (2.14).

$$\bar{f}(\mathbf{x},t) = \int\int\int G(\mathbf{x}-\xi;\Delta)f(\mathbf{x},t)d\xi \tag{2.13}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}} + u' \tag{2.14}$$

$$p = \bar{p} + p'$$

In Eq. (2.13), G represents the filter function, Δ represents the filter width, which is usually related to the grid size.

The filtering defines a local spatial averaging process and separates the resolved scale from the subgrid scales. The LES equations can be derived by filtering the N-S equations, Eq. (2.2). Then the LES equations can be expressed as in, Eq. (2.15) and Eq. (2.16), where the nonlinearity of the convection term produces again an extra term.

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \tag{2.15}$$

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\bar{u}_i \bar{u}_j) = -\frac{\nabla \bar{p}}{\rho} + \frac{\partial}{\partial x_j}\left(\nu \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) + \tau_{ij}\right) \tag{2.16}$$

In Eq. (2.16), the extra term τ_{ij}, is referred to as the subgrid-scale tensor in the LES framework.

The sugrid-scale tensor can be expressed as the decomposition Eq. (2.17). It can be further decomposed as the sum of the Leonard stress tensor L_{ij}, the cross-stress tensor C_{ij} and the subgrid-scale Reynolds stress tensor R_{ij}, Eq. (2.17). The Leonard-stress tensor represents the interaction of the resolved scales. The cross-stress tensor contains the cross interaction between the resolved scales and the subgrid scales. The Reynolds-stress tensor describes the interaction of the subgrid scales.

$$\begin{aligned} \tau_{ij} &= \overline{u_i u_j} - \bar{u}_i \bar{u}_j \\ &= L_{ij} + C_{ij} + R_{ij} \\ &= (\overline{\bar{u}_i \bar{u}_j} - \bar{u}_i \bar{u}_j) + (\overline{\bar{u}_i u'_j} + \overline{u'_i \bar{u}_j}) + \overline{u'_i u'_j} \end{aligned} \tag{2.17}$$

The subgrid-scale model used in this thesis is the one-equation eddy viscosity model. More discussion can be found in Chapter III. For a detailed discussion of LES calculation for incompressible flows, see *Sagaut* (2006).

2.2 Decomposition of the Navier-Stokes Problem

In this section, the velocity decomposition for 3D, unsteady flows is developed. The decomposition is unique after showing that the viscous potential is uniquely defined for a given velocity field with an appropriate boundary condition.

After applying the decomposition, the velocity vector is decomposed into two

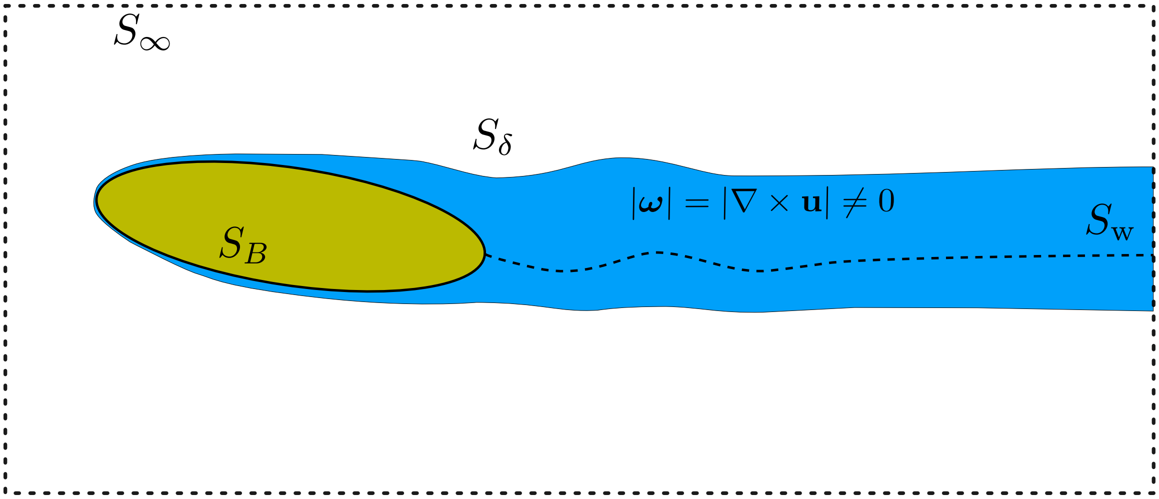


Figure 2.2: sketch of the flow field and vortical region

components,

$$\mathbf{u}(\mathbf{x}, t) = \nabla\varphi(\mathbf{x}, t) + \mathbf{w}(\mathbf{x}, t) \tag{2.18}$$

where $\nabla\varphi$ and $\mathbf{w}$ are the viscous-potential velocity and the vortical velocity respectively. All three velocities are assumed to vary in time but not necessarily with the same time scale. The time dependence of each term in Eq. (2.18) is discussed in more detail in Section 3.2.3. All three components of the velocity are divergence free.

In viscous flows, the vorticity ω vanishes exponentially at infinity *Batchelor* (2000). Hence the irrotational region defined in this work is where the vorticity is negligible for all practical purposes, *Morino et al.* (2007). This means the vortical region denotes the complement of the irrotational region. A sketch demonstrating the vortical region around the body and wake is shown in Fig. 2.2. The vorticity thickness, (i.e. the body-normal distance away from the body boundary where the flow becomes practically irrotational) is denoted as δ. The illustration of δ and S_δ can be found in a zoom-in sketch of Fig. 2.2, shown in Fig. 2.3.

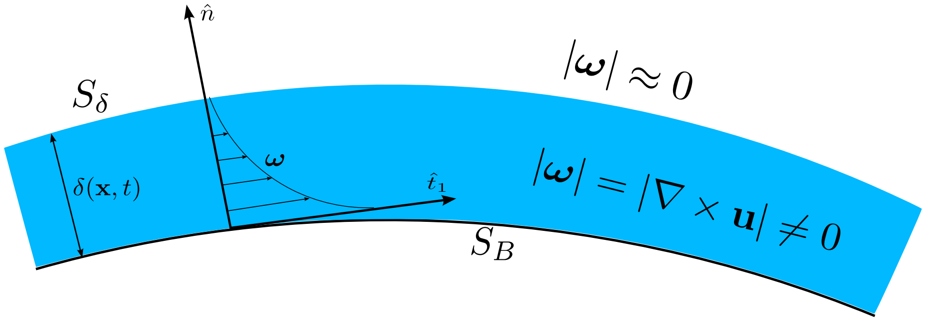

$\hat{n}$
S_δ
$|\boldsymbol{\omega}| \approx 0$
$\delta(\mathbf{x},t)$
$\boldsymbol{\omega}$
$\hat{t}_1$
$|\boldsymbol{\omega}| = |\nabla \times \mathbf{u}| \neq 0$
S_B

Figure 2.3: $\delta(\mathbf{x},t)$ defined in local-orthogonal coordinate

$$\mathbf{w} = 0 \qquad \text{for } |\mathbf{x}| \geq \delta \tag{2.19}$$

$$\nabla\varphi = \mathbf{u} \qquad \text{for } |\mathbf{x}| \geq \delta \tag{2.20}$$

The velocity decomposition is not uniquely defined without proper boundary conditions Morino (1986). Various forms can be devised for different calculation purposes Morino (1990). In this work we seek a decomposition such that the vortical velocity $\mathbf{w}$ is zero outside the rotational region of the flow field, Eq. (2.19) and Eq. (2.20). Then a Neumann boundary condition on S_δ is uniquely defined as in Eq. (2.21) due to Eq. (2.20).

$$\frac{\partial\varphi}{\partial n} = u_n \qquad \text{on } S_\delta \tag{2.21}$$

We also have the divergence and curl of the viscous potential specified Eq. (2.22) and Eq. (2.23).

$$\nabla\cdot\nabla\varphi = \nabla^2\varphi = 0 \tag{2.22}$$

$$\nabla\times\nabla\varphi = 0 \tag{2.23}$$

The vector field, $\nabla\varphi$, is uniquely defined according to Helmholtz theorem. The proof can be conducted as follow. Assuming there are two potential functions φ_1, φ_2 that satisfy the above defined problem (Eq. (2.21) - Eq. (2.23)), the difference between the two solutions is φ_d. The difference of the velocity field that these two potential admit is $\nabla\varphi_d = \mathbf{u}_d$.

$$\int_V \mathbf{u}_d \cdot \mathbf{u}_d dV = \int_V \nabla\varphi_d \cdot \nabla\varphi_d dV \tag{2.24}$$

$$= \int_V \nabla \cdot (\varphi_d \nabla\varphi_d)\, dV \tag{2.25}$$

$$= \int_{S_\infty + S_\mathrm{B}} \varphi_d \cancelto{0}{\frac{\partial\varphi_d}{\partial n}} dS = 0 \tag{2.26}$$

The total difference of the two potential velocity fields through the whole flow domain, which has the boundaries S_∞ and S_B, is expressed as in Eq. (2.24). After manipulation and using the divergence-free condition, we can arrive at Eq. (2.26). Because the two potential satisfy the same boundary conditions, $\partial_n\varphi_d = 0$ on both S_∞ and S_B. Then the difference of the velocity field is zero everywhere in the flow field and the viscous potential is uniquely defined. Hence, the decomposition is uniquely defined for a given velocity field.

2.3 Navier-Stokes Sub-problem

The Navier-Stokes problem is decomposed into a Navier-Stokes sub-problem and a viscous potential sub-problem by applying the decomposition stated in Eq. (2.18). The Navier-Stokes sub-problem is defined similarly as the Navier-Stokes problem, except that it is defined within a greatly reduced domain, V_R, Fig. 2.4. The boundary of the reduced domain, S_R, is chosen to encompass the rotational region of the flow field. The velocity boundary condition Eq. (2.27) on the reduced domain boundary

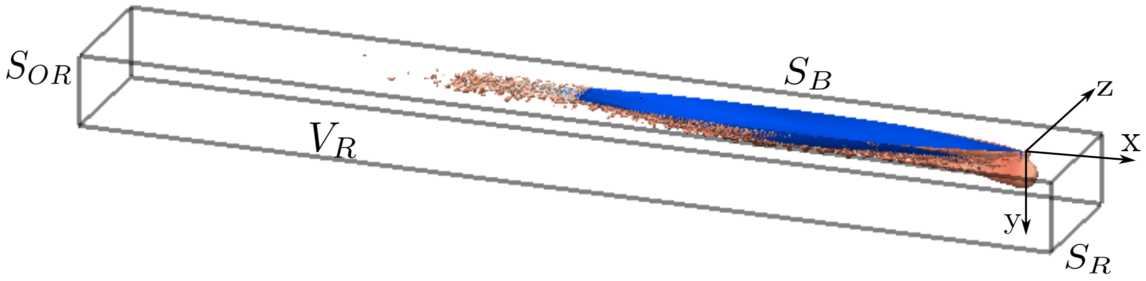


Figure 2.4: The schematic of the computational domain

is prescribed through the viscous-potential velocity.

$$\mathbf{u}(\mathbf{x}, t) = \nabla\varphi(\mathbf{x}, t) \qquad \text{on } S_{\mathrm{R}} \tag{2.27}$$

The total velocity vector $\mathbf{u}$ is calculated by solving the Navier-Stokes sub-problem together with a BVP (Boundary-Value Problem) that governs φ.

2.4 Viscous Potential Sub-problem

The unknown variable we want to solve in the viscous potential sub-problem is the viscous potential φ. The domain for the viscous potential sub-problem is shown in Fig. 2.5. The governing equation is the Laplace equation, Eq. (2.28), the far-field boundary condition is Eq. (2.29) and the viscous boundary condition is Eq. (2.30), which is derived in Section 2.4.1.

$$\nabla^2\varphi = 0 \tag{2.28}$$

$$\lim_{|\mathbf{x}|\to\infty} \nabla\varphi = -U_\infty\ \hat{\mathbf{i}} \tag{2.29}$$

$$\frac{\partial \varphi}{\partial n} = \chi \qquad\qquad \text{on } S_B \text{ and } S_w \tag{2.30}$$

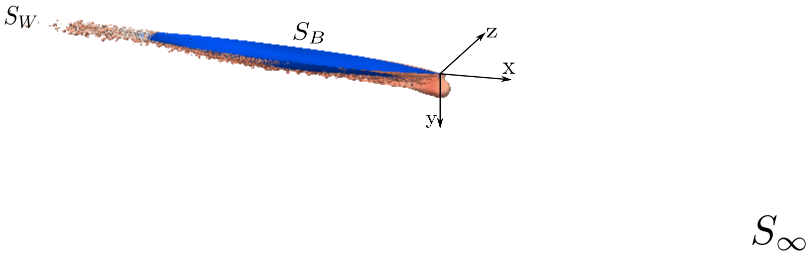


Figure 2.5: The schematic of the computational domain

2.4.1 Viscous Boundary Condition for the Viscous Potential

Based the velocity decomposition discussed in Section 2.2, the other boundary condition that needs to be satisfied is Eq. (2.21). As the flow field evolves, the shape and volume of the vortical region changes, $S_\delta = S_\delta(\mathbf{x}, t)$. It is computationally expensive to re-discretize the boundary S_δ, and reformulate the coefficients in the discrete equations. Hence, a transpiration-velocity type of technique similar to the idea of *Lighthill* (1958) is used in our velocity decomposition approach. The body surface S_B and a wake surface S_w are chosen to be the boundaries. The wake surface S_w is directly behind the body as shown in Fig. 2.5 and is used to capture the viscous effects at the wake. Then, it is necessary to derive boundary conditions on S_B and S_w that are equivalent to the boundary condition Eq. (2.21).

Based on the no-slip boundary condition on S_B, Eq. (2.32) can be derived

$$\mathbf{u} = \nabla\varphi + \mathrm{w} = 0 \qquad\qquad \text{on } S_\mathrm{B}, \tag{2.31}$$

which can be rearranged as:

$$\nabla\varphi = -\mathbf{w}. \tag{2.32}$$

Then, a Neumann boundary condition Eq. (2.33) is obtained.

$$\frac{\partial \varphi}{\partial n} = -w_{\mathrm{n}} \qquad \text{on } S_{\mathrm{B}} \tag{2.33}$$

Integrating the divergence-free relation for $\mathbf{w}$ in the local-normal direction Eq. (2.34) and combining it with the Neumann boundary condition Eq. (2.33), a new Neumann boundary condition Eq. (2.35) can be derived,

$$\begin{aligned} &\nabla \cdot \mathbf{w} = 0 \\ &\int_0^\delta \frac{\partial w_n}{\partial n} + \frac{\partial w_{t1}}{\partial t1} + \frac{\partial w_{t2}}{\partial t2} dn = 0 \\ &w_n(\mathbf{x}_b) = \int_0^\delta \left(\frac{\partial w_{t_1}}{\partial t_1} + \frac{\partial w_{t_2}}{\partial t_2} \right) \mathrm{d}n - w_n(\mathbf{x}_\delta) \end{aligned} \tag{2.34}$$

$$\begin{aligned} \frac{\partial \varphi}{\partial n} &= -\mathbf{w} \cdot \hat{n} \qquad \text{on } S_{\mathrm{B}} \\ &= -w_n(\mathbf{x}_b) \\ &= -\int_0^\delta \left(\frac{\partial w_{t_1}}{\partial t_1} + \frac{\partial w_{t_2}}{\partial t_2} \right) \mathrm{d}n - \cancelto{0}{w_n(\mathbf{x}_\delta)} = \chi \end{aligned} \tag{2.35}$$

where subscripts t_1 and t_2 represent the local tangential directions, $\mathbf{x}_b$ and $\mathbf{x}_\delta$ are points on the boundaries S_B and S_δ respectively. A similar boundary condition is also developed in *Morino* (1986). This Neumann boundary condition Eq. (2.35) relates the vortical velocity and the transpiration velocity at the boundary introduced by the viscous potential. It can be viewed as a generalization of the equivalent-source approach in *Lighthill* (1958). Without assuming the boundary layer thickness being small, this boundary condition is applicable for separated flows. More discussion of the boundary condition for the viscous potential can be found in Section 3.2.

In Eq. (2.35), $\mathbf{w}$ is not known before the correct viscous-potential velocity is calculated. An iterative boundary condition for the solution of the viscous potential is developed in Section 3.2.1. This non-homogeneous Neumann boundary condition and its varied forms are referred to as the viscous boundary condition throughout this thesis. It is worth noting that the viscous potential which satisfies this viscous boundary condition is different than the inviscid potential which satisfies the conventional non-penetration boundary condition. Through the viscous boundary condition, the viscous potential is able to capture the viscous effects, even in the case of massive separation, *Edmund et al.* (2013); *Chen et al.* (2015), (e.g. flow around a circular cylinder).

To capture the viscous effects at the wake, a viscous boundary condition for the wake surface S_w is also needed. Instead of the no-slip boundary condition, a relation Eq. (2.37) can be found for the wake surface boundary condition,

$$\begin{aligned} \mathbf{u} &= \mathbf{u}(\mathbf{x}_\mathrm{w}) \qquad\qquad \text{on } S_\mathrm{W} \\ &= \nabla\varphi + \mathrm{w} \end{aligned} \tag{2.36}$$

which can be rearranged as

$$\nabla\varphi = \mathbf{u}(\mathbf{x}_\mathrm{w}) - \mathbf{w}, \tag{2.37}$$

where $\mathbf{x}_\mathrm{w}$ represents the points on S_w and $\mathbf{u}(\mathbf{x}_\mathrm{w})$ represents the velocity at the location of the wake surface.

Similar to the derivation of the viscous boundary condition for the body surface, a viscous boundary condition for the wake surface can be derived Eq. (2.38).

$$\begin{aligned} \frac{\partial\varphi}{\partial n} &= u_n(\mathbf{x}_\mathrm{w}) - w_\mathrm{n}(\mathbf{x}_\mathrm{w}) \qquad\qquad \text{on } S_\mathrm{w} \\ &= u_n(\mathbf{x}_\mathrm{w}) - \int_0^\delta \left(\frac{\partial w_{t_1}}{\partial t_1} + \frac{\partial w_{t_2}}{\partial t_2} \right) \mathrm{d}n - \cancelto{0}{w_n(\mathbf{x}_\delta)} = \chi \end{aligned} \tag{2.38}$$

Although the time variable is not explicitly present in the Laplace governing equation, the unsteadiness is implicitly included through the viscous boundary condition because δ and $\mathbf{w}$ are both functions of time. The time dependency may be due to the unsteadiness from the body motion, or the flow unsteadiness due to turbulent or transitional flow in the boundary layer and viscous wake. Based on the characteristics of the unsteadiness, different unsteady algorithms can be developed *Chen et al.* (2015). The unsteady algorithm has been successfully applied to solve 2D flow over a cylinder at $Re = 140$. As shown in Fig. 2.6 (a), the Von Kármán vortex street is present. In Fig. 2.6, "VD" denoted the solutions by using the velocity-decomposition-coupled solver with a reduced domain, "NSL" and "NSR" represent the solutions by using a conventional FVM solver with a large and reduced domain respectively. As can been seen, the "VD" solutions match well with the "NSL" solutions, while the "NSR" solutions fail to do so. More details can be found in *Chen* (2014) and *Chen et al.* (2015). The unsteady algorithm used in this work is discussed in Section 3.2.3

2.4.2 Viscous Potential for Lifting Flows

The Kutta condition is needed for a potential flow formulation to model lifting flow correctly and to avoid the D'Alembert's paradox. In general, the Kutta condition is a condition that is imposed mostly based on physical consideration to allow the fluid to leave the trailing edge of the body in a physical manner.

It is well known that for 2D potential flows with a closed body, the domain is not simply connected and the solution is not unique. The wake surface or barrier w can be inserted in the domain and the cirulation Γ can be specified to uniquely defined a solution, as shown in Eq. (2.39), where $\mathbf{u}_\mathrm{d} = \mathbf{u}_1 - \mathbf{u}_2$ and $\phi_\mathrm{d} = \phi_1 - \phi_2$, the subscripts "1" and "2" denote the different solutions that satisfy the problem. S_W^+ and S_W^- are each side of the wake surface respectively, as shown in Fig. 2.7.

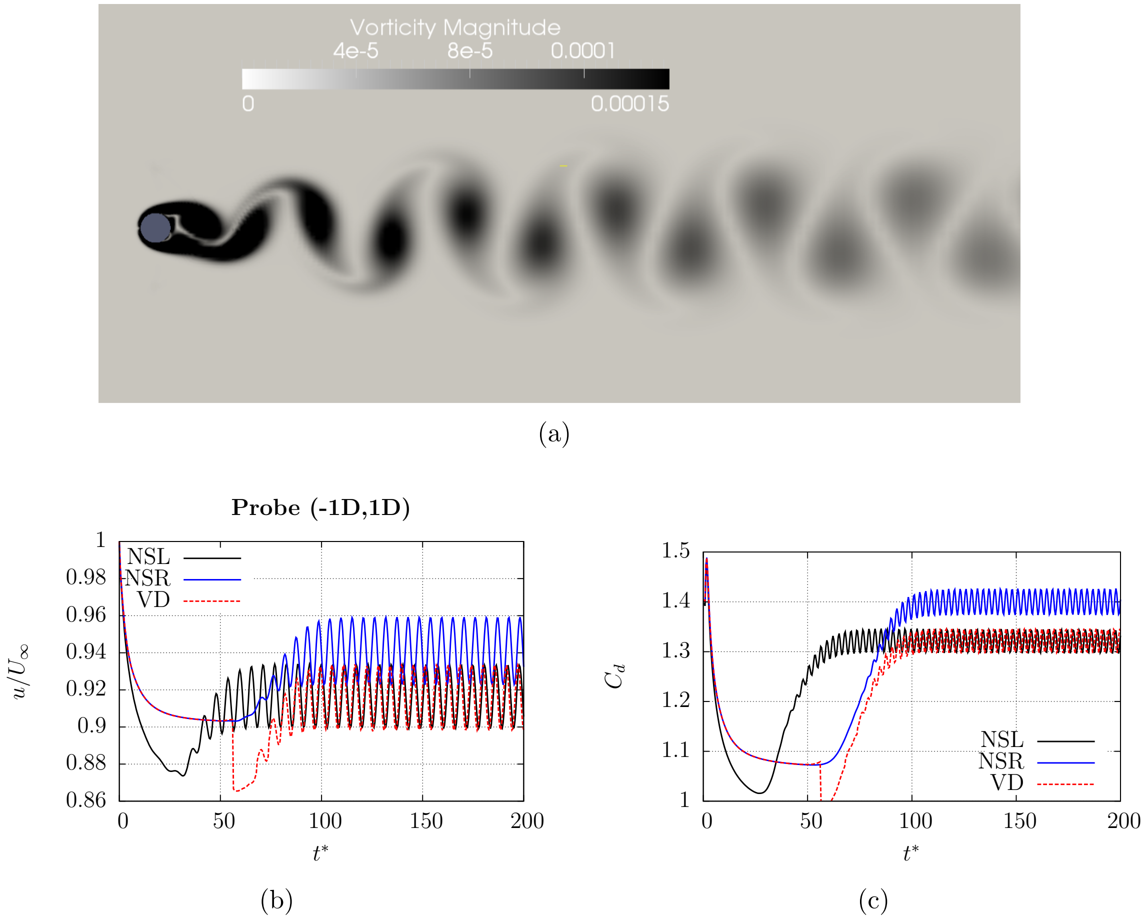


Figure 2.6: Solutions for 2D flow over a circular cylinder at $Re = 140$: (a) contour of vorticity magnitude (b) stream-wise velocity time-history comparison (c) drag-coefficient comparison

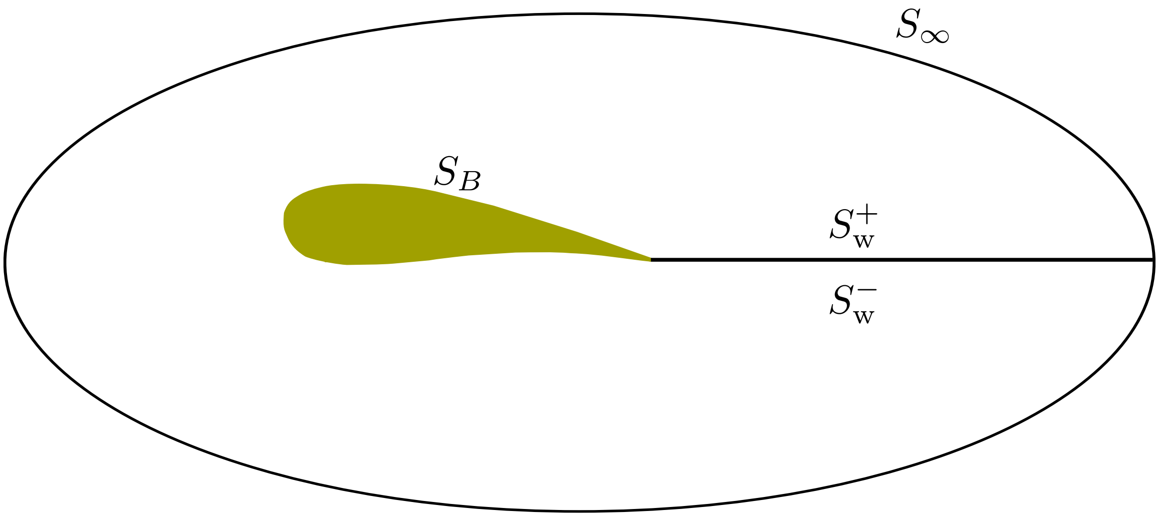


Figure 2.7: Doubly connected region exterior to an 2D airfoil

$$
\begin{aligned}
\int_V \mathbf{u}_\mathrm{d}\cdot\mathbf{u}_\mathrm{d}\,dV &= \int_V \nabla\phi_d\cdot\nabla\phi_d\,dV \qquad (2.39)\\
&= \int_V \nabla\cdot\left(\phi_d\nabla\phi_d\right)dV\\
&= \int_{S_\infty+S_\mathrm{B}} \phi_d \cancelto{0}{\frac{\partial\phi_d}{\partial n}} dS + \int_{S_\mathrm{W}^+} \phi_d \frac{\partial\phi_d}{\partial n} dS - \int_{S_\mathrm{W}^-} \phi_d \frac{\partial\phi_d}{\partial n} dS = 0\\
&= \int_{S_\mathrm{W}^+ + S_\mathrm{W}^-} \left(\phi_d^+ - \phi_d^-\right)\frac{\partial\phi_d}{\partial n} dS = 0\\
&= \int_{S_\mathrm{W}^+ + S_\mathrm{W}^-} \left(\phi_1^+ - \phi_2^+ - \phi_1^- + \phi_2^-\right)\frac{\partial\phi_d}{\partial n} dS = 0\\
&= \int_{S_\mathrm{W}^+ + S_\mathrm{W}^-} \left(\Gamma_1 - \Gamma_2\right)\frac{\partial\phi_d}{\partial n} dS = 0
\end{aligned}
$$

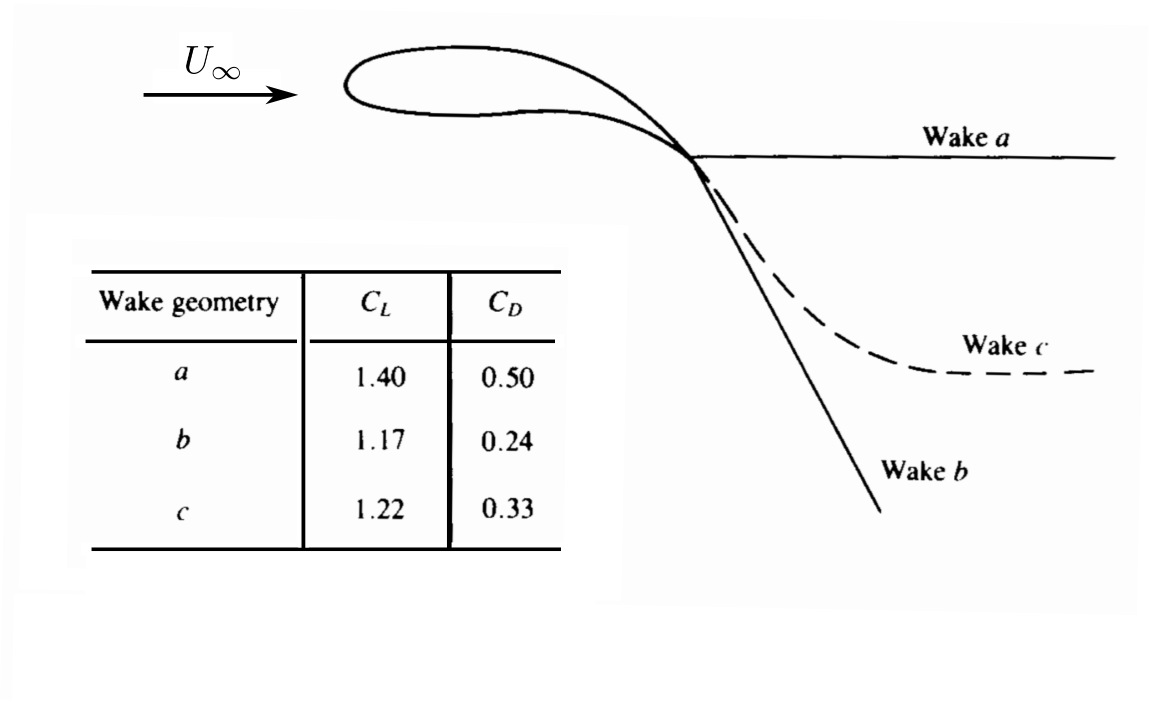
In *Rosemurgy* (2014), a circulation condition is developed for the viscous potential to solve 2D steady lifting flows. The circulation condition is to match the circulation produced by the viscous potential with the one obtained in the N-S solution. By using this circulation condition, the velocity decomposition solver is able to solve 2-D lifting flows accurately and efficiently.

Alternatively, another way to find a unique solution is to specify a Neumann condition at the wake surface through the velocity there. Hence $\partial_n \phi_d = 0$. The Neumann condition at the wake surface can be calculated from the solution of the N-S subproblem.

In 3D flows, it is not clear an equivalent circulation condition similar to the one in *Rosemurgy* (2014) can be derived. Most of the research on the condition for the wake surface in 3D lifting flows consider the wake surface stems from the trailing edge of the body and define it as a stream surface. Then various conditions regarding the potential jump or normal velocity can then be developed. *Hess* (1974) discussed different types of Kutta conditions that can be developed and compared the solutions calculated using each of the Kutta conditions. *Morino and Bernardini* (2001) and *Bassanini et al.* (1999) present the Kutta condition as a means to remove the edge singularity that is due to the discontinuity of the surface gradient. *Katz and Plotkin* (2001) demonstrated the effects of the wake surface alignment on the solutions. As shown in Fig. 2.8, the lift and drag coefficients are clearly different when using different wake geometry. The solution by using wake c is closest to the experimental measurements. So having the wake surface properly aligned is important for accurate solutions.

Most research consider a foil like body that has a sharp trailing edge. As pointed out in *Hess* (1974), for a 3D body that has continuous curvature, it is not clear how the flow leaves the trailing edge of the body before a solution is computed. When the flow is separated and the viscous wake thickened, the zero-thickness wake surface in conventional potential flow model is no longer valid to correctly represented the flow field downstream of the body.

For the viscous potential consider in this work, information throughout the flow field, including at the wake surface, can be obtained from the solution of the N-S subproblem, since the two subproblems are coupled. Then a boundary condition, that correctly represent the physics, can be imposed at the wake surface in the form of



Wake geometry	C_L	C_D
a	1.40	0.50
b	1.17	0.24
c	1.22	0.33

Figure 2.8: Effect of wake geometry (adopted from FIGURE 9.6 in *Katz and Plotkin* (2001))

Eq. (2.38). It is worth noting that the wake surface for the viscous potential does not have to be a stream surface, unlike the conventional wake surfaces *e.g.* *Morino and Bernardini* (2001). The body surface and the wake surface introduce a transpiration velocity to model the presence of the vortical region. More on this is discussed in Section 3.2.1.

CHAPTER III

Numerical Implementation

In this chapter, the numerical schemes used to solve the N-S problem and the viscous-potential sub-problems are described. Then, the numerical implementation for different components in the velocity decomposition solver is discussed.

3.1 Solution Method for the Navier-Stokes Problems

The turbulence models used for URANS and LES calculations are first described. Then the numerical schemes used to discretized the N-S equations and the transport equations in the turbulence model is stated. The initial and boundary conditions are also presented.

3.1.1 Turbulence Modeling

3.1.1.1 URANS

As discussed in Section 2.1.1, the Reynolds stress term needs to be modeled. The applicability of the velocity decomposition solver is independent of the turbulence model. In this research, k-ω SST model (*Menter* (1994),*Menter and Esch* (2001)) is used for all URANS calculations.

The k-ω SST model is based on a Baseline model, which is essentially blending the k-ω and k-ϵ using a blending function, (the overbar is dropped from the averaged quantities to simplify the notation)

$$\frac{\partial k}{\partial t}+\frac{\partial}{\partial x_j}\left(u_j k\right)=\frac{\partial}{\partial x_j}\left(\Gamma_k \frac{\partial k}{\partial x_j}\right)+Q_k \tag{3.1}$$

$$Q_k=\min\left(\mathrm{P_k},\ \mathrm{c_1}\beta^*\mathrm{k}\omega\right)-\beta^*\omega\mathrm{k}$$

$$\frac{\partial \omega}{\partial t}+\frac{\partial}{\partial x_j}\left(u_j \omega\right)=\frac{\partial}{\partial x_j}\left(\Gamma_\omega \frac{\partial \omega}{\partial x_j}\right)+Q_\omega \tag{3.2}$$

$$Q_\omega=\frac{\gamma}{\nu_t}P_k-\beta\omega^2-(F_1-1)2\alpha_\omega\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}$$

$$\tag{3.3}$$

where

$$\Gamma_k=\nu+\alpha_k\nu_t,\ \Gamma_\omega=\nu+\alpha_\omega\nu_t,\ P_k=\tau_{ij}\frac{\partial u_i}{\partial x_j} \tag{3.4}$$

The coefficients ϕ of the model are blended using F_1, $\phi=F_1(\phi_1-\phi_2)+\phi_2$, where $\phi_1,\ \phi_2$ are the coefficients of the $k-\omega$ and the $k-\epsilon$ model respectively:

$$\alpha_{k1}=0.85034,\ \alpha_{\omega1}=0.5,\ \gamma_1=0.5532,\ \beta_1=0.075,\ \beta^*=0.09,\ c_1=10 \tag{3.5}$$

$$\alpha_{k2}=0.85034,\ \alpha_{\omega2}=0.5,\ \gamma_2=0.5532,\ \beta_2=0.075$$

$$F_1=\tanh(\arg_1^4);\ \arg_1=\min\left(\max\left(\frac{\sqrt{\mathrm{k}}}{\beta^*\omega\mathrm{y}};\ \frac{500\nu}{\mathrm{y}^2\omega};\right)\ \frac{4\alpha_{\omega_2}\mathrm{k}}{\mathrm{CD_{k\omega}y^2}}\right) \tag{3.6}$$

$$F_2=\tanh(\arg_2^2);\ \arg_2=\max\left(2\frac{\sqrt{\mathrm{k}}}{\beta^*\omega\mathrm{y}};\ \frac{500\nu}{\mathrm{y}^2\omega}\right);\ \mathrm{CD_{k\omega}}=\max\left(2\alpha_{\omega2}\frac{1}{\omega}\frac{\partial\mathrm{k}}{\partial\mathrm{x_j}}\frac{\partial\omega}{\partial\mathrm{x_j}};\ 10^{-10}\right)$$

where y is the distance to wall.

By doing the blending, the baseline model can take advantage of the free-stream independence of the $k-\epsilon$ model in the outer part of the boundary layer and retain the robust and accurate formulation of the Wilcox $k-\omega$ model in the near wall region.

Furthermore, to make the model handle strong adverse pressure gradients flows, the kinematic turbulent viscosity is defined by blending two expressions Eq. (3.7)

$$\nu_t = \frac{a_1 k}{max\left(a_1\omega, \sqrt{2}\,|S|\,F_2\right)} \tag{3.7}$$

where $a_1 = 0.31$, $|S| = \left|\frac{1}{2}(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i})\right|$ is the magnitude of the mean strain rate tensor and F_2 is a blending function. The first part of the expression for the kinematic turbulent viscosity $\nu_t = \frac{k}{\omega}$ is from the conventional $k - \omega$ turbulence model. The second part $\nu_t = \frac{a_1 k}{\sqrt{2}|S|F_2}$is based on Bradshaw's assumption, which states that the shear stress in a boundary layer is proportional to the turbulent kinetic energy k. By relating the shear stress to turbulent kinetic energy k, the shear stress also satisfies a transport equation, and hence the name "SST" (Shear-Stress-Transport). For a detailed discussion of turbulence model for URANS caculation, the readers are referred to a text related to turbulence modeling for CFD, such as *Wilcox* (1998) and *Moukalled et al.* (2015).

3.1.1.2 LES

The subgrid-scale stress term is modeled using the eddy viscosity approximation, similar to the URANS turbulent model. The deviatoric part of the subgrid-scale stress tensor is proportional to the resolved strain rate tensor.

$$\tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = \nu_{\text{sgs}}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) \tag{3.8}$$

where ν_{sgs} is the subgrid-scale turbulent kinematic viscosity. The subgrid-scale turbulent kinematic viscosity is related to the subgrid-scale turbulent kinetic energy by Eq. (3.9).

$$\nu_{\text{sgs}} = C_k \bar{\Delta} k_{\text{sgs}}^{\frac{1}{2}} \tag{3.9}$$

where $\bar{\Delta}$ is the characteristic filter width, which is calculated as the cube-root of the cell volume in this work. $C_k = 0.094$ and $k_{sgs} = \frac{1}{2}(\overline{u_i u_i} - \bar{u}_i \bar{u}_i)$ is the subgrid-scale kinetic energy. Then the subgrid-scale turbulent kinetic energy k_{sgs} is governed by a transport equation. For detail of the subgrid-scale turbulence model, see *Kim and Menon* (1995).

$$\frac{\partial k_{sgs}}{\partial t} + \frac{\partial}{\partial x_j}\left(\bar{u}_j k_{sgs}\right) = \frac{\partial}{\partial x_j}\left((\nu + \nu_{\text{sgs}})\frac{\partial k}{\partial x_j}\right) + \tau_{ij}\frac{\partial \bar{u}_i}{\partial x_j} - C_\epsilon \frac{k_{\text{sgs}}^{0.5}}{\bar{\Delta}} \tag{3.10}$$

where $C_\epsilon = 1.05$. When stated, a dynamic procedure proposed *Germano et al.* (1991) is used to provide local approximation for the constants C_k and C_ϵ in the turbulent model.

3.1.2 Initial and Boundary Conditions

For external flows, the N-S problem is defined on an infinite domain, which is impossible to discretize Generally in applications, the far-field boundary S_∞ of the flow domain is placed far away from the body to minimize any blockage effect that can be generated by the necessarily finite numerical domain. The influence of the domain size is studied in Chapter IV. For computational purposes, the far-field boundary S_∞ is separated into two boundaries, the inlet boundary S_{I} and the outlet boundary S_{O}. Boundary conditions Eq. (3.11) and Eq. (3.12) for velocity and pressure are applied at the inlet and outlet boundary.

$$\mathbf{u} = \mathbf{U}_\infty, \quad \frac{\partial p}{\partial n} = 0 \qquad \text{on } S_I \tag{3.11}$$

$$\frac{\partial \mathbf{u}}{\partial n} = 0, \qquad p = 0 \qquad \text{on } S_{\text{O}} \tag{3.12}$$

In the Navier-Stokes sub-problem, the inlet boundary of the computational domain is very close to the body, and the free-stream inlet boundary condition Eq. (3.11) is not valid. The viscous-potential velocity is used as the velocity boundary condition

Eq. (2.27). The Neumann boundary condition for pressure can also be calculated from the viscous-potential velocity by applying the Bernoulli's equation and a two-point finite difference approximation Eq. (3.13).

$$\frac{\partial p}{\partial n} = \frac{\rho}{2}\frac{|\nabla\varphi(\mathbf{x}_f)|^2 - |\nabla\varphi(\mathbf{x}_f - \epsilon \cdot \hat{n})|^2}{\epsilon} \qquad \text{on } S_{\mathrm{R}} \tag{3.13}$$

$\mathbf{x}_{\mathrm{f}}$ is a cell face center at the boundary, ϵ is a small distance, $\hat{n}$ is the cell face normal vector.

When a wall modeled grid is used, a wall function is applied to estimated values for near-wall cells. For turbulence variables, a Dirichlet boundary condition is set at the inlet boundary for k, ω, ν_t and k_{sgs}, ν_{sgs}, while a zero-gradient boundary condition is set at the outlet boundary.

3.1.3 Numerical Schemes

The numerical schemes used to discretize the conventional N-S problem and sub-problem are the same and described as follows. The Navier-Stokes equations are discretized with the Finite-Volume Method (FVM). A transient N-S solver from the OpenFOAM ® *Weller et al.* (1998); *OpenFOAM* (2016) open-source CFD toolkit is used to numerically solve the Navier-Stokes equations (*Jasak* (1996)).

When conducting the URANS simulations, the gradient and Laplacian terms are integrated using Gaussian integration with linear interpolation of the unknowns from the cell center to the face centers. Explicit non-orthogonal correction has been applied for the calculation of surface normal gradients. The convective terms are also calculated with Gaussian integration, but second-order accurate linear upwind interpolation is used instead. The time integration is achieved by implicit-Euler scheme for all cases. After applying the discretization schemes described above, a PISO-type procedure is used to solve for the velocity and pressure coupling. For the discretization of the transport equation of k and ω, the numerical schemes used are consistent, except that linear interpolation is used for the convection term.

For LES calculations, the convection terms are also calculated with Gaussian integration, but the interpolation scheme blends linear (75%) and linear upwind (25%). The time integration is achieved by a second-order backward scheme. Linear interpolation with explicit non-orthogonal correction is used for the Laplacian terms. For the transport equation Eq. (3.10), the same numerical schemes are used except that a TVD scheme with Sweby flux limiter (*Sweby* (1984)) is used for the convection terms.

In the momentum equations, the time derivative, convection, diffusion terms are discretised implicitly. After applying the discretization schemes described above, a PISO-type procedure, is used to solve for the velocity and pressure coupling.

3.2 Viscous Potential Solution Method

For the viscous-potential subproblem, the governing Laplace equation, Eq. (2.28), can be transformed into a Boundary Integral Equation (BIE) Eq. (3.14), using Green's second identity,

$$-2\pi\varphi = \iint_{S_{\mathrm{B}}\cup S_{\mathrm{w}}} \frac{\partial\varphi}{\partial n}G - \varphi\frac{\partial G}{\partial n}\mathrm{dS} \tag{3.14}$$

where G is the free-space Green function that satisfies the Laplace equation.

The viscous boundary condition for the viscous potential is applied to Eq. (3.14). Then the first integral becomes a forcing term $\mathcal{F}$, and the Eq. (3.14) becomes a Fredholm equation of the second kind Eq. (3.15).

$$-2\pi\varphi + \iint_{S_{\mathrm{B}}\cup S_{\mathrm{w}}} \varphi\frac{\partial G}{\partial n}\mathrm{ds} = \mathcal{F} \tag{3.15}$$

In viscous flows, it is important to account for the viscous effects in the wake. To do so, we impose the viscous boundary condition for the viscous potential on a wake surface S_{w} directly after the body. The wake surface essentially acts as an extension of the body to capture the evolving viscous wake. The configuration of the wake surface is important for an accurate solution of the viscous potential.

The BIE Eq. (3.15) is solved by a direct BEM approach. A higher-order Boundary-Element-Method potential flow solver, Aegir, is used to solve for the viscous potential, instead of the 2-D panel method used in previous work. As shown in *Kring et al.* (1999) - *Joncquez et al.* (2012), the flexibility and robustness of Aegir has been demonstrated for various types of problems such as free-surface flows around very large marine structures, and seakeeping and maneuvering of naval vessels. In Aegir, the geometry of the body is represented by Non-Uniform Rational B-Splines (NURBS) surfaces. The velocity potential on the body is expressed as a collection of arbitrary-order B-splines in the parametric space Eq. (3.16), which is independent of the body geometry,

$$\varphi(u,v) = \sum_{m=1}^{M+k-1} \sum_{n=1}^{N+k-1} \tilde{\varphi}_{mn} \mathbf{B}_m(u) \mathbf{B}_n(v) \tag{3.16}$$

where M, N are the number of elements in each direction, k is the order of the B-spline, $\tilde{\varphi}_{mn}$ contains the spline coefficients and $\mathbf{B}$ are the basis functions of order k *Kring et al.* (1999).

Using Aegir allows us to have different discretization for BEM and FVM solver. This is especially desired for performing high-fidelity simulations, because the mesh is very fine for the FVM solver and the BEM solver only needs a moderate resolution to resolve the viscous potential. Another advantage of using Aegir is that to solve the same problem for a fixed relative error, a higher-order BEM solver has much fewer unknowns and require less computational time compared to a constant panel BEM solver, as demonstrated by *Maniar* (1995). Having fewer unknowns in the BEM solver also means a faster computation when the BEM solver evaluates velocity in the flow field for updating the boundary condition of the FVM solver, after the viscous potential is solved. The details of the discretization for both the BEM and FVM solvers of each test case are presented in Chapter IV - VI

The integral equation can be transformed to a linear system of equations by two different approaches: the collocation method and the Galerkin method. For the collocation method, the centroids of the basis functions in the parametric space are used as collocation points. The integration for both of the methods is calculated with a Gaussian-Legendre quadrature rule. Then solving for the viscous potential in BIE Eq. (3.15) is essentially solving for the spline coefficients $\tilde{\varphi}_{mn}$ in Eq. (3.16).

Sclavounos (1987) demonstrated that for the solution of the BIE for an ellipse translating in an infinite fluid, the Galerkin approach is shown to be more accurate compared to the collocation method, with a slightly higher computational effort. Motivated by the higher accuracy, the Galerkin method is used in this work.

3.2.1 Iterative Viscous Boundary Condition for the Viscous Potential

As noted in Section 2.4.1, the vortical velocity $\mathbf{w}$ in Eq. (2.35) is unknown, and an iterative viscous boundary condition needs to be developed. Following the expression in *Rosemurgy et al.* (2016) and *Edmund et al.* (2013), the iterative viscous boundary condition is derived by using the fact that the term $w_n(\delta)$ in Eq. (2.35) is not zero before the desired viscous potential is determined. Then $w_n(\delta)$ is moved to the left-hand side to act as a residual correction in the next iteration. Then the iterative viscous boundary condition can be expressed as in Eq. (3.17).

$$\begin{aligned}
\frac{\partial \varphi}{\partial n}^{(i+1)} &= -w_n^{(i)}(\mathbf{x}_b) + w_n^{(i)}(\mathbf{x}_\delta) \qquad\qquad \text{on } S_{\mathrm{B}} \\
&= \frac{\partial \varphi}{\partial n}^{(i)}(\mathbf{x}_b) + w_n^{(i)}(\mathbf{x}_\delta) \\
&= \frac{\partial \varphi}{\partial n}^{(i)}(\mathbf{x}_b) + u_n(\mathbf{x}_\delta) - \frac{\partial \varphi}{\partial n}^{(i)}(\mathbf{x}_\delta) \\
&= \chi^{(i)}
\end{aligned} \tag{3.17}$$

The inviscid potential is used as the initial guess for this iterative condition.

For the viscous boundary condition at the wake surface Eq. (2.38), a similar process can be carried out Eq. (3.18).

$$\begin{aligned}
\frac{\partial \varphi}{\partial n}^{(i+1)} &= u_n(\mathbf{x}_{\mathrm{w}}) - w_n^{(i)}(\mathbf{x}_{\mathrm{w}}) + w_n^{(i)}(\mathbf{x}_\delta) \qquad\qquad \text{on } S_{\mathrm{W}} \\
&= \frac{\partial \varphi}{\partial n}^{(i)}(\mathbf{x}_{\mathrm{w}}) + u_n(\mathbf{x}_\delta) - \frac{\partial \varphi}{\partial n}^{(i)}(\mathbf{x}_\delta) \\
&= \chi^{(i)}
\end{aligned} \tag{3.18}$$

The iterative viscous boundary condition is identical on the body and wake surface. This also implies that for the viscous potential, the wake surface and the body surface physically have the same function, that is to represent the vortical region.

This makes the way of discretizing the wake surface different from the one for the inviscid potential. For a BEM solving inviscid potential flows, researchers usually try to align the wake surfaces with the stream surfaces. However, for the viscous potential calculated in this work, it is more important to place the wake surface inside the vortical wake to capture the viscous effect. The shape of the wake surface should be topologically similar to the shape of the vortical wake, so that when determining the boundary of the vortical region S_δ using the wake surface normal, it can provide a good representation of the boundary of the vortical region at the wake. The determination of the boundary of the vortical region is discussed in detail in Section 3.2.2.

The iterative viscous boundary condition Eq. (3.17)-Eq. (3.18) is essentially a form of Picard iteration. The normal component of the vortical velocity acts as a residual correction and it is eventually driven to zero or a negligible value. Based on these observations, other forms of the iterative viscous boundary condition can be derived. For example, the secant method can be applied to gain possible improvement for convergence. However, this is not necessary as discussed in Section 3.2.1.1 and demonstrated in Chapter IV, the current form of iterative viscous boundary condition Eq. (3.17)-Eq. (3.18) converges rapidly for our target applications.

3.2.1.1 Linear stability analysis

In previous work, the iterative viscous boundary condition Eq. (3.17) is applied to successfully solve the viscous potential for multiple test cases. However, the stability of the iterative viscous boundary condition has not been studied, although the iterations are shown to converge rapidly for most of the test cases, usually around three iterations for attached flows and no more than ten iterations for separated flows. To better understand the convergence of the iterative viscous boundary condition and to bring some confidence of its application, the linear stability of the iterative boundary condition Eq. (3.17) is analyzed and related to the physics of the flow field in this section.

To have a clear view of the analysis, a constant source panel method, Eq. (3.19), is used to represent the viscous potential.

$$\frac{\partial \varphi}{\partial n}(\mathbf{x}) = [C(\mathbf{x})]\,\{\sigma\} \tag{3.19}$$

where $[C(\mathbf{x})]$ is the coefficient matrix evaluated at the location $\mathbf{x}$ and $\{\sigma\}$ is the vector of the source strengths. Plugging in Eq. (3.19), Eq. (3.17) yields Eq. (3.20)

$$\begin{aligned} [C(0)]\,\{\sigma\}^{(i+1)} &= [C(0)]\{\sigma\}^{(i)} - [C(\delta)]\{\sigma\}^{(i)} + \{u_n(\delta)\} \\ \{\sigma\}^{(i+1)} &= \big\{[I] - [C(0)]^{-1} \cdot [C(\delta)]\big\}\,\{\sigma\}^{(i)} + [C(0)]^{-1} \cdot \{u_n(\delta)\} \end{aligned} \tag{3.20}$$

where $[I]$ is the identity matrix, $[C(\delta)]$ and $[C(0)]$ are the coefficient matrices evaluated at the boundary of the vortical region and the body boundary respectively. Let $[M] = \{[I] - [C(0)]^{-1} \cdot [C(\delta)]\}$, and $[S] = [C(0)]^{-1} \cdot \{u_n(\delta)\}$ such that Eq. (3.20) can be expressed as Eq. (3.21).

$$\{\sigma\}^{(i+1)} = [M]\{\sigma\}^{(i)} + [S] \tag{3.21}$$

For the iterative function Eq. (3.21) to be linearly stable, the magnitudes of all eigenvalues of matrix $[M]$ must be smaller than 1, $|\lambda([M])| < 1$. In cases for Reynolds numbers approaching infinity ($Re \to \infty$), the thickness of the boundary layer approaches zero ($\delta \to 0$), $[C(\delta)] \to [C(0)]$ and $\lambda([M]) \to 0$, the stability approaches superstable, and the potential becomes the inviscid potential.

As discussed in *Batchelor* (2000), the distance δ that vorticity transported away from the body is of order of magnitude $\sqrt{\nu t}$, when t is of small value (t is time). When the flow is fully developed ($t \to \infty$), δ is of order $\sqrt{\nu L/U_\infty} = L/\sqrt{Re}$. For most of the engineering applications, the Reynolds number is in the range of $10^6 \sim 10^9$, the convection of vorticity in the downstream direction is much stronger than the diffusion of vorticity away from the body. The vorticity is convected downstream before it can spread far laterally, which gives rise to a thin vortical region around the body and wake (small δ). So for these engineering applications, the iteration converges rapidly. This is consistent with the cases studied by the research group, the number of iterations required for streamlined body is found to be around three (*Chen and Maki* (2016); *Rosemurgy* (2014); *Edmund* (2012a)).

For cases with bluff bodies and massive separation, which means larger δ, the convergence of the iteration may be slower. Such types of flows have been studied in *Edmund* (2012a); *Rosemurgy et al.* (2012); *Chen et al.* (2015), *i.e.* flow over a 2D circular cylinder at Re = 60, 140, and 320,000. In *Chen et al.* (2015), flow over cylinder at Re = 140 was studied (some of the results are shown in Fig. 2.6). At this Re number, a Von Kármán vortex street is formed at the wake of the cylinder and δ can be as large as five to six times of the diameter of the cylinder. These test cases can represent almost the worst case scenario for the iterative viscous boundary condition. For such cases, the required number of iterations is no more than 10. In our target applications, δ is much smaller because of the higher Reynolds number. Hence the convergence of the iterative boundary condition should not be a concern for our intended applications.

Although a simplified representation of the potential (constant source panel) is used in the analysis, the conclusion should apply in the general case, including the higher-order BEM in Aegir, because the numerical representation of the potential in BEM, eventually can be expressed as a coefficient matrix times some source strengths, or weights, as in Eq. (3.19).

3.2.2 Determination of the Boundary of the Vortical Region

To apply the viscous boundary condition for the viscous potential the total velocity at the boundary of the vortical region of the flow field $u_n(\delta)$ is needed, and thus vorticity thickness δ must be determined. Both the irrotational and the vortical flow regions can be unsteady, which means their shapes and volumes can change in time ($\delta = \delta(\mathbf{x}, t)$) as well as the flow variables ($i.e.\mathbf{u}, p$) in the regions. The vortical regions for a case of flow over a oil tanker ship hull in different time instances are shown in Fig. 3.1. As can be seen, the vortical region (colored in red) at the developing stage of the flow field shown in the upper figure of Fig. 3.1 is different from the one of the

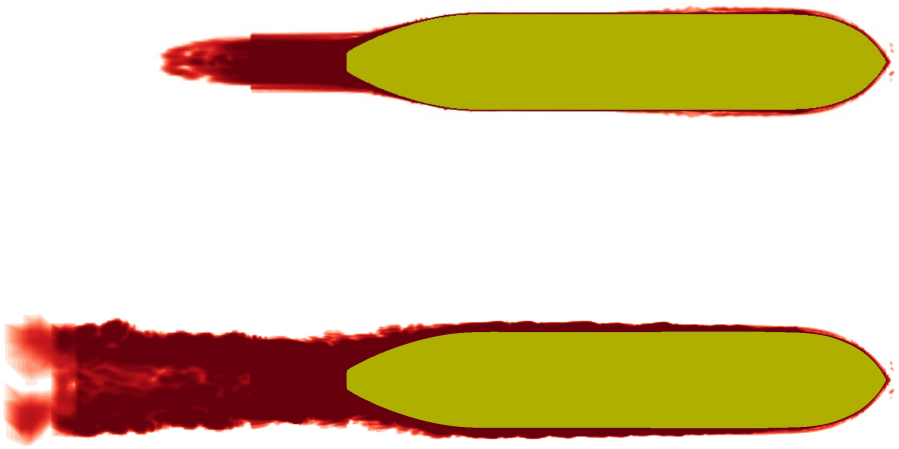


Figure 3.1: Vortical region at different time of the flow field

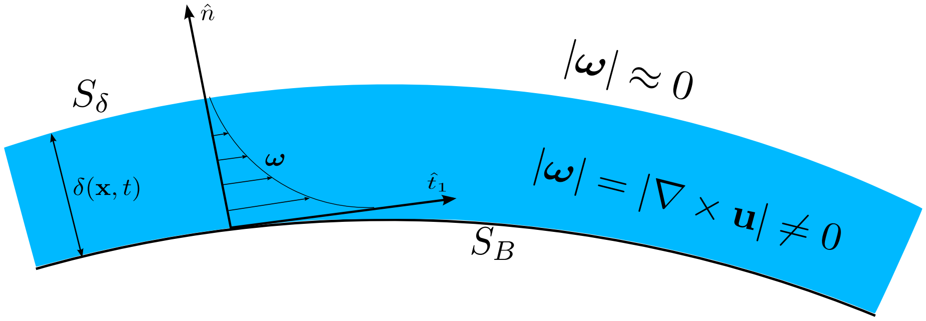

$\hat{n}$
S_δ
$|\boldsymbol{\omega}| \approx 0$
$\delta(\mathbf{x}, t)$
$\boldsymbol{\omega}$
$\hat{t}_1$
$|\boldsymbol{\omega}| = |\nabla \times \mathbf{u}| \neq 0$
S_B

Figure 3.2: $\delta(\mathbf{x}, t)$ defined in local-orthogonal coordinate

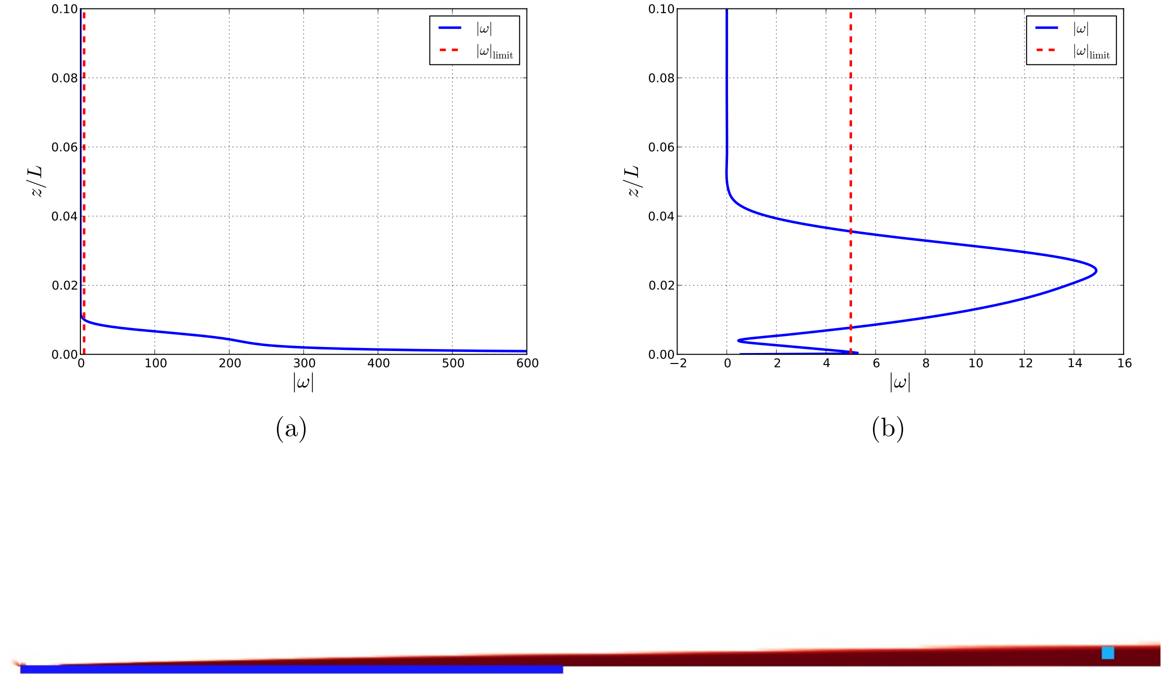
fully developed flow field shown in the lower figure of Fig. 3.1.

As illustrated in Fig. 3.2, $\delta(\mathbf{x}, t)$ is the normal distance from the body where the magnitude of vorticity $|\boldsymbol{\omega}|$ is negligible. In numerical simulation, it is not practical to find the vanishing magnitude of the vorticity. Hence, an user-defined threshold value $|\boldsymbol{\omega}|_{\text{limit}}$ for the negligible magnitude of the vorticity is used to determine the distance δ in the local-normal direction.

It is important to define a physical meaningful threshold value $|\boldsymbol{\omega}|_{\text{limit}}$ to accurately determine δ. At first, it might seem obvious to define a small value ϵ_ω as the vorticity threshold Eq. (3.22).

$$|\boldsymbol{\omega}|_{\text{limit}} = \epsilon_\omega \tag{3.22}$$

However, it is difficult to find a threshold value that will work for all different cases.



(a)

(b)

(c)

Figure 3.3: Profile of vorticity magnitude for flow over a finite-length flat plate: (a) at the center of the flat plate (b) at the viscous wake region downstream to the flat plate; (c) vorticity magnitude contour plot (the thick blue line marks the location of the flat plate)

As the vorticity is convected downstream from the body and diffused laterally, the total magnitude of the vorticity integrated across the wake region will not change, but the vorticity is spread out to give rise to a thicker wake and the local magnitude of vorticity is smaller. Then using a global threshold value can incorrectly identify the part of the viscous wake region as as irrotational. An example is shown in Fig. 3.3, with profiles of vorticity magnitude for the flow over a finite-length flat plate sampling at the center of the flat plate and the wake. A vorticity threshold is defined as $|\boldsymbol{\omega}|_{\text{limit}} = \epsilon_\omega = U_\infty/L$, where L is the length of the flat plate. As can been seen, a threshold that is suitable to define the negligible vorticity at the center of the flat plate Fig. 3.3(a) can be specified. However, this threshold misses part of the vorticity when used in the wake Fig. 3.3(b). The location defined as the boundary of the vortical region is shown as the light blue dot in the vorticity magnitude contour plot Fig. 3.3(c) and it is clearly still inside the vortical region. So it is important to define a dynamic and localized vorticity threshold value.

Some insight can be gained by taking the curl product of the solution of the first Stokes problem. The vorticity distribution can be expressed as:

$$\boldsymbol{\omega} = -\frac{U_\infty}{\sqrt{\pi \nu t}} \exp\left(-\frac{y^2}{4\nu t}\right). \tag{3.23}$$

As the flow developed, the variable t can be changed with x/U_∞ based on Rayleigh's analogy (*Rosenhead* (1963))

$$\boldsymbol{\omega} = -U_\infty \sqrt{\frac{U_\infty}{\pi \nu x}} \exp\left(-\frac{y^2 U_\infty}{4\nu x}\right). \tag{3.24}$$

In the wake region, we take the curl product of the asymptotic solution based on boundary layer thoery for wake behind a body (*Schlichting and Gersten* (2000))

$$\boldsymbol{\omega} = -\frac{C_d}{8\sqrt{\pi}} \sqrt{\frac{U_\infty L}{\nu}} (\frac{x}{L})^{-0.5} \frac{U_\infty y}{\nu} \frac{U_\infty}{x} \exp\left(-\frac{y^2 U_\infty}{4\nu x}\right), \tag{3.25}$$

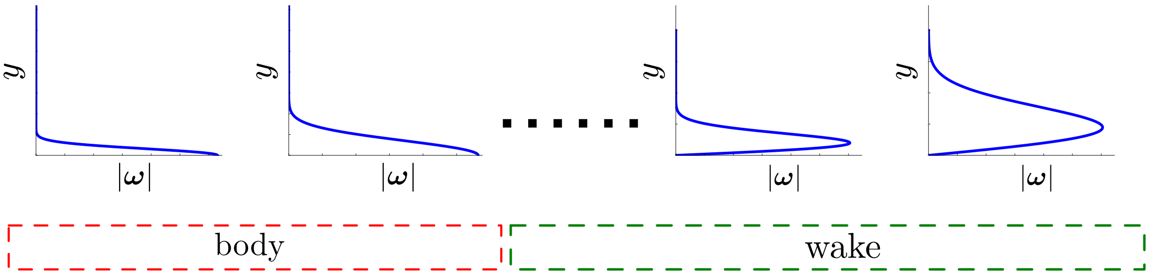


Figure 3.4: Approximated vorticity distribution based on boundary layer theory

where C_d is the drag coefficient.

These estimations imply that the vorticity essentially behaves as the solution of the heat conduction equation and its magnitude decays exponentially away from the source (wall). The approximated vorticity distribution estimated by the similarity solutions of boundary layer theory Eq. (3.24)-(3.25) are plotted in Fig. 3.4 for different locations. Therefore, a relative vorticity threshold can be defined as a fraction of the local maximum vorticity magnitude $|\boldsymbol{\omega}_{\max}|$ (Eq. (3.26))

$$|\boldsymbol{\omega}|_{\text{limit}} = \alpha_\omega |\boldsymbol{\omega}_{\max}|, \tag{3.26}$$

where α_ω is a small fraction value. The only requirement for α_ω is that it is selected as a small value such that a negligible vorticity-magnitude limit $|\omega|_{\text{limit}}$ can be calculated. Based on the results of all the test cases studied in previous work and current research, the velocity-decomposition algorithm is able to provide accurate solutions as long as $\alpha_\omega \leq 1\%$. In this thesis, $\alpha_\omega = 0.1\%$ is used for all cases. Effects of different values of α_ω (ranging from 0.01% to 10%) are studied for flows over 2D canonical bodies in *Edmund* (2012a). The relative threshold Eq. (3.26) performs well in finding the boundaries of the vortical regions for various test cases in previous work *Edmund* (2012a), *Rosemurgy* (2014), *Chen and Maki* (2016). However, this definition of the vorticity threshold is not without problems. When the local threshold value becomes small, it can be effected by the fictitious vorticity due to numerical error and results in a δ value that is larger than the correct value. This will make the reduced computational domain (encompassing just the vortical region) used by the velocity decomposition solver larger than its necessary size. Since the savings of the computational effort of using the velocity decomposition approach mainly comes from the reduction of the domain size, this is undesirable. The remedy is to add a filter criteria and the new vorticity value is defined as

$$|\boldsymbol{\omega}|_{\text{limit}} = \max\left(\alpha_\omega |\boldsymbol{\omega}_{\max}|,\ |\boldsymbol{\omega}|_{\text{filter}}\right), \tag{3.27}$$

where $|\boldsymbol{\omega}|_{\text{filter}} = \alpha_{\text{f}}\ U_\infty/L$ is the vorticity filter value and α_{f} is usually set as 0.1. For unsteady flow simulations, this vorticity filter is also added to prevent fictitious numerical fluctuations at the area where the vorticity has not yet developed.

Because the BEM and the FVM solver employ different discretizations, the representation of the body boundary must be connected. The interpolation points from Aegir are paired with the closest wall face center for the body surface and the closest cell center for the points on wake surface. This pairing procedure only need be done once and the cell indexes are stored and can be re-used. The geometry in Aegir is represented by NURBS, and this is considered more accurate and has continuous surface gradients compared to the flat-panel-surface representation in the FVM. The surface normal from Aegir discretization is selected as the local normal direction for the sampling process in determining δ.

To correctly determine the value of δ, it is important to have a robust algorithm that can sample outward to a user-defined maximum distance $\delta_{\max}$ to obtain a list of cells that lies in the local-normal direction of the body or wake surfaces and locate the one that has negligible magnitude of vorticity. The list of cells in the local-normal direction is determined by recursively finding the neighbor cell that is most aligned with the local normal vector. The normal distance between the N^{th} cell within this list and the boundary surface are required to be larger than that of the $(N-1)^{\text{th}}$ cell. Once the cell with negligible vorticity magnitude is located, linear interpolation is used to determine the value for δ within the cells. This algorithm performs well in previous work for cases with canonical geometries and structured mesh *Edmund* (2012a), *Rosemurgy* (2014), *Chen* (2014).

In this work, 3D cases with realistic geometries are studied and unstructured mesh generated by the snappyHexMesh application in OpenFOAM is used. Using unstructured mesh greatly reduces the time required by the mesh generation process, especially for cases with complicated geometries. However, an unstructured mesh can lead to challenges for finding cells in the local-normal direction. As noted in Section 6.2 in *Rosemurgy* (2014), the algorithm to determine the value of δ sometimes fails for cells with high aspect-ratio or large skewness. Moreover, this problem tends to occur more frequently for the test cases with unstructured meshes studied in this thesis, because the cells in unstructured meshes are not aligned with the wall normal direction as in most of structured meshes. To resolve this issue, an additional criteria that ensures the cells within the list are unique, is added for searching the list of cells in the local normal direction.

For simulations that involve complicated geometries, such as ship hulls, problems of finding a correct δ may rise. Some examples are shown in Fig. 3.5, where the FVM unstructured mesh and the vorticity-magnitude contour (colored in red) for a case with a oil tanker ship hull is shown. For demonstration purposes, an overly coarse mesh is shown and the vorticity-magnitude contour is plot in log scale. The body boundary (the ship hull) S_{B} is colored in yellow, S_{t} is a boundary of the domain and the vortical region is colored in red. The stern and the bow region of the flow field is shown in Fig. 3.5 (a) and (b) respectively. One of the scenarios that may cause problems for finding the correct value of δ is when part of the body surface is concave, as shown in Fig. 3.5 (a). When the sampling process reaches a boundary of the computational domain, the cells that are not aligned with the local-normal vector might get selected and this will lead to a false δ value. In this case, δ is set to zero, once the sampling process reaches the boundaries of the computational domain and the traditional non-penetration boundary condition is applied for that location. Usually δ is set to zero when the local vorticity magnitude is negligible or the vortical region is very thin, and $\delta = 0$ is a valid approximation. When δ equals zero, the boundary condition on the body and wake surface can be changed to the non-penetration boundary condition because of Eq. (2.21). Fig. 3.5(b) shows an example of the case just described.

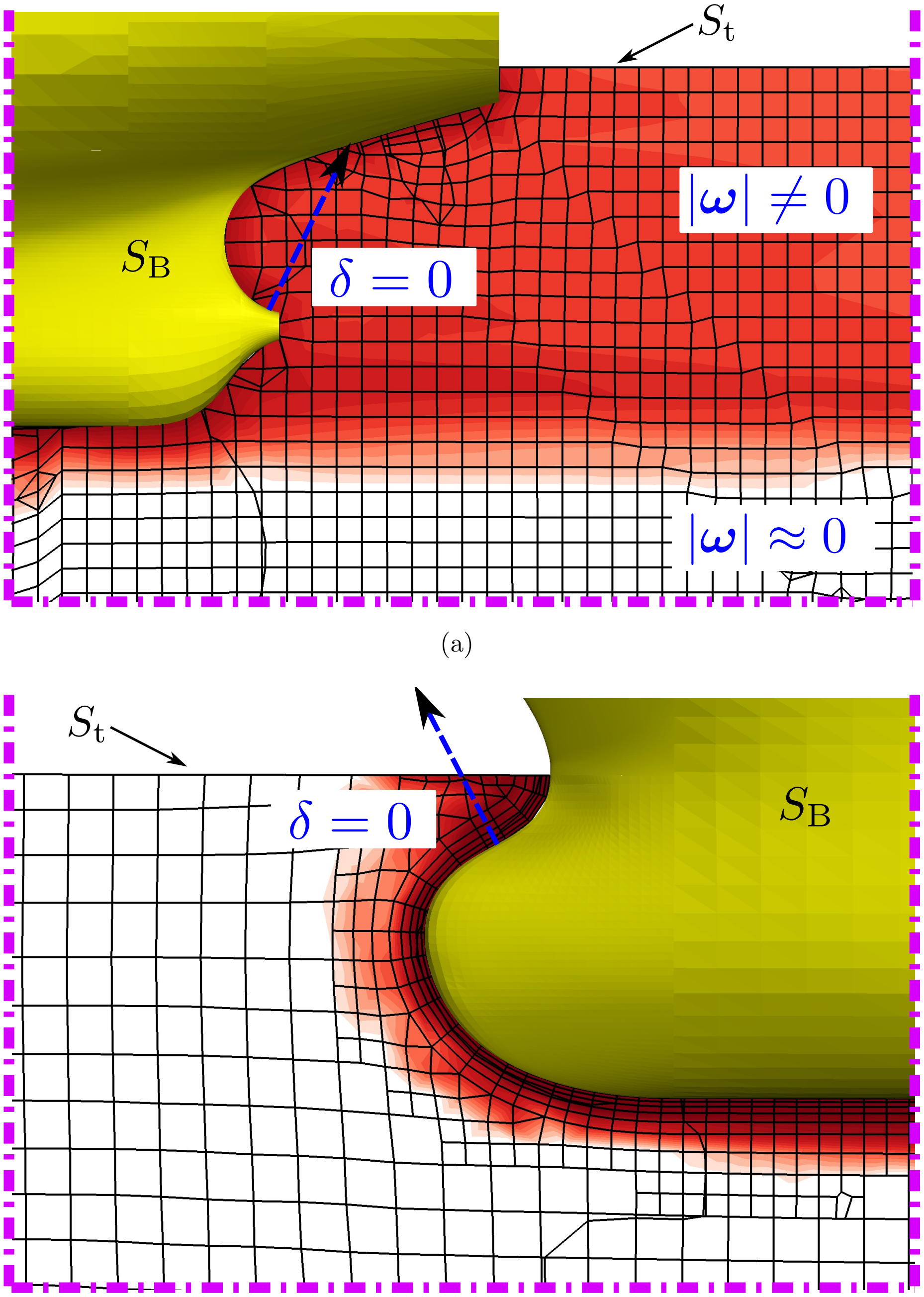
When the area of the boundary surface, where δ is set to zero, is located inside the vortical region (Fig. 3.5(b) shows an example of this case), setting the boundary condition to non-penetration condition may not be a suitable approximation. Since a physically meaningful boundary condition can not be found, the boundary condition for this location should take the average value of the ones from its neighboring locations so that a smooth profile for the boundary condition over the body and wake surface can be maintained. This averaging is done implicitly, because of the Galerkin approach is used to solve the BIE (as described in Section 3.2).

The steps of the procedure for determining the vorticity thickness δ are summarized below:

D1. Obtain a list of cells by sampling from the boundary cell face center outward in local-normal direction until the user-defined maximum distance, δ_{max} is reached, then find the maximum of the vorticity magnitude, $|\boldsymbol{\omega}|_{max}$ within this list of cells.

D2. If $|\boldsymbol{\omega}|_{max}$ is smaller than a user-defined filter value, $|\boldsymbol{\omega}|_{\text{filter}} = \alpha_{\text{f}}\ U_{\infty}/L$, δ is set to zero.

D3. Starting from the cell with the maximum vorticity magnitude, sample outword


S_t
$|\boldsymbol{\omega}| \neq 0$
S_B
$\delta = 0$
$|\boldsymbol{\omega}| \approx 0$
S_t
$\delta = 0$
S_B

(a)

(b)

Figure 3.5: Determination of δ for ship hull geometries: (a) the stern region (b) the bow region; of the flow field

in local-normal direction to find the cell with vorticity magnitude smaller than the vorticity threshold $|\boldsymbol{\omega}|_{\text{limit}}$.

D4. If the cell with negligible vorticity magnitude is not found or the sampling process reaches the boundaries of the domain, δ is set to zero and the inviscid non-penetration boundary condition is applied for this location.

3.2.3 Coupling of the Viscous Potential and the Total Velocity in Time

The velocity decomposition algorithm has been successfully applied in *Edmund* (2012a); *Rosemurgy* (2014) to various steady flow problems, including 2D and axis-symmetric, non-lifting, single-phase flows, 2D single-phase lifting flows and 2D fully submerged free-surface flows.

However, to be able to apply the velocity decomposition approach for transient 3D flow simulations, An algorithm to couple the viscous potential and the total velocity in time needs to be developed. *Edmund* (2012b) has demonstrated the feasibility of applying the velocity decomposition approach for unsteady flows. *Chen et al.* (2015) has proposed two unsteady algorithms to apply the velocity decomposition approach to solve unsteady flow problems and promising results have been presented. In this work, the instantaneous-velocity-based method in *Chen et al.* (2015) is adopted. The coupling algorithm is described in this section. In Eq. (3.28), all three variables are assumed to vary in time, but not necessarily in the same time scale.

$$\mathbf{u}(\mathbf{x},t) = \nabla\varphi(\mathbf{x},t) + \mathbf{w}(\mathbf{x},t) \tag{3.28}$$

For instance, the small-time-scale turbulent flow is mostly contained inside the rotational flow region and has little influence on the time dependence of the viscous potential, while the unsteadiness of the viscous potential mainly depends on the disturbance with large time scale, such as the shedding of vortices. Therefore, we allow

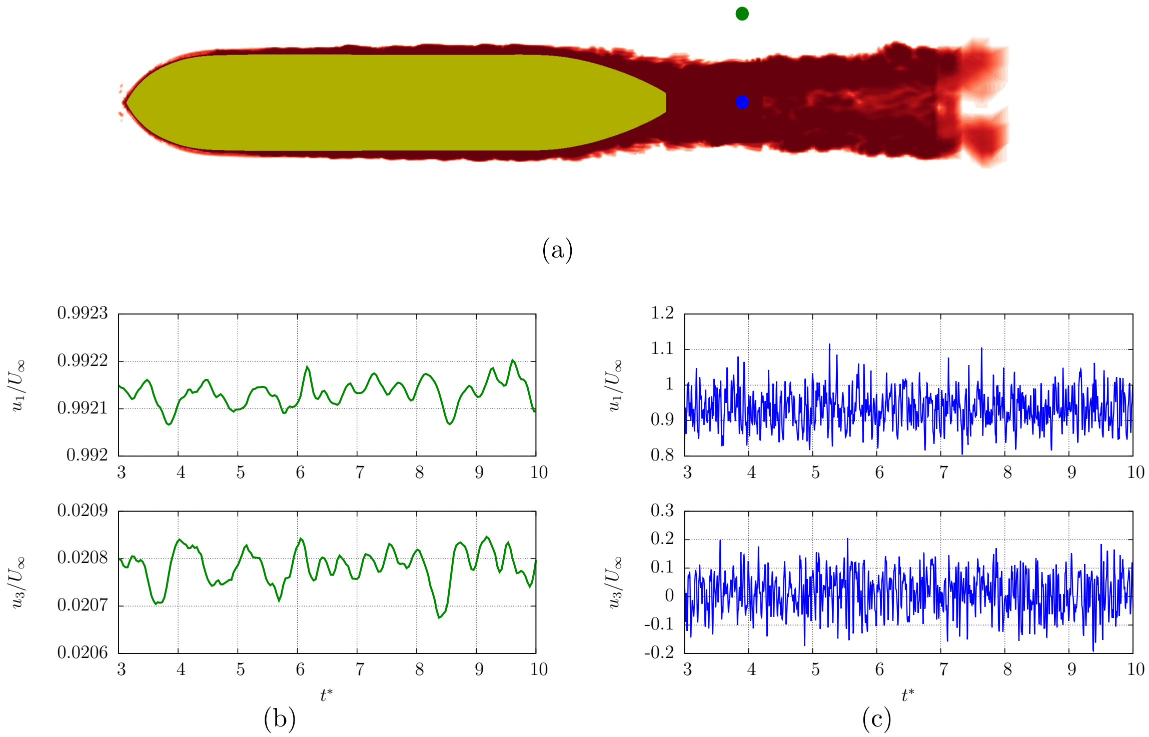


Figure 3.6: (a) Vorticity-magnitude contour and probe locations for a case of flow over an oil tanker ship hull; Velocity time history for flow (b)outside the vortical region (c)inside the vortical region.

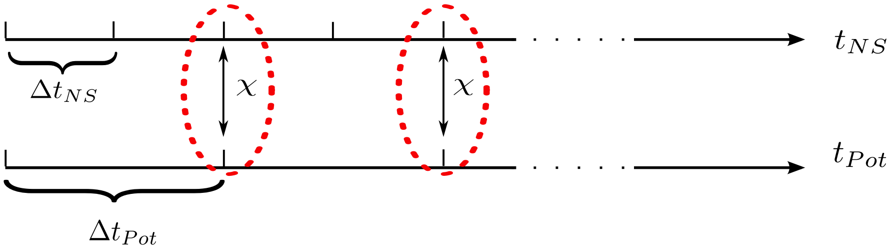


Figure 3.7: Tightly couple between viscous potential and total velocity

the solution of the viscous potential to be done at a different time increment than the field solution. Typical velocity time histories in the irrotational and vortical flow regions are shown in Fig. 3.6 (b) and (c) respectively. The locations where the velocity time histories are recorded are marked as green and blue dots in the vorticity contour plot shown in Fig. 3.6(a). It should be noted that the results shown in Fig. 3.6 are generated by a LES calculation. So the fluctuations in the velocity time history are due to the resolved large-scale turbulence.

As shown in Fig. 3.7, the solutions of the viscous potential and the Navier-Stokes sub-problems evolve in time with different time step sizes, but they are tightly coupled through the viscous body boundary condition χ, where the subscripts NS and Pot represent the Navier-Stokes and the viscous potential sub-problems respectively.

The process of solving the viscous-potential velocity and applying it as the boundary condition for the Navier-Stokes sub-problem is referred to as an update. The time when each update is performed can be controlled by the user through the expression Eq. (3.29).

$$t_{\text{update}} = t_0 + n_{\text{update}} \cdot \Delta t_{\text{update}} \qquad (n_{\text{update}} = 0, 1, 2... < N_{\text{update}}) \tag{3.29}$$

where t_0 is the time of the initial update, n_{update} is the update counter that indicates the number of updates that has been made, Δt_{update} is the time interval between updates which means how frequently updates should be made, N_{update} is the total number of updates that needs to be applied. If necessary, the update frequency can be made the same as the time step size in Navier-Stokes sub-problem $\Delta t_{\text{update}} = \Delta t_{\text{NS}}$. However, this does not happen for most of flow problems of interest.

Fig. 3.8 shows the flow chart of the coupling procedure between Aegir and the FVM. The labels *OF* and *Aegir* indicate the procedure conducted within OpenFOAM or Aegir, respectively. Each step of the flow chart is described below:

1. In OpenFOAM, solve the Navier-Stokes sub-problem with the given initial and boundary conditions until the update conditions are reached .

2. If the update conditions are reached, Aegir sends the interpolation points x_{A} and the local normals n_{A} to OpenFOAM.

3. In OpenFOAM, calculate the vorticity thickness, δ for each x_{A}, based on the magnitude of the vorticity, $\omega = \nabla \times \mathbf{u}$, which is essentially to determine the location where the flow becomes irrotational. Send points on the reduced domain boundary $\boldsymbol{x}_{\text{S}_{\text{R}}}$ to Aegir.

4. In Aegir, calculate the viscous potential by iterating the following steps for specified number of times. A convergence criteria relating the magnitude of the vortical velocity may also be specified as implemented in *Chen et al.* (2015); *Rosemurgy et al.* (2016).

   (a) Calculate the vortical velocity, $\mathbf{w}^{(i)} = \mathbf{u} - \nabla\varphi^{(i)}$. (The superscripts, (i), denote the iteration counter.)

   (b) Calculate the viscous potential using the iterative form of the viscous boundary condition, Eq. (3.17)-(3.18). The inviscid potential ϕ_{inv} is used as initial guess.

   (c) If the specified iteration number is met, the viscous-potential velocities at the boundary of the reduced domain $\nabla\varphi(\boldsymbol{x}_{\text{S}_{\text{R}}})$ are calculated. As discussed in Section 3.2.1.1, the iterative viscous boundary condition converges within three iterations for most of the cases and within ten iterations for cases with massive separations. In this thesis, ten iterations are used for all test cases. Moreover, a convergence criteria related to the magnitude of the vortical velocity $|\mathbf{w}|$ can also be used (see *Rosemurgy* (2014), *Chen* (2014)).

5. OpenFOAM reads viscous-potential velocities $\nabla\varphi(\boldsymbol{x}_{\mathrm{S_R}})$ from Aegir, and updates the new boundary condition for the Navier-Stokes sub-problem. Go back to step 1.

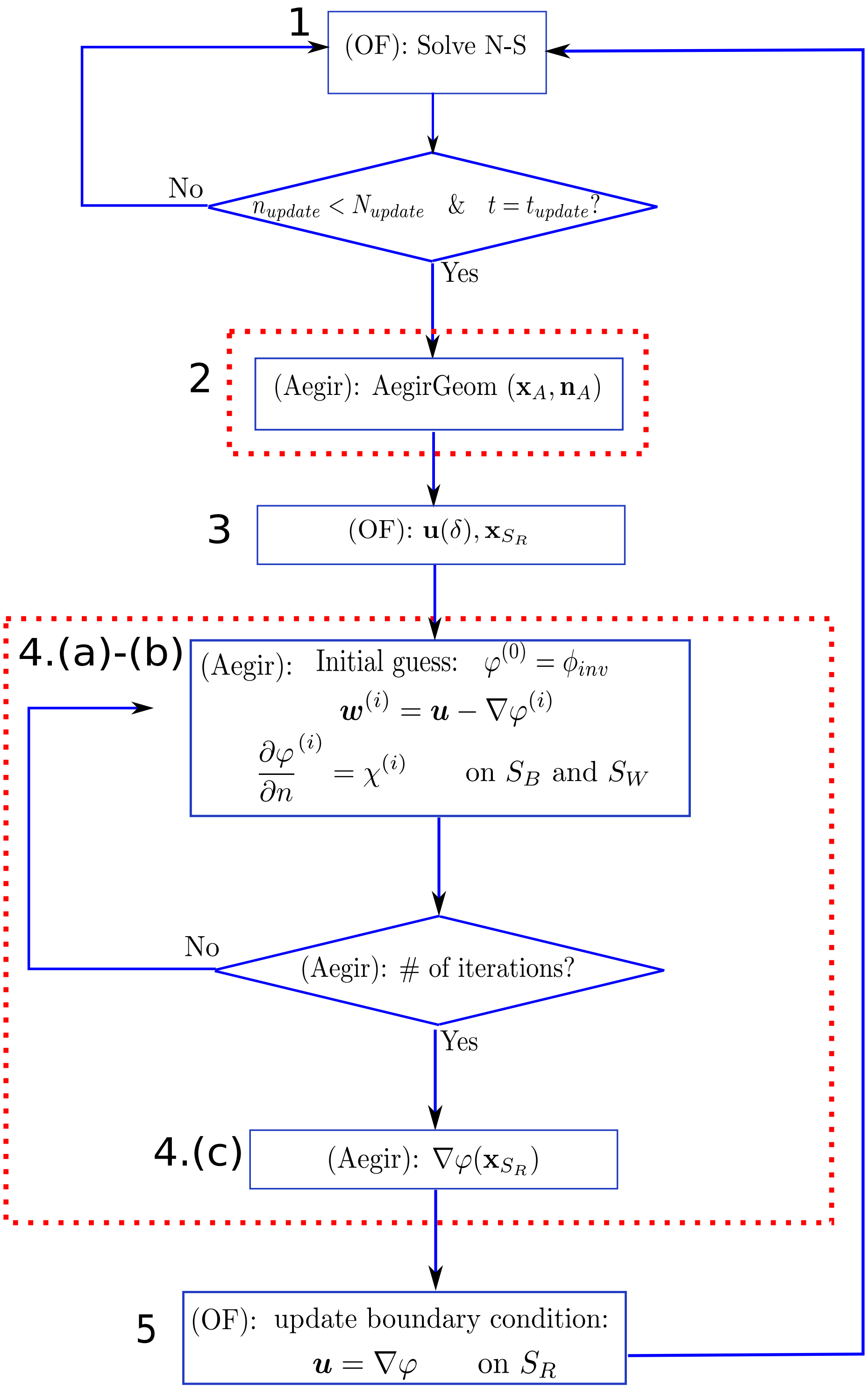

1
(OF): Solve N-S
No
$n_{update} < N_{update}$ & $t = t_{update}$?
Yes
2
(Aegir): AegirGeom $(\mathbf{x}_A, \mathbf{n}_A)$
3
(OF): $\mathbf{u}(\delta), \mathbf{x}_{S_R}$
4.(a)-(b)
(Aegir): Initial guess: $\varphi^{(0)} = \phi_{inv}$
$\boldsymbol{w}^{(i)} = \boldsymbol{u} - \nabla\varphi^{(i)}$
$\frac{\partial\varphi}{\partial n}^{(i)} = \chi^{(i)}$ on S_B and S_W
No
(Aegir): # of iterations?
Yes
4.(c)
(Aegir): $\nabla\varphi(\mathbf{x}_{S_R})$
5
(OF): update boundary condition:
$\boldsymbol{u} = \nabla\varphi$ on S_R

Figure 3.8: Coupling between Aegir and the viscous solver

CHAPTER IV

Results for Canonical Problems

Two sets of test cases are studied in this section, laminar and turbulent flow over a finite-length flat plate, and turbulent double-body flow over a Wigley hull. The flat plate cases are selected due to the simple geometry so that the viscous effects on the potential can be isolated. The inviscid-potential velocity for the flat plate cases is equal to the free-stream velocity, $\nabla\phi_{\text{inv}} = U_\infty$, everywhere in the domain, while the viscous potential accounts for the viscous effect, i.e. the boundary layer and the wake. The Wigley hull is selected to represent a stream-lined body.

For each test case, the benchmark solutions are calculated by using a conventional FVM Navier-Stokes solver on a large computational domain (denoted as NSL). The viscous-potential velocity is first studied, then the velocity decomposition coupled solver is used to solve for the cases on a greatly reduced domain. The solutions from the coupled solver (denoted as VD) are compared with the benchmark solutions.

4.1 Laminar Flow over a Finite-Length Flat Plate

Laminar flow over a finite-length flat plate mounted on a slip wall is studied in this section. The Reynolds number for this case is: $\text{Re} = U_\infty L/\nu = 2000$, where U_∞ is the free-stream velocity, L is the length of the flat plate and ν is the kinematic viscosity of the fluid. The schematic of the computational domains for both the FVM and the BEM are shown in Fig. 4.1. Only half of the domain is calculated due to the symmetry of the problem. In Fig. 4.1(a), symmetry boundary condition is applied on S_{C}, a slip boundary condition is applied S_{L}. S_{I} is the inlet boundary, S_{B} is the body boundary and S_{O} is the outlet boundary. The origin of the coordinate system is located at the center of the leading edge of the flat plate. The x-axis is positive pointing upstream of the flat plate, the z-axis is positive pointing upwards, and the y-axis is defined according to the right-hand rule. In Fig. 4.1(b), panels on the flat plate and the wake are colored in yellow and blue respectively, external panels (colored in gray) are added to model the slip wall.

The details of the discretization in OpenFOAM for the NSL solution and in Aegir for the viscous potential are shown in Table 4.1 and Table 4.2. A discretization convergence study has been conducted for both the FVM solution in OpenFOAM and the BEM solution in Aegir (shown in Appendices A and B). x_{inlet} and x_{outlet} represent the distance between the body and the inlet and outlet boundaries, as shown in Fig. 4.1. Note the large difference in the number of unknowns that need to be solved in FVM (OpenFOAM) and BEM (Aegir) (Table 4.1 - 4.2).

x_{inlet}	x_{outlet}	total # of cells
$50L$	$1L$	$795,156$

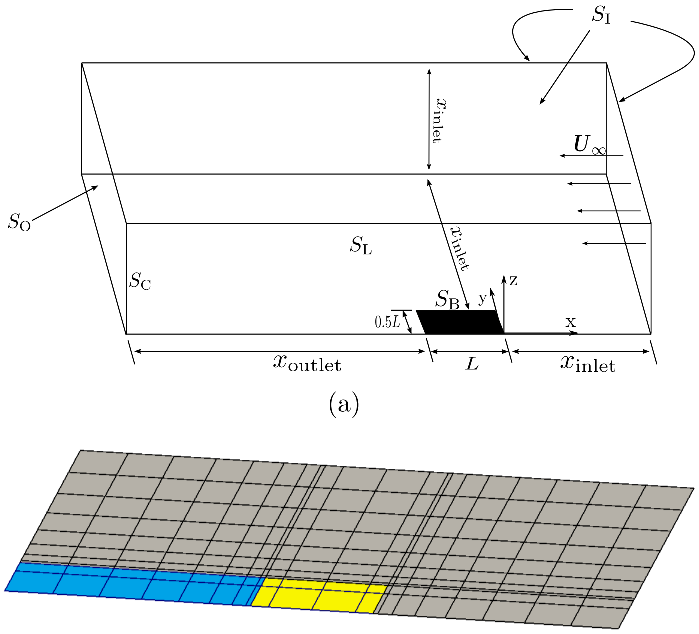
Table 4.1: Discretization in OpenFOAM for the NSL solution for laminar flow over a finite flat plate

wake-panel extent	total # of panels	# of body panels	# of wake panels
1L	160	8	8

Table 4.2: Discretization in Aegir for the viscous potential for laminar flow over a finite flat plate

4.1.1 Viscous Potential

In this case, the inviscid-potential velocity equals to the free-stream velocity $\nabla\phi_{\mathrm{inv}} = \mathbf{U}_{\infty}$. On the other hand, the viscous potential accounts for the viscous



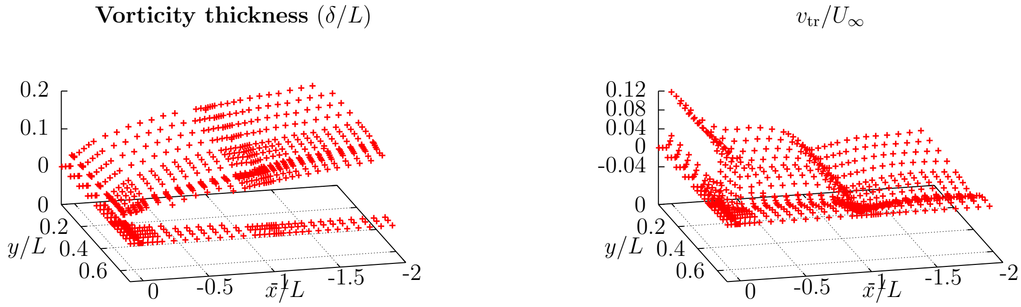
(a)

(b)

Figure 4.1: Sketch of the computational domain: (a) in OpenFOAM; (b) in Aegir

effects, i.e. the vorticity thickness δ and the transpiration flux due to the boundary layer. The vorticity thickness δ is shown in Fig. 4.2(a). As expected, δ takes the shape of the boundary layer and the vortical wake. The thickening of the vortical wake due to the balance of convection and diffusion of the vorticity are well represented by δ. Fig. 4.2(b) represents the transpiration velocity introduced by the viscous-potential velocity, which equals to $u_n(\delta)$, the vertical component of the total velocity at the edge of the vortical region. As can be seen, the viscous potential introduces an influx near the leading edge of the flat plate ($x/L = 0$), that acts to block the flow due to the presence of the boundary layer.

To better examine the accuracy of the decomposition used in this work, the vortical velocity computed with the viscous potential can be compared to that calculated with the inviscid potential. Fig. 4.3 shows contours on the symmetry plane of the dimensionless vorticity and the vortical velocity. Here $\mathbf{w}$ represents the difference between the total velocity $\mathbf{u}$ and the viscous-potential velocity as $\mathbf{w} = \mathbf{u} - \nabla\varphi$.



(a)

(b)

Figure 4.2: Vorticity thickness δ and transpiration velocity v_{tr} due to the boundary layer and wake for laminar flow over a finite flat plate

The vortical velocity calculated based on the inviscid-potential velocity is defined as $\mathbf{w}_{\mathrm{inv}} = \mathbf{u} - \nabla\phi_{\mathrm{inv}}$. All values ($\mathbf{w}^*$, $\mathbf{w}^*_{\mathrm{inv}}$, $\boldsymbol{\omega}^*$) are non-dimensionalized by the free-stream velocity U_∞ and the body length L. The upper right of the Fig. 4.3 shows the contour of the vorticity field which represents the vortical region. As can be seen on the upper left of the Fig. 4.3 , the vortical velocity is only noticeable inside the vortical region and vanishes outside the vortical region. Hence, the viscous-potential velocity agrees well with the total velocity outside the vortical region. On the other hand, the inviscid vortical velocity $\mathbf{w}_{\mathrm{inv}}$ shown in the lower left of Fig. 4.3, does not vanish until several body-lengths away from the body. This is consistent with the results of the inlet location study in Section 4.1.2.

For a more detailed comparison the velocity profiles at four sampling locations for the viscous-potential velocity and the total velocity from NSL are compared in Fig. 4.4. As can be seen, the viscous-potential velocity agrees well with the total velocity.

As discussed in Section 3.2.1.1, the iterative boundary condition Eq. (3.17) for the viscous potential should converged quickly for all targeted applications. The velocity profiles of the viscous-potential velocity after different number of iterations

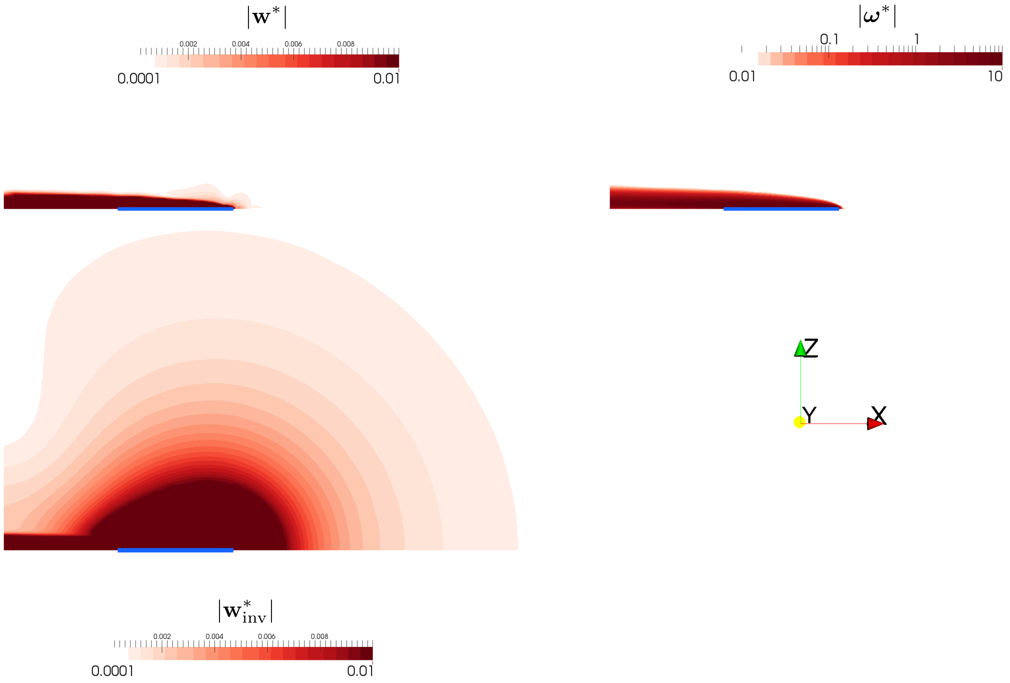
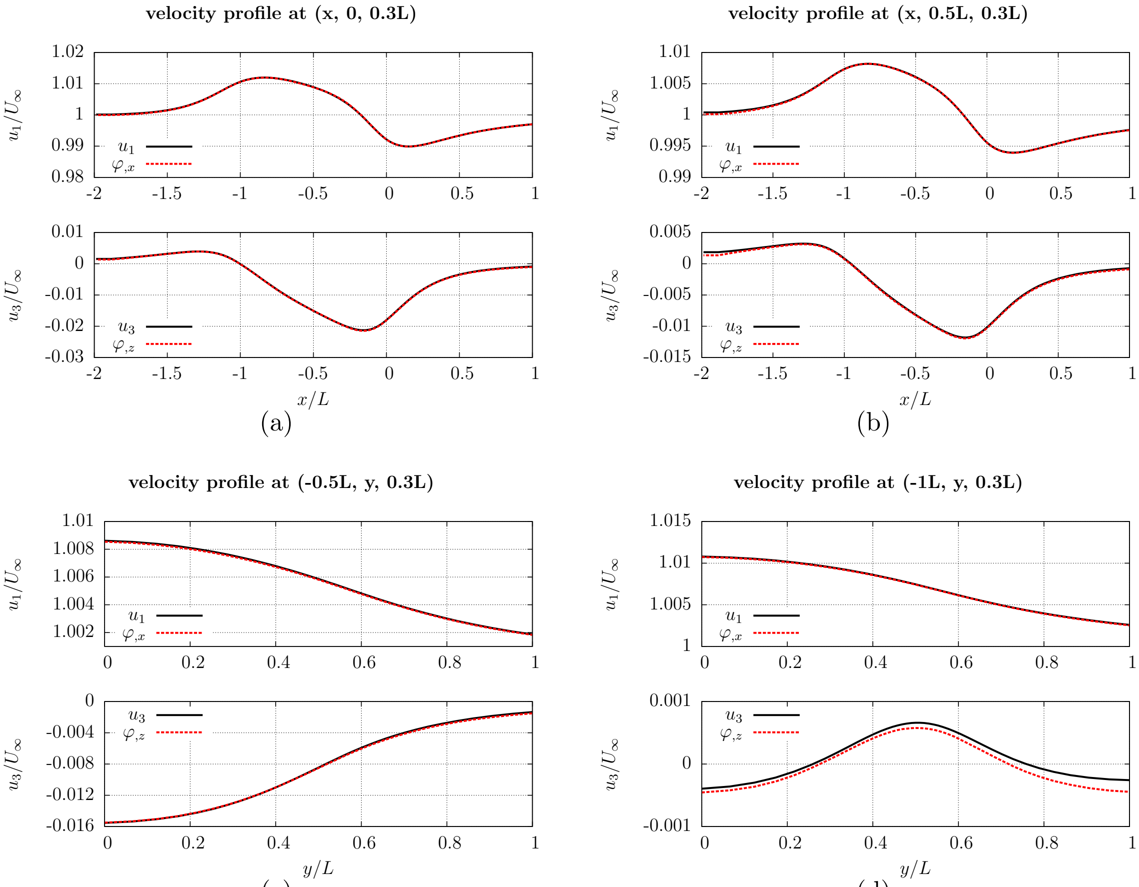


Figure 4.3: Contour plots of the magnitude of the vortical velocities $\mathbf{w}^*, \mathbf{w}^*_{\text{inv}}$ and vorticity $\boldsymbol{\omega}^*$ (Blue line marks the location of the flat plate, superscript "$*$" denotes non-dimensionalized values)



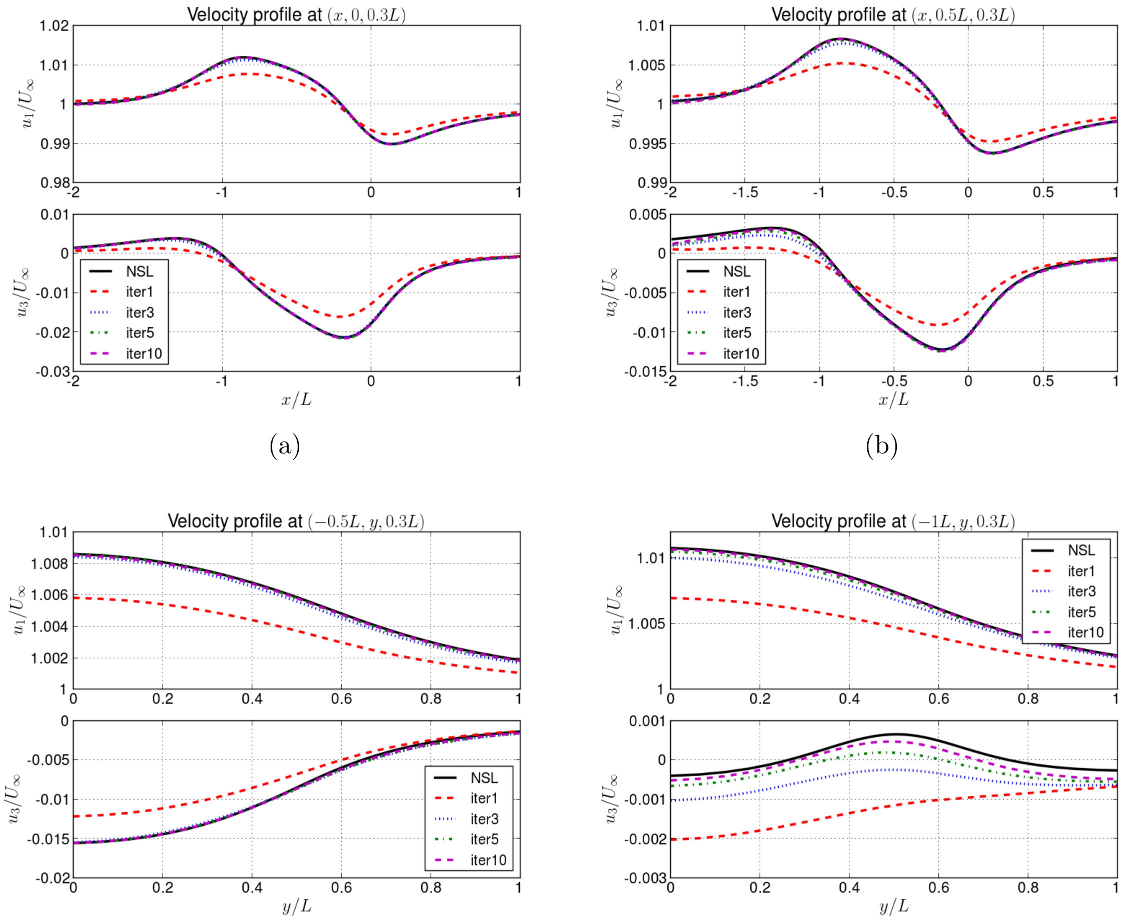
(a)

(b)

(c)

(d)

Figure 4.4: Streamwise and vertical velocity profile comparison



(a)

(b)

(c)

(d)

Figure 4.5: Streamwise and vertical velocity profile at different iterations

are compared with the NSL solutions in Fig. 4.5. As shown, the viscous-potential velocity converges to the NSL solution mostly after three iterations. This is consistent with our previous findings *Edmund* (2012a); *Rosemurgy* (2014) and the analysis in Section 3.2.1.1.

4.1.2 Influence of the Inlet Boundary Condition

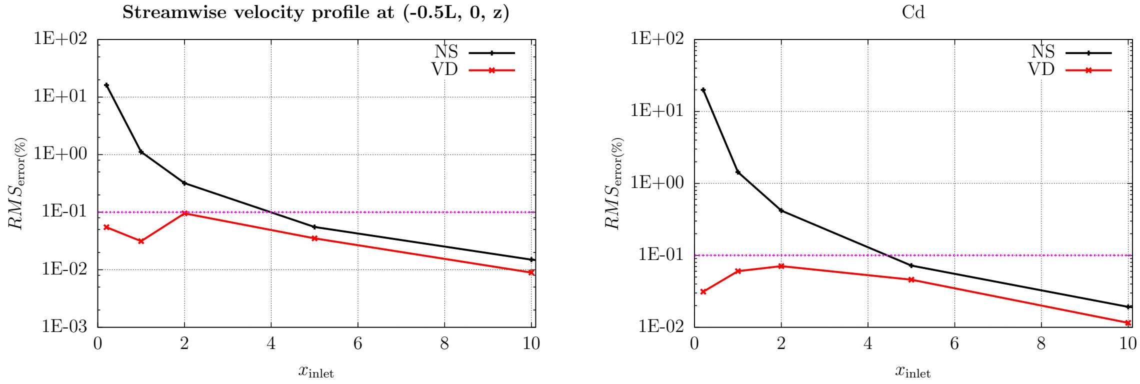
As described in Chapter II, the velocity tends to the free-stream value infinitely far from the body. However, with conventional field methods, it is not possible to discretize an infinite domain, so one is forced to determine a sufficiently far distance to place the external boundaries. Hence, this leads to a relatively large domain and more computation is required. Some researchers have applied the inviscid-potential velocity as the inlet boundary condition instead of the free-stream velocity to improve the solution on a modest size domain *Eça and Hoekstra* (2009). Even though the inviscid-potential velocity can provide some improvement on the solutions, the improvement is limited because the inviscid-potential velocity does not account for the viscous effects unlike the viscous-potential velocity being able to fully describe the flow field outside the vortical region.

The results of the inlet-location-dependence study for the conventional solver and the velocity decomposition solver are shown in Fig. 4.6. For all the cases in the study, the outlet location is fixed at one body length downstream $x_{\text{outlet}} = 1L$, and the inlet location is vaired as $x_{\text{inlet}} = 0.2L, 1L, 2L, 5L, 10L$. All the meshes are identical in their common regions. In Fig. 4.6, the RMS(Root-Mean-Square) error of the streamwise velocity profile at the center of the flat plate and the drag coefficient are shown. The RMS error is calculated using Eq. (4.1). The NSL Solution ($x_{\text{inlet}} = 50L$) is used as the benchmark solution. The RMS error of the velocity profile shown in Fig. 4.6 is calculated using thirty data points on the velocity profile sampling at the center of the flat plate from $(-0.5L, 0, 0)$ to $(-0.5L, 0, 0.9)$.

$$RMS_{\text{error}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{u_{x,i} - u_{\text{NSL,i}}}{u_{\text{NSL,i}}}\right)^2} \times 100\% \tag{4.1}$$

The results generated by using a conventional solver with the free-stream velocity inlet boundary condition are shown in black and denoted as "NS". The results generated by the velocity decomposition coupled solver (using viscous-potential velocity as inlet boundary condition) are shown in red and denoted as "VD".

As can be seen for both cases, the error generally decreases as x_{inlet} increases. For all values of x_{inlet} the error is smaller for the VD results. This demonstrates that the influence of a flat plate boundary layer is measurable a large distance from



(a)

(b)

Figure 4.6: RMS error for velocity profiles and drag coefficient of inlet-extent(x_{inlet}) study

the plate, and the VD solution that satisfies the radiation condition naturally does not impose any domain blockage on the solution. When using a conventional field method it is necessary to use a computational domain that extends at least several body lengths from the plate, depending on the desired accuracy that one seeks in the solution. On the other hand, when x_{inlet} is small (e.g. $x_{\text{inlet}} \leq 1L$), the accuracy of the solutions is substantially improved by using the velocity decomposition solver, as shown in red. The slow decay of the perturbation due to the body boundary can also be seen in Fig. 4.3. As shown in more detail in Section 4.1.3, the velocity decomposition solver can retain the accuracy of the solution when solving the same case on a greatly reduced domain. The rise in RMS error around $x_{\text{inlet}} = 2L$ may be due to the discretization error in the BEM solver, the iteration error from the iterative viscous boundary condition, or noise that is introduced by interpolating onto a set of points, or having error in the NSL solution. These errors can be reduced by using a finer discretization in the BEM solver and more iterations when enforcing the iterative viscous boundary condition.

4.1.3 Velocity Decomposition for Transient Flow Simulation

The velocity decomposition coupled solver is applied to solve the same case on a greatly reduced domain. The mesh for the reduced domain is identical to the one used for the NSL solutions, in the common regions. The details of the mesh for the reduced domain are shown in Table 4.3. The inlet boundary condition is updated using the viscous-potential velocity every 0.1 flow-over ($\Delta t_{\text{update}} = 0.1t^* = 0.1L/U_\infty$).

x_{inlet}	x_{outlet}	total # of cells
$0.2L$	$1L$	40,095

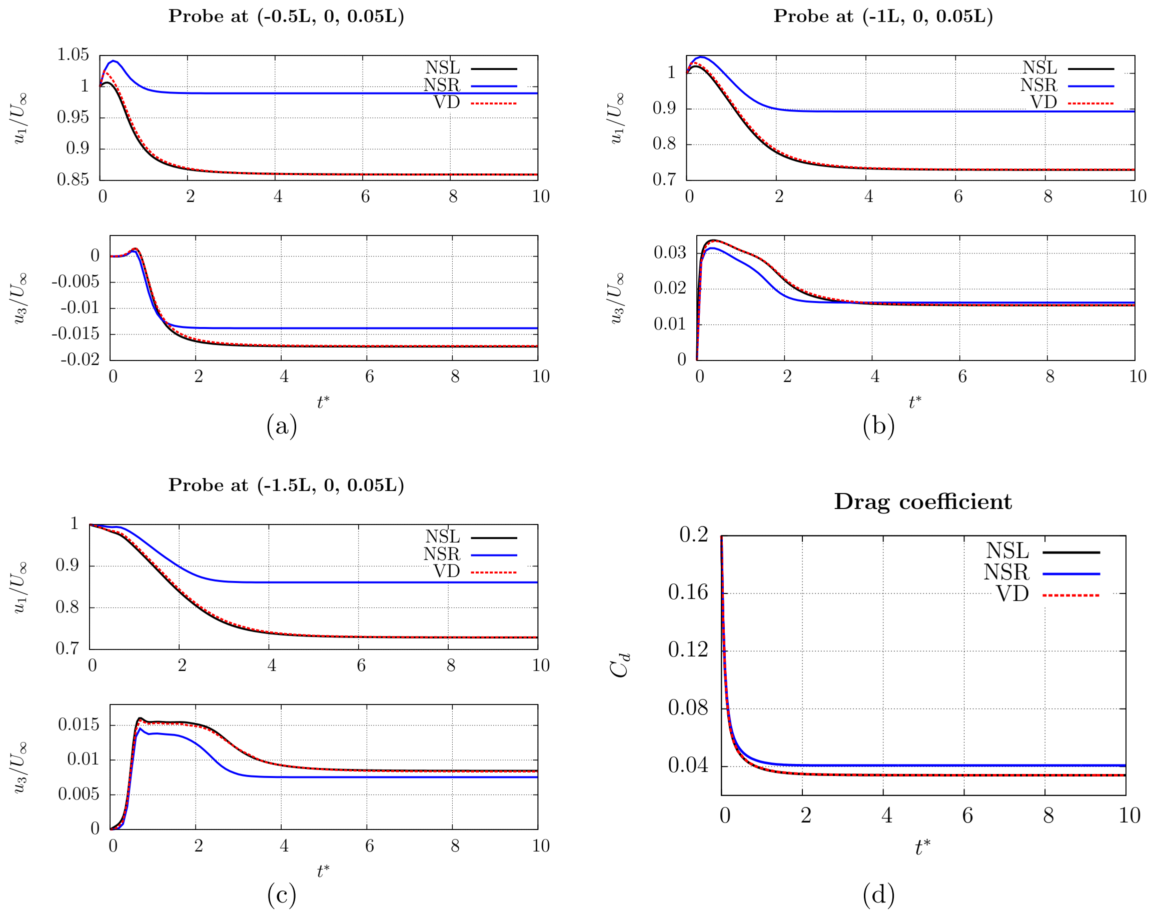
Table 4.3: Discretization in OpenFOAM for the reduced domain for laminar flow over a finite flat plate

The time histories of the streamwise and vertical velocity at three different locations, and the drag coefficient, are compared with the NSL solutions in Fig. 4.7. The Navier-Stokes solutions calculated on the reduced domain using the free-stream velocity as the inlet boundary conditions, denoted as NSR, are plotted to demonstrate the improvement of the VD solutions. As can be seen, the VD solutions agree well the NSL solutions after several updates are applied, while the NSR solutions fail to do so. Hence, the velocity decomposition solver is able to solve the same case accurately on a greatly reduced domain.

The total CPU time for the NSL solution equals roughly 31.78 hours, while the one for the VD solution with 100 updates is about 2.35 hours (Table 4.4). The CPU time is calculated by multiplying the walltime with the number of processors used to compute the solution for 10 flow-overs. The averaged time for each update (solving the viscous potential and applying it to update the inlet boundary condition) is about

NSL-total	VD-total	avg(VD-update)
31.78	2.35	0.019

Table 4.4: The comparison of the CPU hours for laminar flow over a finite flat plate



(a)

(b)

(c)

(d)

Figure 4.7: The comparison of streamwise, vertical velocity and drag coefficient time histories

0.019 hours. It should be noted that the improvement of the computational efficiency for the VD solver is currently not optimized. The improvement can be potentially improved by reducing the amount of file I/O in the numerical implementation of the coupling. Another way to achieve further computational saving is to apply fewer updates, since after 40 updates the viscous potential has converged to a steady state. Currently, the coefficient matrix for evaluating the velocity at the inlet boundary is re-formulated at the beginning of each update. When the distance between the body and the inlet boundary is not changed, this coefficient matrix can be stored and reused for the following updates and this allows additional computational savings.

4.2 Turbulent Flow over a Finite Flat Plate

In this section, turbulent flow (Re $= 5\times10^6$) over a finite-length flat plate mounted on a slip wall is studied. The computational domains in OpenFOAM and Aegir are similar to the ones in the laminar case, shown in Fig. 4.1. The discretization for both the FVM and the BEM are refined for the turbulent flow. The details of the discretization used in OpenFOAM for the NSL solution and in Aegir for the viscous potential are shown in Table 4.5 and Table 4.6. The mesh in OpenFOAM is made such that the average y^+ is approximately one and no wall model is applied. The k-ω SST turbulence model described in Section 3.1.1, is used.

y^+_{avg}	x_{inlet}	x_{outlet}	total # of cells
1.02	2L	0.5L	5,154,750

Table 4.5: Discretization in OpenFOAM for the NSL solution for turbulent flow over a finite flat plate

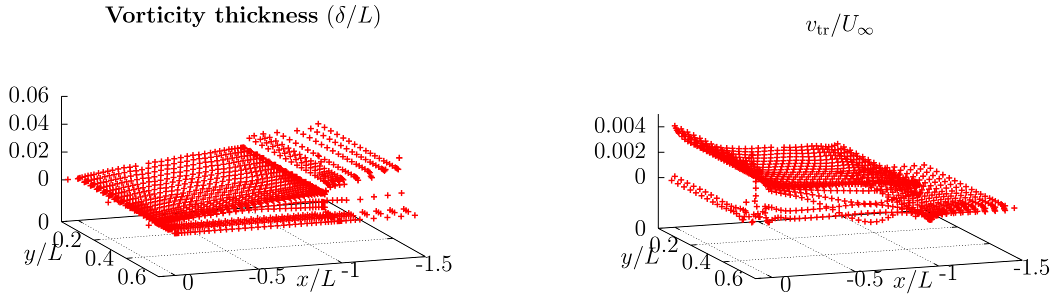
wake-panel extent	total # of panels	# of body panels	# of wake panels
0.5L	162	50	20

Table 4.6: Discretization in Aegir for the viscous potential for turbulent flow over a finite flat plate

4.2.1 Viscous Potential

The inviscid-potential velocity is the free-stream velocity, while the viscous potential accounts for the effects caused by the boundary layer and the wake. Both the vorticity thickness δ and transpiration velocity introduced by the viscous potential are shown in Fig. 4.8. As can be seen, the vorticity thickness δ is about a third of the one in laminar case and the transpiration velocity is roughly 20 times smaller than in laminar case.

Fig. 4.9 shows the contour plots of the vortical velocity on the centerplane of the flow domain. As shown in upper left of Fig. 4.9, the region where the vortical



(a)

(b)

Figure 4.8: Vorticity thickness δ and transpiration velocity v_{tr} due to the boundary layer and wake for turbulent flow over a finite flat plate

velocity $\mathbf{w}^*$ is noticeable matches with the vortical region shown in the upper right. The inviscid vortical velocity shown in the lower left of Fig. 4.9, persists about half a body length away from the body.

The velocity profile comparison of the total velocity and the viscous-potential velocity is shown in Fig. 4.10. As can be seen, both the streamwise and vertical components of the viscous-potential velocity agree with the total velocity. The small difference in the vertical velocity profile shown in Fig. 4.10 may be related to the vorticity filter value used for determining the δ value. However, the difference is very small, about $0.02\%U_\infty$.

4.2.2 Velocity Decomposition for Transient Flow Simulation

The velocity decomposition coupled solver is applied to solve the same case on a greatly reduced domain. The mesh for the reduced domain is identical to the one of the NSL solutions in the common regions. The details of the mesh for the reduced domain are shown in Table 4.7. The inlet boundary condition is updated with the viscous-potential velocity every 0.1 flow-over ($\Delta t_{\mathrm{update}} = 0.1t^* = 0.1L/U_\infty$).

The time histories of the streamwise and vertical velocity at three different locations, and the drag coefficient, are compared with the NSL solutions in Fig. 4.11.

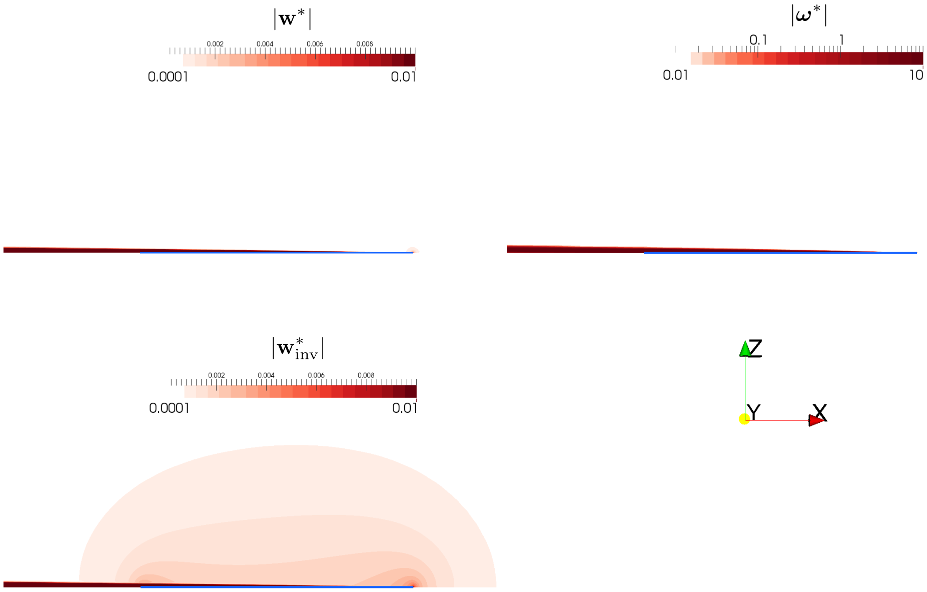


Figure 4.9: Contour plots of the magnitude of the vortical velocities $\mathbf{w}^*, \mathbf{w}^*_{\text{inv}}$ and vorticity $\boldsymbol{\omega}^*$ (Blue line marks the location of the flat plate, superscript "$*$" denotes non-dimensionalized values)

x_{inlet}	x_{outlet}	total # of cells
$0.1L$	$0.5L$	$1,287,792$

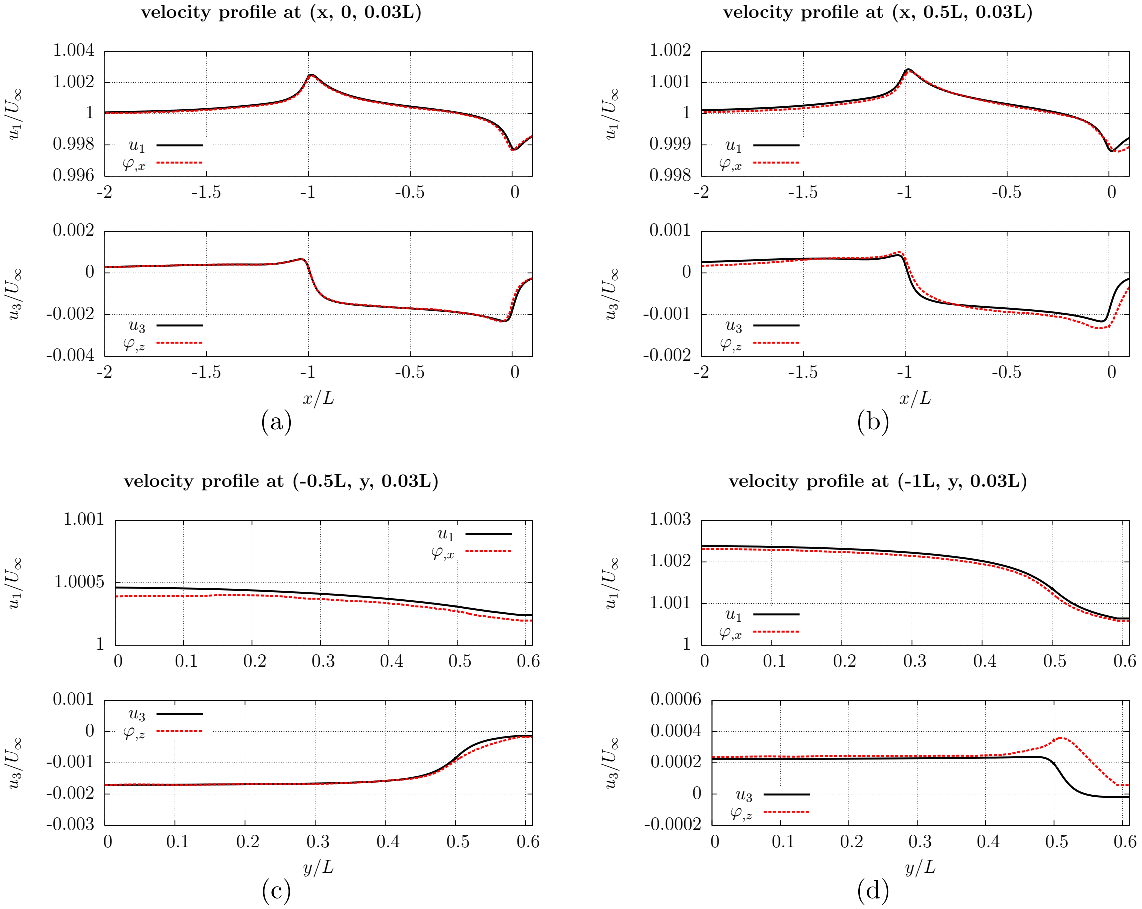
Table 4.7: Discretization in OpenFOAM for the reduced domain for turbulent flow over a finite flat plate

The NSR solutions are plotted to show the improvement of the VD solutions. As can be seen, the VD solutions agree well the NSL solutions after several updates are applied, while the NSR solutions do not match with the NSL solutions. The resolution of the time dependence of the flow field can be improved by using a smaller update interval Δt_{update}. From the results shown in Fig. 4.11, the velocity decomposition solver demonstrates the ability to conduct URANS calculations accurately on a greatly reduced domain for turbulent flow over a finite flat plate.

NSL-total	VD-total	avg(VD-update)
1410.73	461.61	0.069

Table 4.8: The comparison of the CPU hours for turbulent flow over a finite flat plate

The total CPU time for the NSL solution equals roughly 1410.73 hours, while the



(a)

(b)

(c)

(d)

Figure 4.10: Streamwise and vertical velocity profile comparison

one for the VD solution with 50 updates is about 461.61 hours (Table 4.8). The CPU time is calculated by multiplying the walltime with the number of processors used to compute the solution for 10 flow-overs. The averaged time for each update (solving the viscous potential and applying it to update the inlet boundary condition) is about 0.069 hours. It should be noted that the improvement of the computational efficiency for the VD solver is currently not optimized. The improvement can be potentially improved by reducing the amount of file I/O in the numerical implementation of the coupling. Another way to achieve further computational saving is to apply fewer updates, since after 20 updates the viscous potential has converged to a steady state. Currently, the coefficient matrix for evaluating the velocity at the inlet boundary is re-formulated at the beginning of each update. When the distance between the body and the inlet boundary is not changed, this coefficient matrix can be stored and reused for the following updates and this allows additional computational savings.

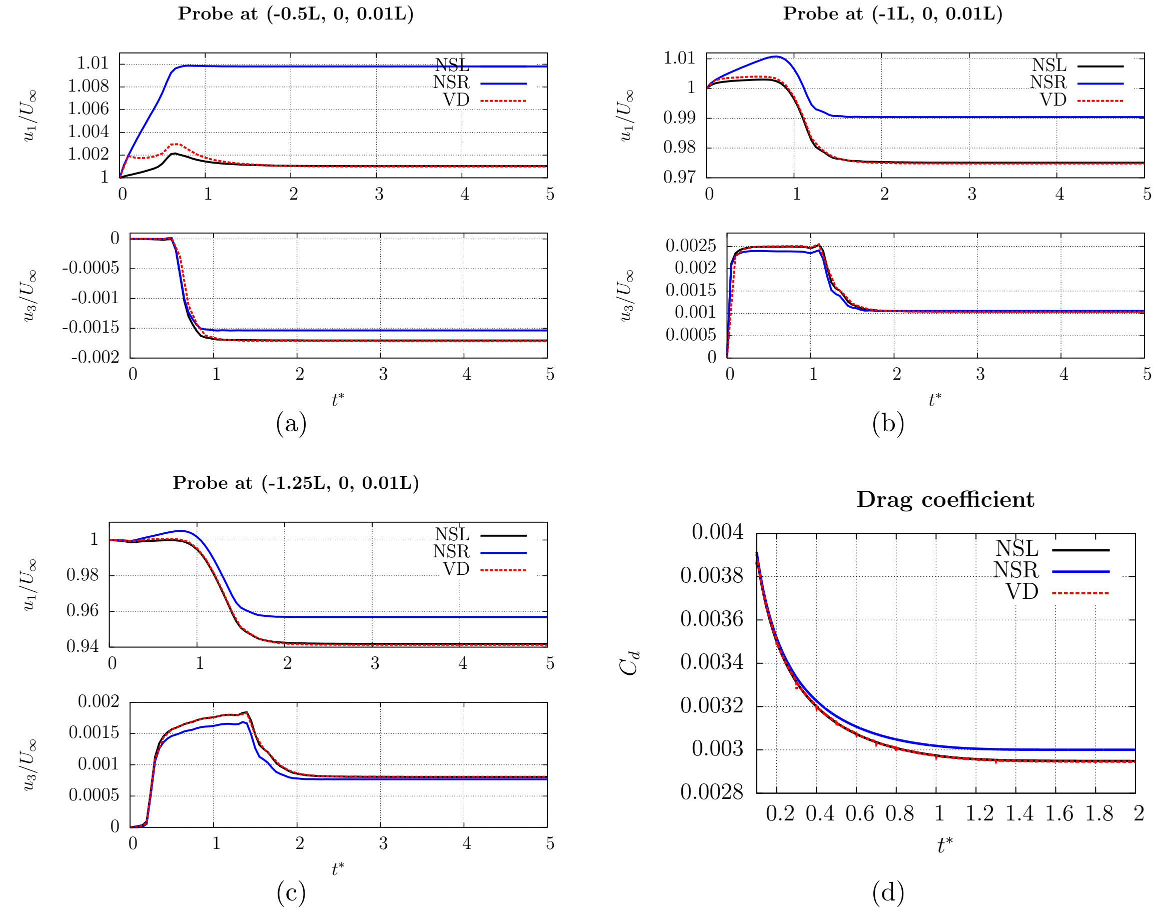


Figure 4.11: The comparison of streamwise, vertical velocity and drag coefficient time histories

4.3 Turbulent Flow over a Wigley Hull

In this section, impulsively started turbulent flow (Re $= 4 \times 10^6$) over a Wigley hull is studied. The computational domains in OpenFOAM and Aegir are shown in Fig. 4.12. The origin of the coordinate system is located at the center of the Wigley hull. A symmetry boundary condition is applied on the top boundary S_{t} (no free-surface effect is considered in this study). S_{I} is the inlet boundary and S_{o} is the outlet boundary. In Fig. 4.12(b), the body panels are colored in yellow and the wake panels are in blue. The turbulence is modeled through a LES and the one equation eddy viscosity model described in Section 3.1.1, is used to account for the unresolved turbulence scales.

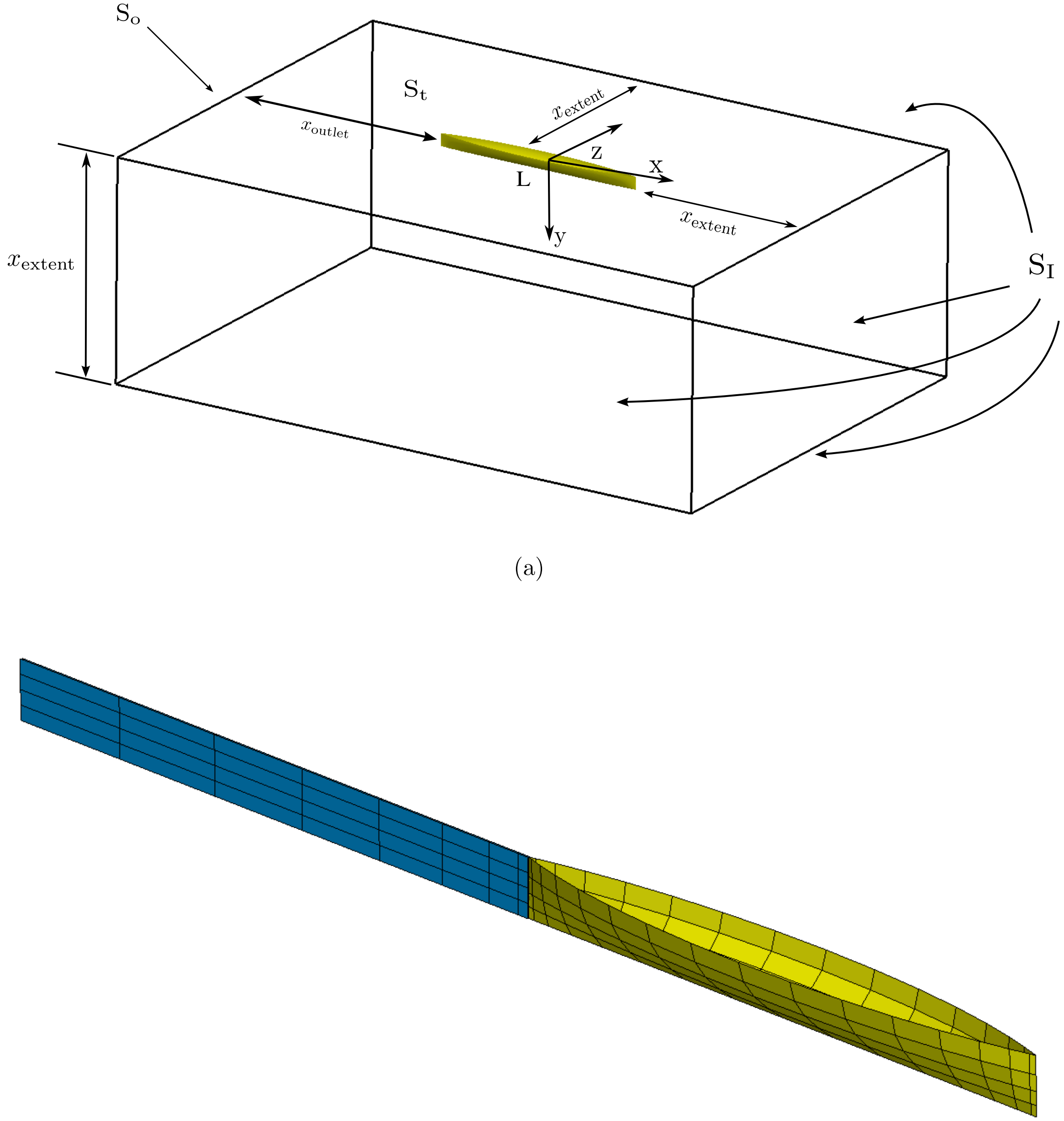
The details of the discretization used in OpenFOAM for the NSL solution and in Aegir for the viscous potential are shown in Table 4.9 and Table 4.10. The mesh in OpenFOAM is made such that the average y^+ is approximately 10. A wall function using Spalding's universal law is applied on the body boundary. It is worth noting that the FVM mesh used for this test case is overly coarse to be considered for an accurate LES calculation. However, the focus of this work is to demonstrate the capability of the velocity-decomposition approach to be applied with different turbulence modeling techniques. To conduct a valid LES simulation, at least 85% of the turbulent kinetic energy should be resolved.

avg(y^+)	x_{inlet}	x_{outlet}	total # of cells
9.91	$5L$	$1L$	3,323,018

Table 4.9: Discretization in OpenFOAM for the NSL solution for turbulent flow over a Wigley hull

wake-panel extent	body-panel #	total # of panels
$1L$	128	192

Table 4.10: Discretization in Aegir for the viscous potential for turbulent flow over a Wigley hull



(a)

(b)

Figure 4.12: Sketch of the computational domain: (a) in OpenFOAM; (b) in Aegir

4.3.1 Viscous Potential

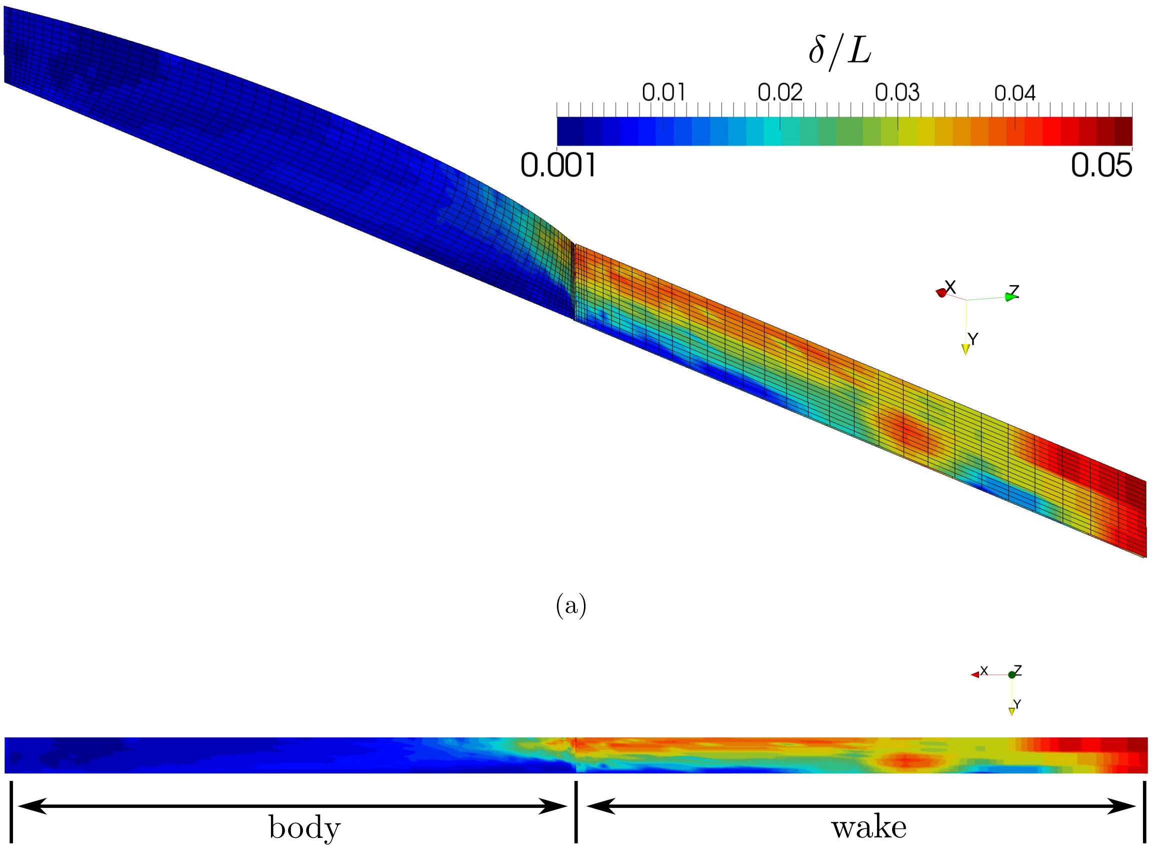
The viscous-potential velocity used in this section is calculated based on the fully developed NSL solutions (after 20 flow-overs). The contour of vorticity thickness δ is plotted on the body and wake surfaces, shown in Fig. 4.13 (a). The perspective and the side views are shown in the upper and lower figure of Fig. 4.13 (a). As expected, δ takes the shape of the boundary layer and the vortical wake. δ is small before the mid-ship, where the flow is attached and boundary layer is thin. After mid-ship, the boundary layer thickens, the flow starts to separate and δ grows larger.

The perspective and side view of the viscous-potential velocity contour on the body and wake surfaces are shown in Fig. 4.14(a) and(b) respectively. The inviscid-potential velocity contour is shown in Fig. 4.14(c). As expected, the inviscid-potential velocity is symmetry on the body. The transpiration velocity introduced by the viscous potential to account for the viscous effect, can be calculated as the difference between the viscous-potential and inviscid-potential velocity Eq. (4.2).

$$v_{\text{tr}} = \frac{\partial \varphi}{\partial n} - \frac{\partial \phi_{\text{inv}}}{\partial n} \qquad \text{on } S_B \text{ and } S_W \tag{4.2}$$

The contour of the transpiration velocity is shown in Fig. 4.14 (d). As can be seen, the transpiration velocity is large at the stern and wake regions to model the viscous effects. It is worth noting that the unsteady patterns at the wake surface for both δ and $\partial_n \varphi$ indicate that viscous potential captures the viscous effects at the instantaneous flow field. The vortical region at the same time instance is shown in the upper right contour plot of Fig. 4.15.

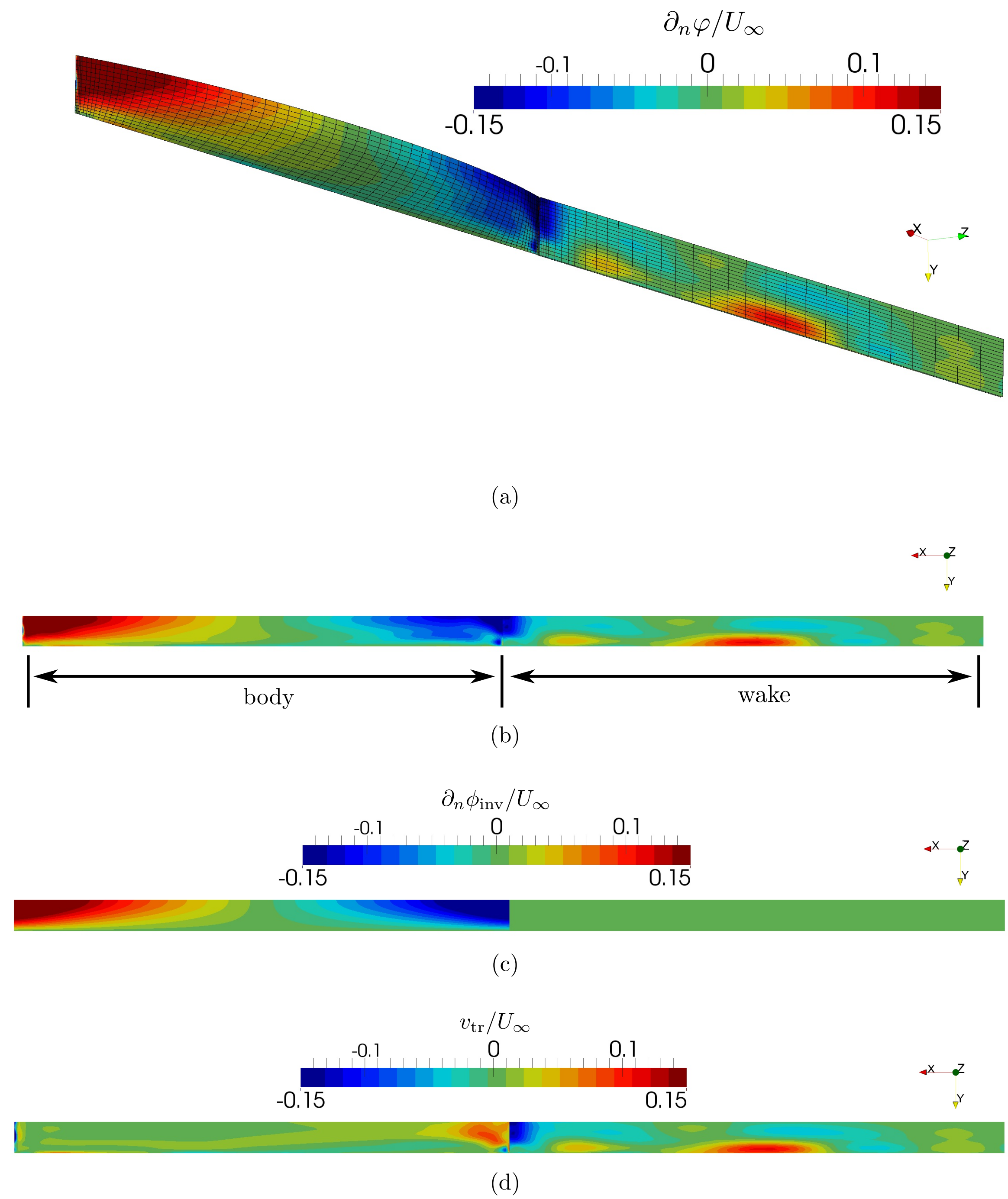
The two vortical velocities $\mathbf{w}^*$ and $\mathbf{w}^*_{\text{inv}}$ are compared in Fig. 4.15. A slice at the calm-water plane and the center plane of the contour plots are shown in Fig. 4.15 (a) and (b) respectively. The vortical region is shown in the vorticity magnitude contour plot at the upper right of Fig. 4.15 (a) and (b). As can be seen, $\mathbf{w}^*$ decreases rapidly



(a)

(b)

Figure 4.13: Vorticity thickness δ for turbulent flow over a Wigley hull: (a) perspective view (b) side view



(a)

(b)

(c)

(d)

Figure 4.14: (a) perspective view (b) side view of viscous-potential velocity; (c) inviscid-potential velocity (d) transpiration velocity for turbulent flow over a Wigley hull

away from the body and wake, while $\mathbf{w}^*_{\mathbf{inv}}$ persists about half body length away from the body and wake.

The viscous-potential velocity profile at four different locations are compared with the total velocity from the NSL solution in Fig. 4.16. As can be seen, both the streamwise and vertical components of the viscous-potential velocity agree well with the total velocity.

The velocity profiles of the viscous-potential velocity after different number of iterations are compared with the NSL solutions in Fig. 4.17. As shown, the viscous-potential velocity converges to the NSL solution mostly after three iterations. This is consistent with the analysis in Section 3.2.1.1.

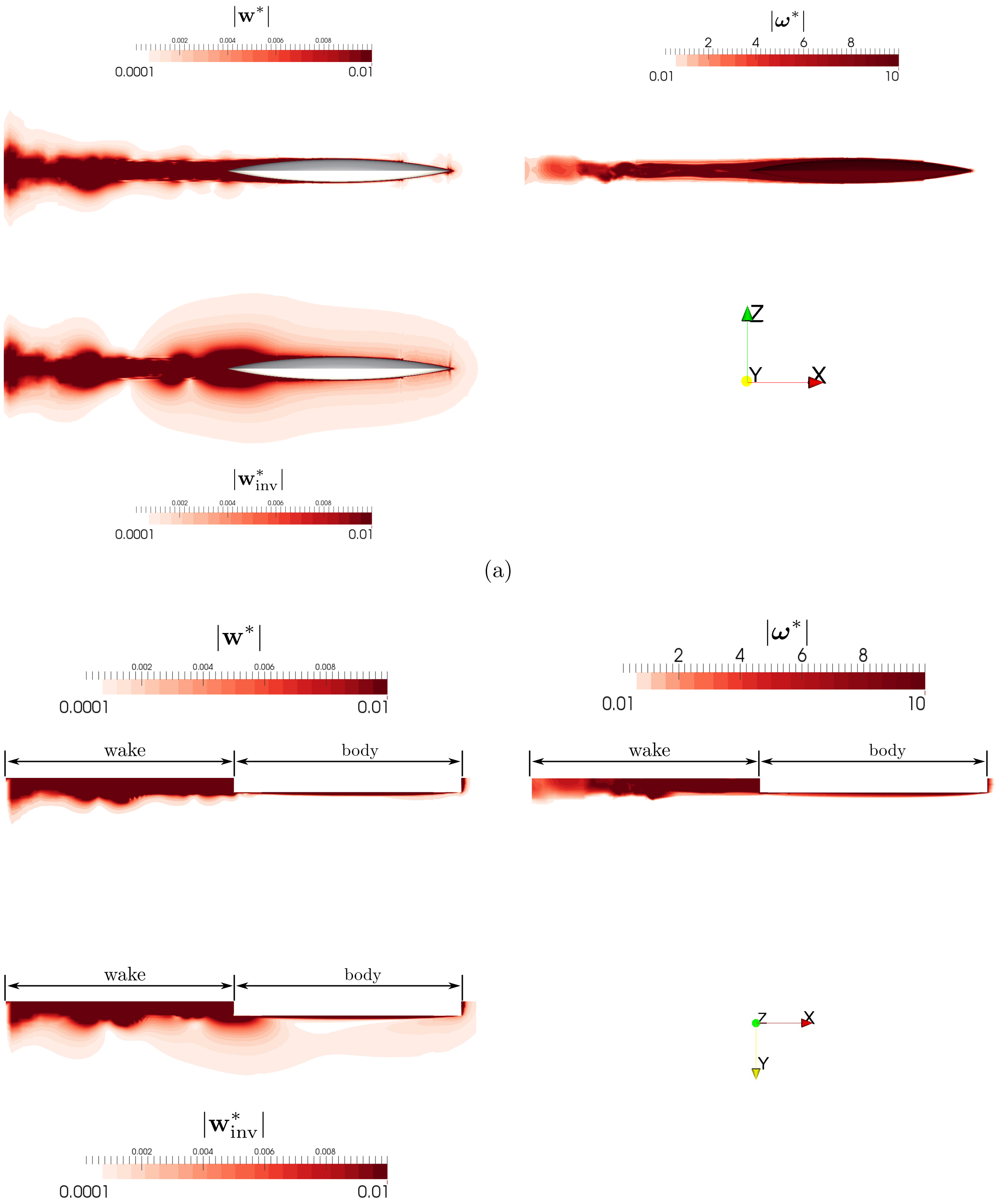
4.3.2 Velocity Decomposition for Transient Flow Simulation

In this section the velocity decomposition solver is applied to solve the same case on a greatly reduced domain. The mesh for the reduced domain is identical to the one of the NSL solutions in their common regions. The details of the mesh for the reduced domain are shown in Table 4.11. The inlet boundary condition is updated with the viscous-potential velocity every 0.1 flow-over ($\Delta t_{\text{update}} = 0.1t^* = 0.1L/U_\infty$).

x_{inlet}	x_{outlet}	total # of cells
$0.05L$	$1L$	$1,826,101$

Table 4.11: Discretization in OpenFOAM for the reduced domain for turbulent flow over a Wigley hull

The time histories of the streamwise and vertical velocity at three different locations, and the drag coefficient, are compared with the NSL solutions in Fig. 4.18. As can be seen, the VD solutions agree well the NSL solutions after several updates are applied. The NSR solutions again do not match with the NSL solutions. The Navier-Stokes solutions calculated on the reduced domain using the inviscid-potential velocity as the inlet boundary conditions, denoted as invPot, are plotted also. As can



(a)

(b)

Figure 4.15: Contour plots of the magnitude of the vortical velocities $\mathbf{w}^*, \mathbf{w}^*_{\text{inv}}$ and vorticity $\boldsymbol{\omega}^*$ for turbulent flow over a Wigley hull: (a) the slice at the calm-water plane (b) the slice at the center plane (superscript "$*$" denotes non-dimensionalized values)

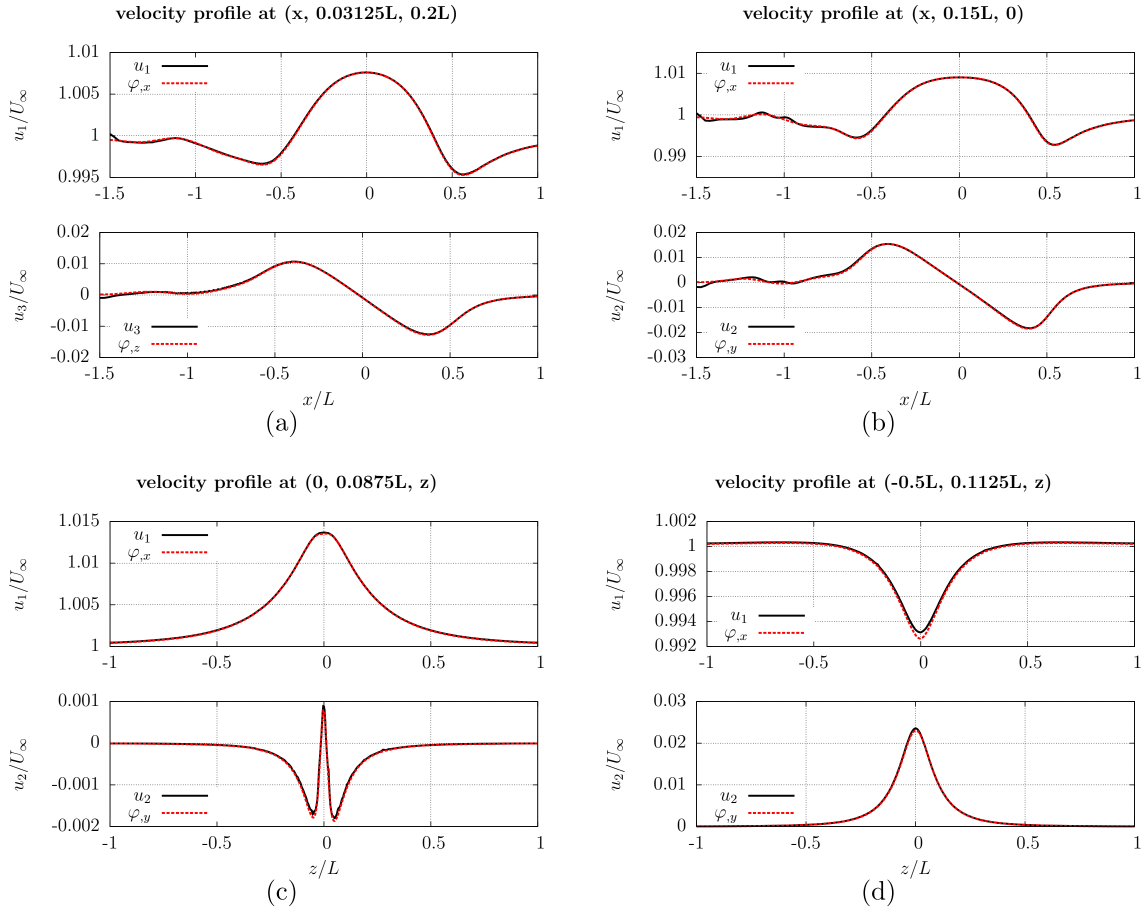
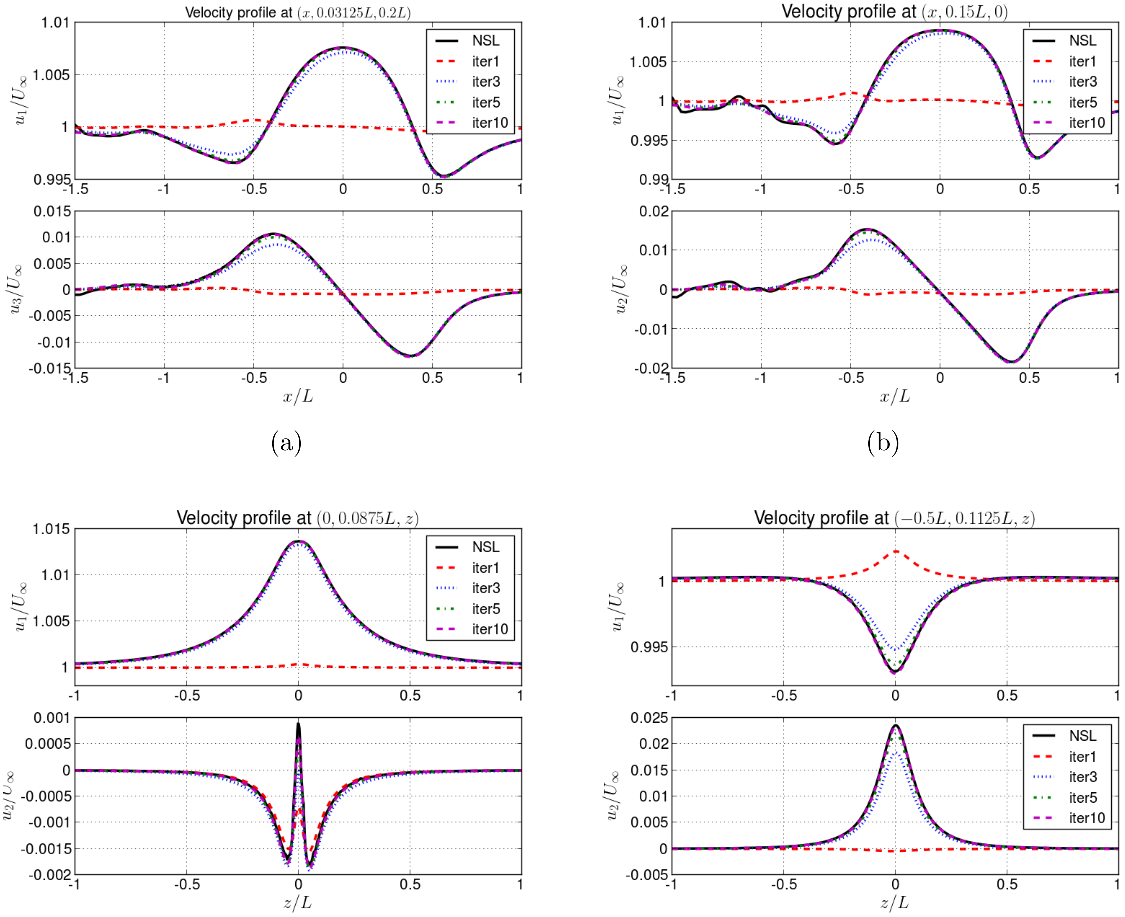


Figure 4.16: Streamwise and vertical velocity profile comparison



(a)

(b)

(c)

(d)

Figure 4.17: Streamwise and vertical velocity profile at different iterations

be seen, the invPot solutions show some improvements comparing to the NSR solutions. However the improvements are limited and the results using the viscous potential match well with the NSL solutions.

NSL-total	VD-total	avg(VD-update)
$3,598$	$1,898$	0.048

Table 4.12: The comparison of the CPU hours for turbulent flow over a Wigley hull

The total CPU time for the NSL solution equals roughly 3,598 hours, while the one for the VD solution with 50 updates is about 1,898 hours (Table 4.12). The CPU time is calculated by multiplying the walltime with the number of processors used to compute the solution for 10 flow-overs. The averaged time for each update (solving the viscous potential and applying it to update the inlet boundary condition) is about 0.048 hours. The improvement of the computational efficiency for the VD solver is currently not optimized. The improvement can be potentially improved by reducing the amount of file I/O in the numerical implementation of the coupling. Another way to achieve further computational saving is to apply fewer updates, since after 20 updates the viscous potential has converged to a steady state. Currently, the coefficient matrix for evaluating the velocity at the inlet boundary is re-formulated at the beginning of each update. When the distance between the body and the inlet boundary is not changed, this coefficient matrix can be stored and reused for the following updates and this allows additional computational savings.

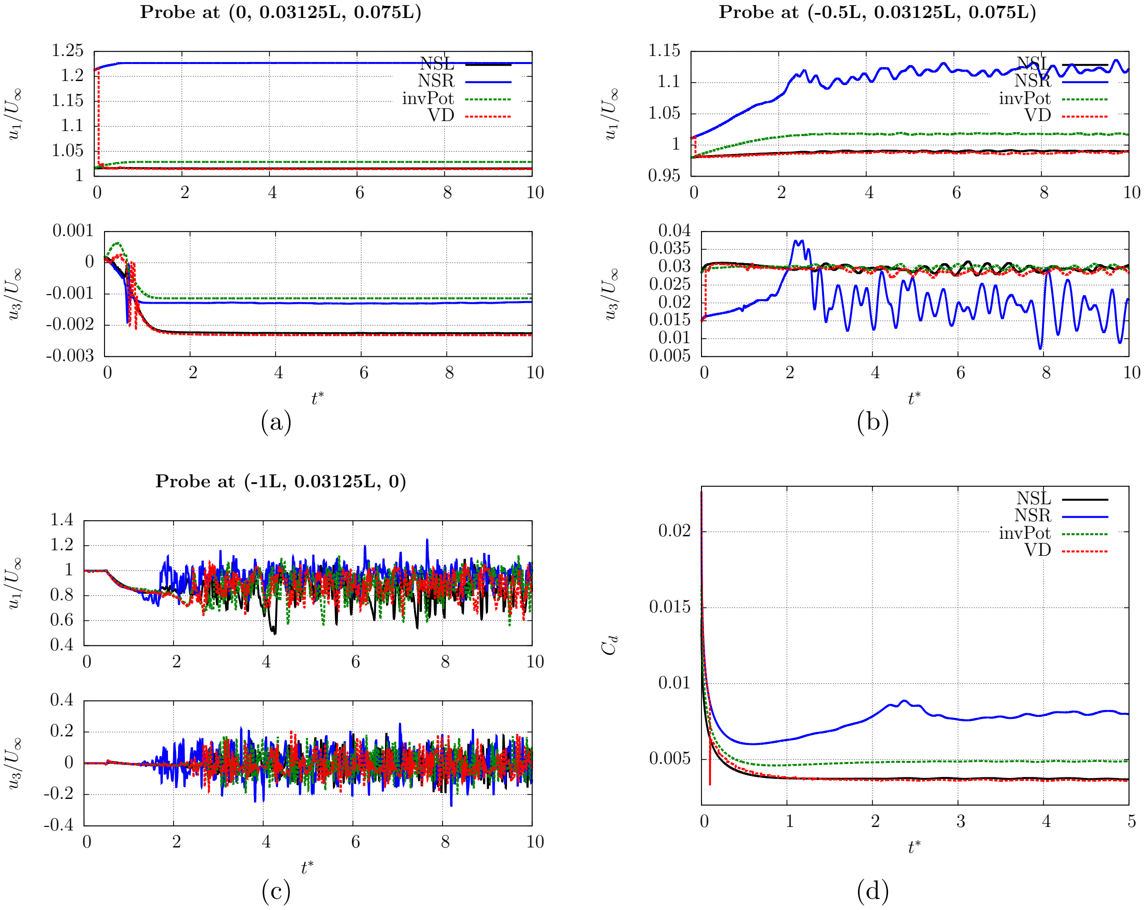


Figure 4.18: The comparison of streamwise, vertical velocity and drag coefficient time histories

CHAPTER V

Flow over Ship Hulls

The ultimate applications for our velocity decomposition solver are simulations for naval vessels or marine structures that operate near or on the free-surface. To be able to accomplish this, the first step is for the solver to have the capability to handle complicated geometries, like ship hulls. In this section, two canonical ship hulls, the DTMB (David Taylor Model Basin) 5415, and KVLCC2 (KRISO Very Large Crude Carrier) ship hulls are studied. The study of the flow field around the ship hulls is important for various aspect of the design process, especially for predicting ship resistance and the design of the propeller. The flow may separate near the stern at the propeller plane, which means the inflow condition for the propeller operating condition is complicated. It is important to calculate the flow field accurately so that the best design decisions can be made for the propeller. In this section, LES is selected to perform the calculation with the VD solver.

For each test case, the benchmark solutions are calculated by using a conventional FVM Navier-Stokes solver on a large computational domain (denoted as NSL). The viscous-potential velocity is first studied, then the velocity decomposition coupled solver is used to solve for the case on a greatly reduced domain. The solutions from the coupled solver (denoted as VD) are compared with the benchmark solutions.

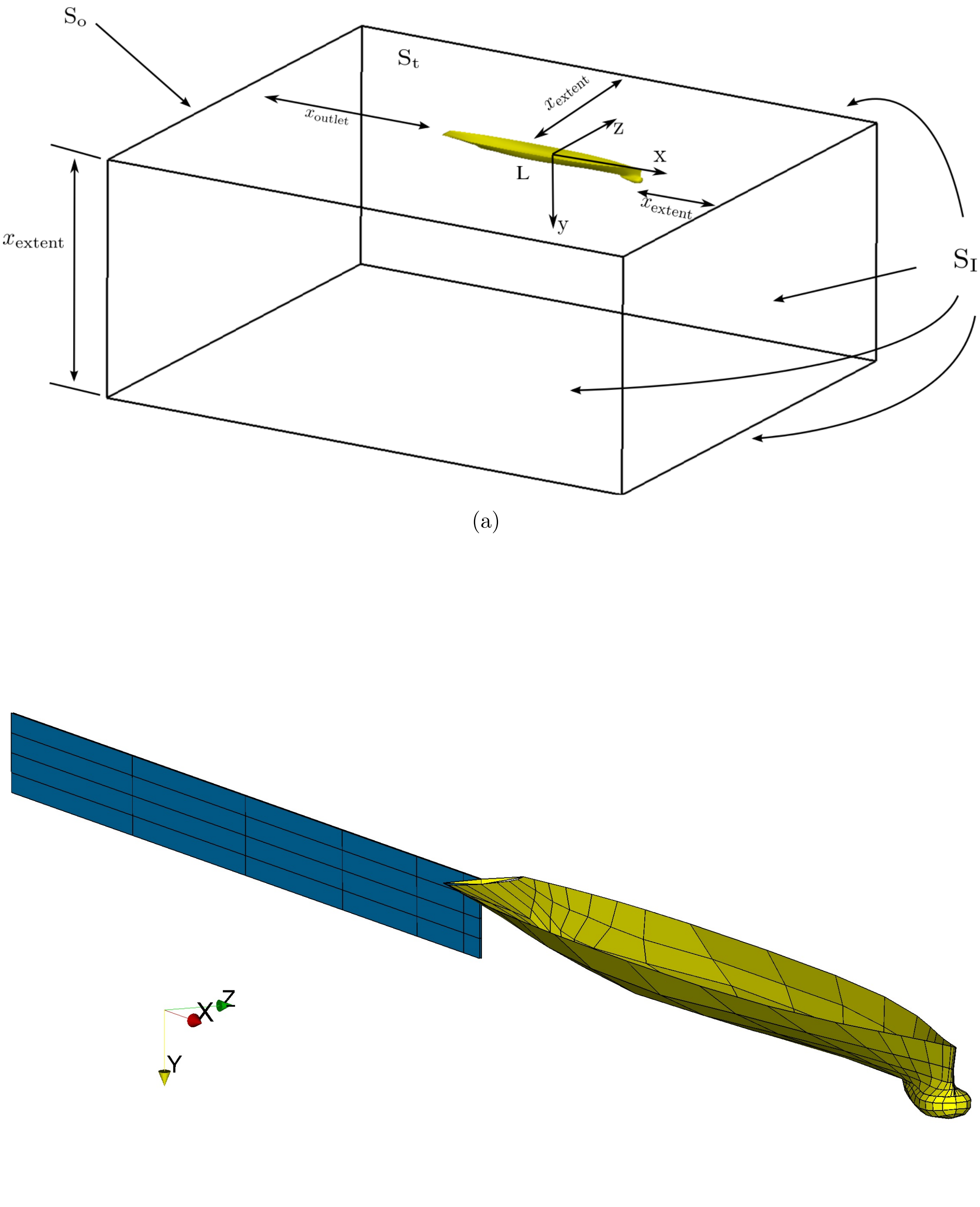
5.1 Turbulent Flow over a DTMB 5415 Ship Hull

In this section, impulsively started turbulent flow ($\mathrm{Re} = 5.13 \times 10^6$) over a DTMB ship hull is studied. With a sonar dome and transom stern, the DTMB 5415 ship hull represents a typical naval combatant. The computational domains in OpenFOAM and Aegir are shown in Fig. 4.12. The origin of the coordinate system is located at the center of the ship hull. A symmetry boundary condition is applied on the top boundary S_t (no free-surface effect is considered in this study). S_I is the inlet boundary and S_o is the outlet boundary. In Fig. 4.12(b), the body panels are colored in yellow and the wake panels are in blue. The turbulence is modeled through LES and the one equation eddy viscosity model is used to account for the unresolved turbulence scales.

The details of the discretization used in OpenFOAM for the NSL solution and in Aegir for the viscous potential are shown in Table 5.1 and Table 5.2. The mesh in OpenFOAM is made such that the average y^+ is approximately 6. A wall function using Spalding's universal law is applied on the body boundary. It is worth noting that the FVM mesh used for this test case is overly coarse to be considered for an accurate LES calculation. However, the focus of this work is to demonstrate the capability of the velocity-decomposition approach to be applied with different turbulence modeling techniques. To conduct a valid LES simulation, at least 85% of the turbulent kinetic energy should be resolved.

y^+_avg	x_inlet	x_outlet	total # of cells
5.7	$5L$	$1L$	$4,026,599$

Table 5.1: Discretization in OpenFOAM for the NSL solution for turbulent flow over a DTMB 5415 hull



(a)

(b)

Figure 5.1: Sketch of the computational domain: (a) in OpenFOAM; (b) in Aegir

wake panel extent	body panels #	total # of panels
1L	240	288

Table 5.2: Discretization in Aegir for the viscous potential for turbulent flow over a DTMB 5415 hull

5.1.1 Viscous Potential

The viscous-potential velocity calculated based on the fully developed NSL solution (after 20 flow-overs) is studied in this section. The perspective, side and bottom views of the contour plot of the vorticity thickness δ are shown in Fig. 5.2. As shown in Fig. 5.2, δ is small over most of the body, except the area directly after the sonar dome and the stern region. In the wake region, δ takes a much larger value ($\delta \sim B/2$), which is of the order of magnitude of half ship beam B. As can be seen in the bottom view of Fig. 5.3, the viscous-potential velocity introduces influx at the stern region to model viscous effects.

The perspective view of the viscous-potential velocity on the body and the wake surfaces is shown in Fig. 5.3 (a). The side view of the viscous-potential velocity, the inviscid-potential velocity and the transpiration velocity contour plots are shown in Fig. 5.3 (b)-(d). As can be seen, the transpiration velocity is mainly noticeable at the wake surface. The bottom view of theses three contour plots are shown in Fig. 5.4. The transpiration velocity becomes large at the keel region behind the sonar dome and this allows the viscous-potential velocity to capture the viscous effects due to the vorticity generated by the sonar dome.

The contour plots of the votical velocity magnitude $\mathbf{w}^*$ and vorticity magnitude $\boldsymbol{\omega}^*$ are shown in Fig. 5.5. Slices taken at the calm-water plane and the center plane are shown. As can be seen, the magnitude of the vortical velocity contour correlates well with the contour plot of the vorticity magnitude. This means the viscous potential can correctly describe the velocity field from the NSL solution.

More detailed examination of the viscous-potential velocity is done by comparing

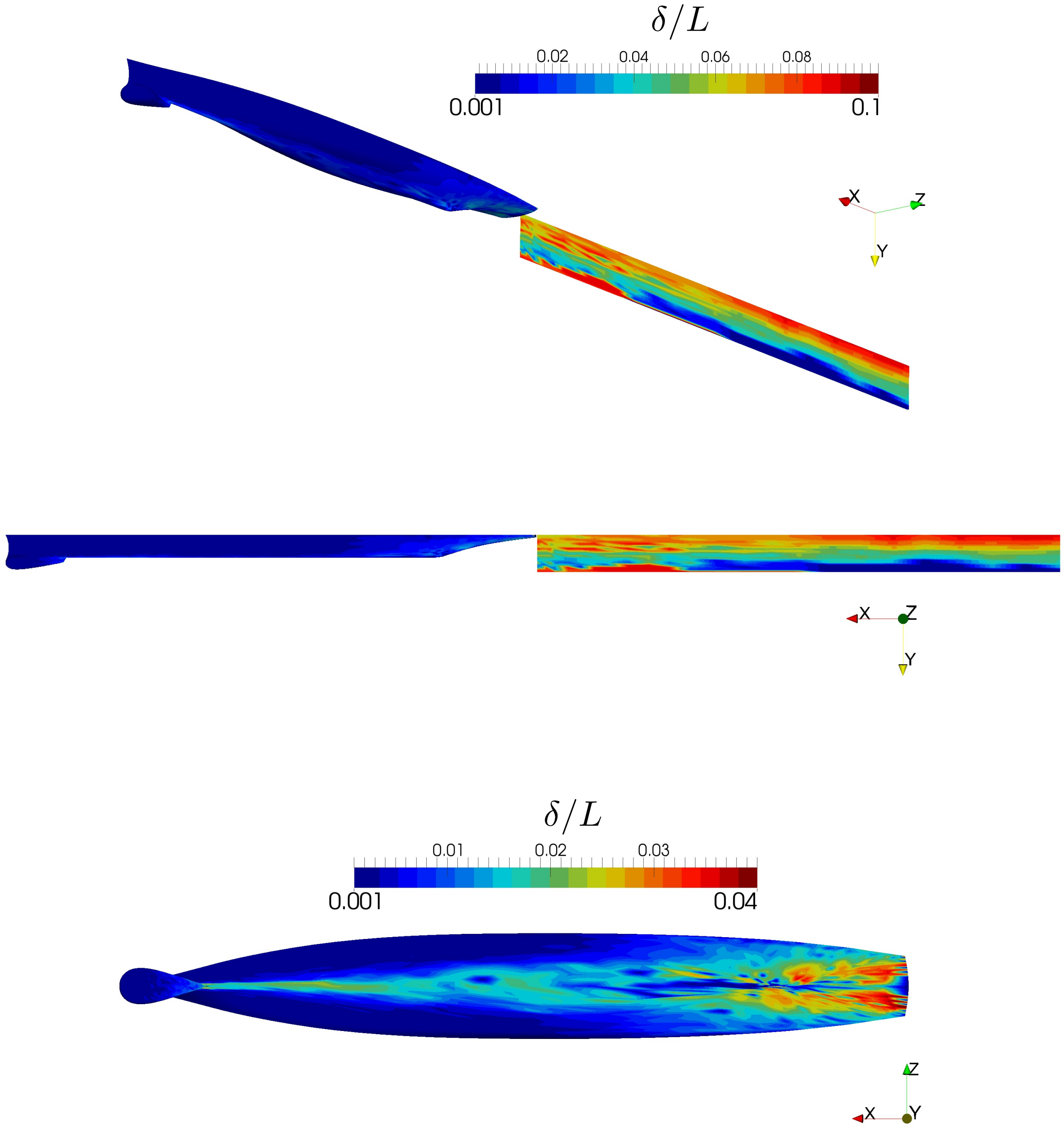
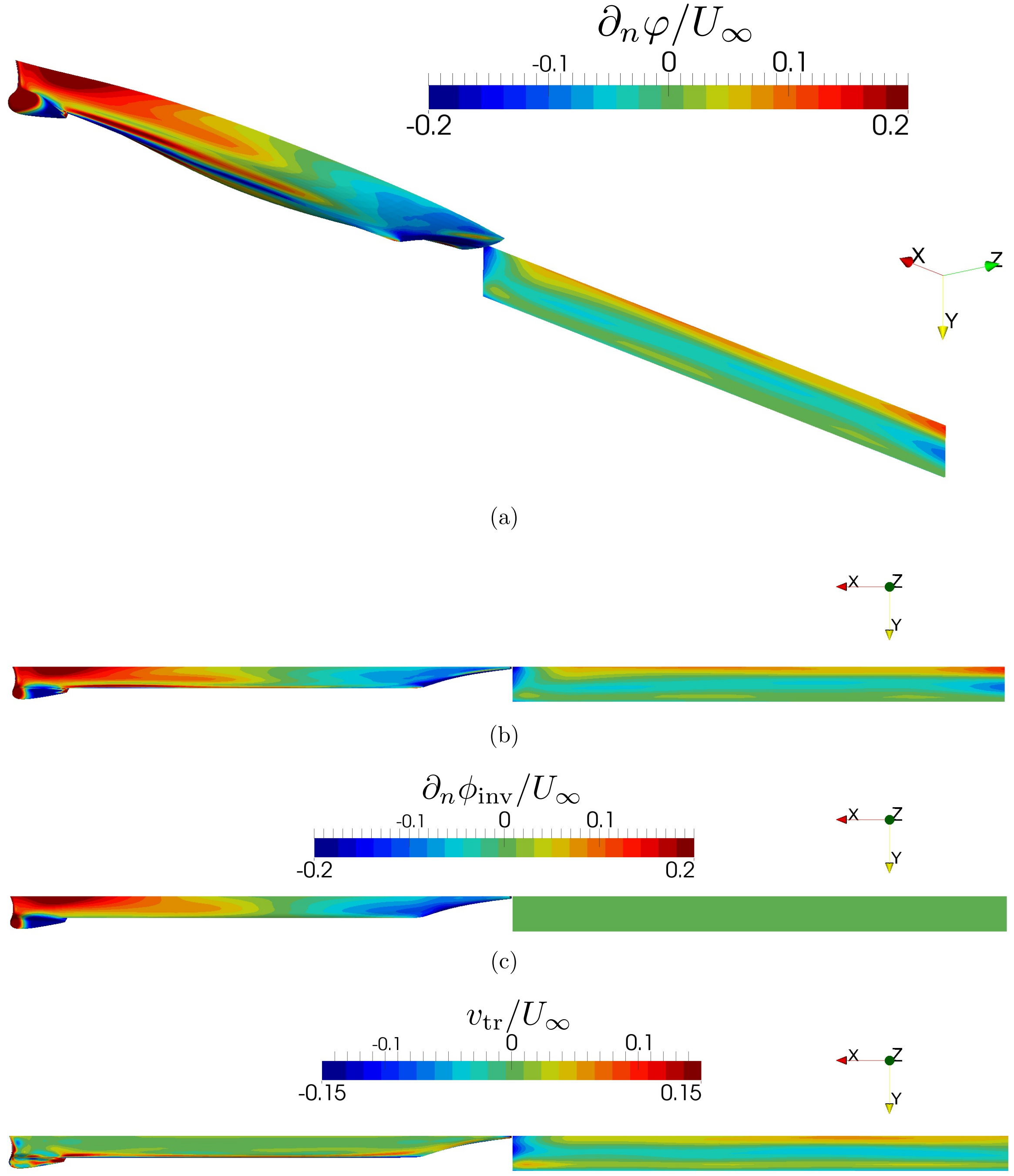


Figure 5.2: Vorticity thickness δ for turbulent flow over a DTMB 5415 ship hull (from top to bottom: perspective, side and bottom view)



(a)

(b)

(c)

(d)

Figure 5.3: (a) Perspective view of the viscous-potential velocity contour; side view of: (b) the viscous-potential velocity ($\partial_n\varphi$) contour (c) the inviscid-potential velocity ($\partial_n\phi_{\text{inv}}$) contour (d) the transpiration velocity (v_{tr}) contour for turbulent flow over a DTMB 5415 ship hull

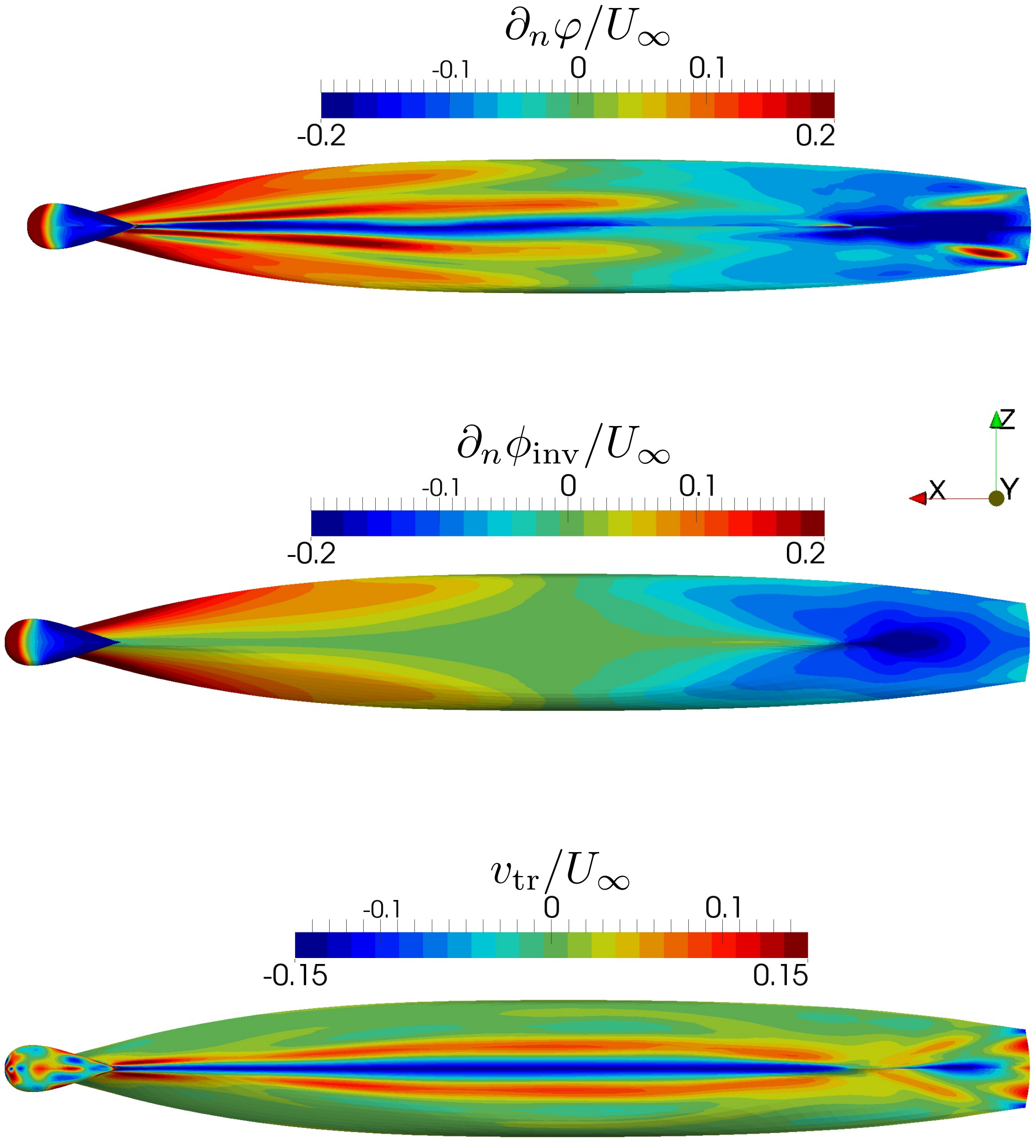


Figure 5.4: Viscous-potential velocity for turbulent flow over a DTMB 5415 ship hull

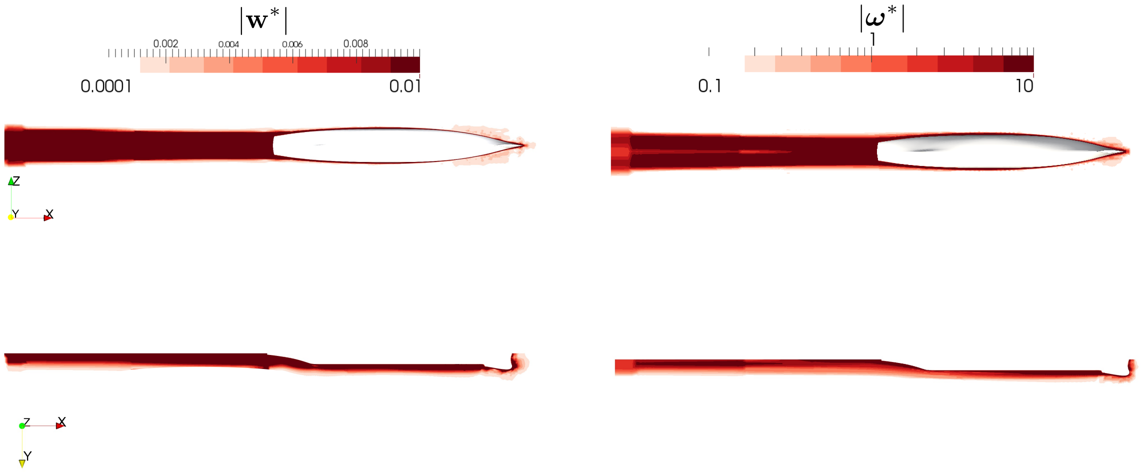


Figure 5.5: Contour plots of the magnitude of the vortical velocities $\mathbf{w}^*$ and vorticity $\boldsymbol{\omega}^*$ for turbulent flow over a DTMB 5415 ship hull (superscript "$*$" denotes non-dimensionalized values)

the velocity profiles. The viscous-potential velocity profile at four different locations are compared with the total velocity from the NSL solution in Fig. 5.6. As can be seen, both the streamwise and vertical components of the viscous-potential velocity agree well with the total velocity.

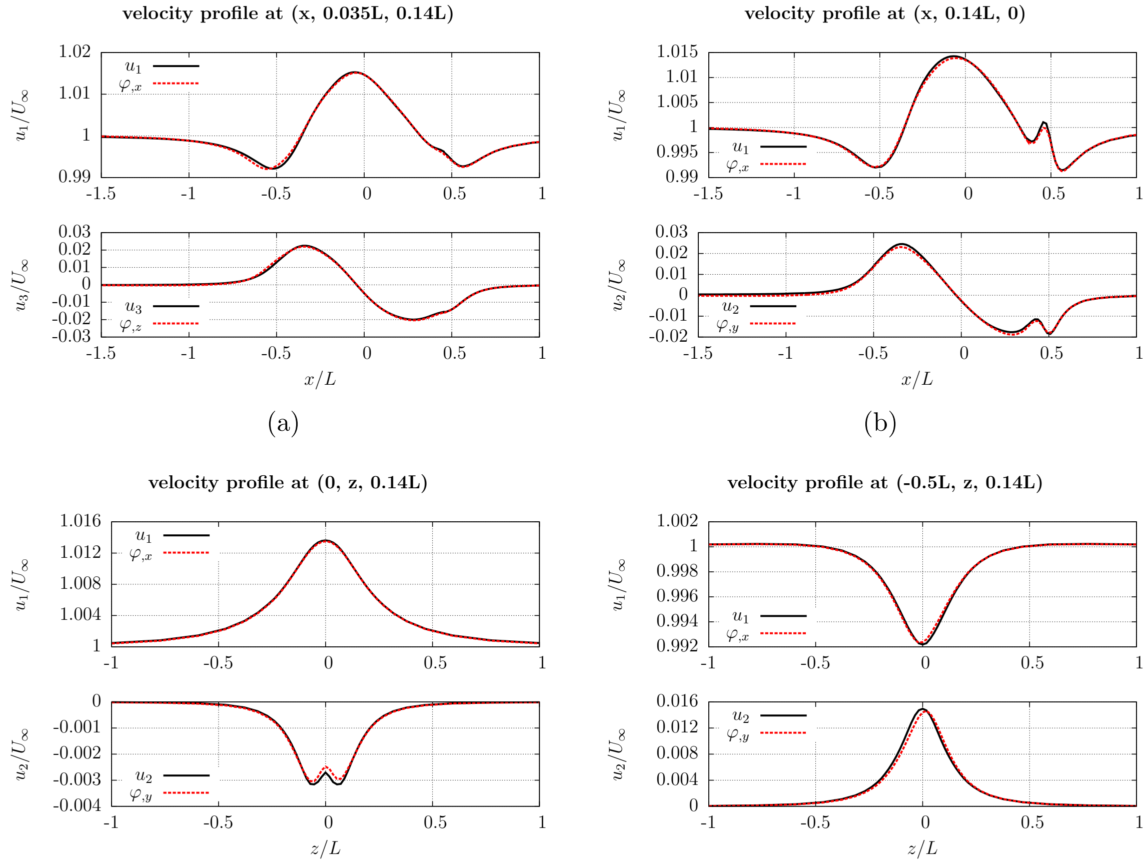
5.1.2 Velocity Decomposition for Transient Flow Simulation

In this section the velocity decomposition solver is applied to solve the same case on a greatly reduced domain. The mesh for the reduced domain is identical to the one of the NSL solutions in their common regions. The details of the mesh for the reduced domain are shown in Table 5.3. The inlet boundary condition is updated with the viscous-potential velocity every 0.1 flow-over ($\Delta t_{\text{update}} = 0.1t^* = 0.1L/U_\infty$).

x_{inlet}	x_{outlet}	# of cells
$0.09L$	$1L$	2,893,037

Table 5.3: Discretization in OpenFOAM for the reduced domain for turbulent flow over a DTMB 5415 ship hull

The time histories of the streamwise and vertical velocity at three different locations, and the drag coefficient, are compared with the NSL solutions in Fig. 5.7. As



(a)

(b)

(c)

(d)

Figure 5.6: Streamwise and vertical velocity profile comparison

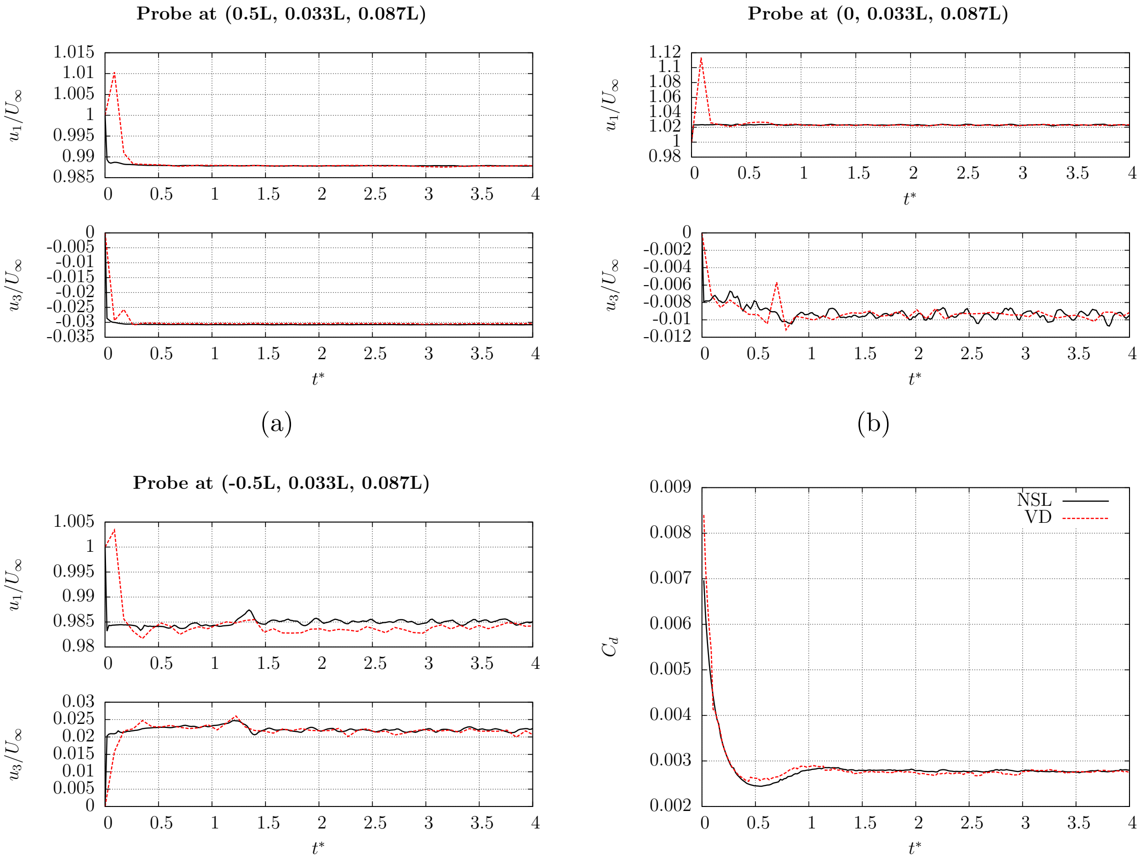
can be seen, the VD solutions agree well the NSL solutions after several updates are applied.

The mean velocity contours of the VD solution at the propeller plane are also compared to the experiment measurement by *Longo et al.* (2007) in Fig. 5.8. As can be seen, the VD solution generally agrees well with the measurement. It should be pointed out that the free-surface is present for the experiment, while only the double-body flow is calculated in this study. So the free-surface effect can cause the difference in the comparison. Moreover, the turbulence model and the relatively coarse mesh used in this study can be the source of disagreement in the comparison.

NSL-total	VD-total	avg(VD-update)
$2,363$	$1,654$	0.047

Table 5.4: The comparison of the CPU hours for turbulent flow over a DTMB 5415 ship hull

The total CPU time for the NSL solution equals roughly 2,363 hours, while the one for the VD solution with 50 updates is about 1,654 hours (Table 5.4). The CPU time is calculated by multiplying the walltime with the number of processors used to compute the solution for 10 flow-overs. The averaged time for each update (solving the viscous potential and applying it to update the inlet boundary condition) is about 0.047 hours. The rest of the run time is used by file I/O, initialization and other area that can be potentially improved. Another way to achieve further computational saving is to apply fewer updates, since after 20 updates the viscous potential has converged to a steady state. Currently, the coefficient matrix for evaluating the velocity at the inlet boundary is re-formulated at the beginning of each update. When the distance between the body and the inlet boundary is not changed, this coefficient matrix can be stored and reused for the following updates and this allows additional computational savings.



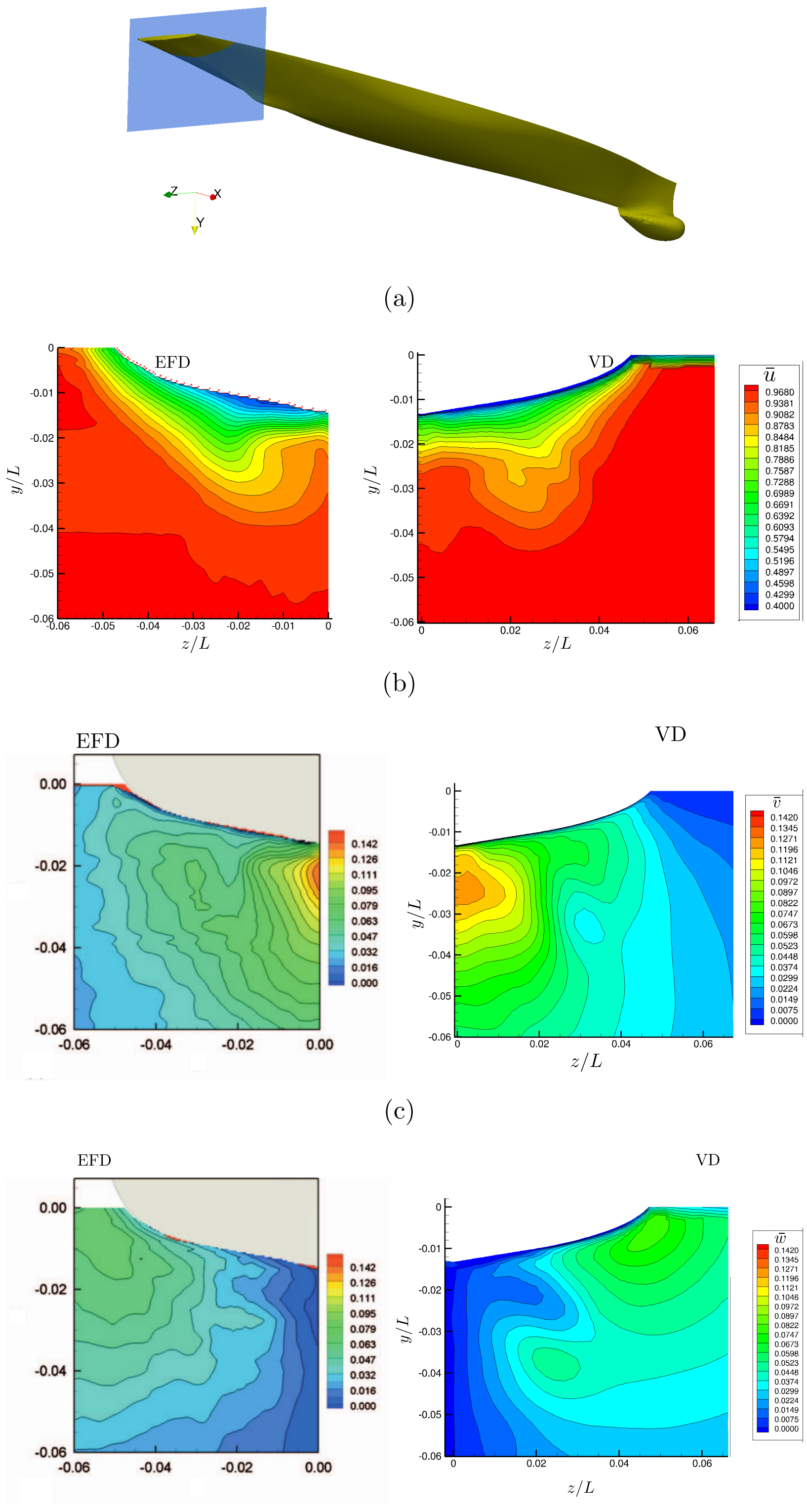
(a)

(b)

(c)

(d)

Figure 5.7: The comparison of streamwise, vertical velocity and drag coefficient time histories for turbulent flow over a DTMB ship hull



(a)

(b)

(c)

(d)

Figure 5.8: Mean velocity profiles at the propeller plane for turbulent flow over a DTMB ship hull

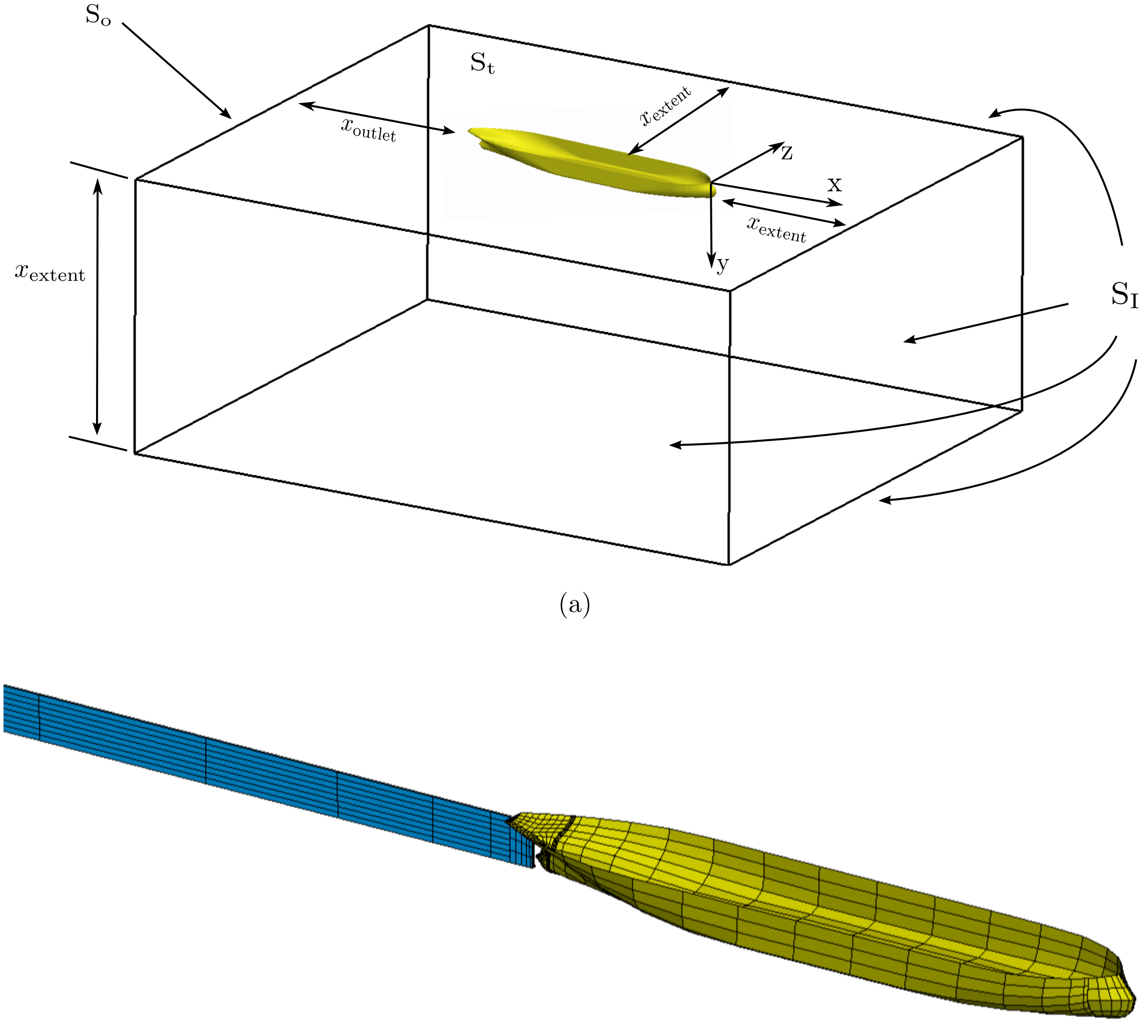
5.2 Turbulent Flow over a KVLCC2 Ship Hull

In this section, impulsively started turbulent flow (Re $= 4.6 \times 10^6$) over a KVLCC2 is studied. The KVLCC2 represents a typical 300,000 tons tanker hull form and has been subject of several experimental and computational studies (*Lee et al.* (2003); *Kim et al.* (2001); *Fureby et al.* (2016); *Larsson et al.* (2010)). The computational domains in OpenFOAM and Aegir are shown in Fig. 5.9. The origin of the coordinate system is located at the center of the ship hull. A symmetry boundary condition is applied on the top boundary S_{t} (no free-surface effect is considered in this study). S_{I} is the inlet boundary and S_{o} is the outlet boundary. In Fig. 5.9(b), the body panels are colored in yellow and the wake panels are in blue. The turbulence is modeled through LES and the localized dynamic one equation eddy viscosity model is used *Kim and Menon* (1995) to account for the unresolved turbulence scales.

The details of the discretization used in OpenFOAM for the NSL solution and in Aegir for the viscous potential are shown in Table 5.5 and Table 4.10. The mesh in OpenFOAM is made such that the average y^+ is approximately 20. A wall function using Spalding's universal law is applied on the body boundary. It is worth noting that the FVM mesh used for this test case is overly coarse to be considered for an accurate LES calculation. However, the focus of this work is to demonstrate the capability of the velocity-decomposition approach to be applied with different turbulence modeling techniques. To conduct a valid LES simulation, at least 85% of the turbulent kinetic energy should be resolved.

x_{inlet}	x_{outlet}	# of cells	y^+_{avg}
5L	5L	2,836,444	17

Table 5.5: Discretization in OpenFOAM for the NSL solution for turbulent flow over a KVLCC2 hull



(a)

(b)

Figure 5.9: Sketch of the computational domain: (a) in OpenFOAM; (b) in Aegir

wake panel extent	body panels #	total # of panels
5L	3081	3186

Table 5.6: Discretization in Aegir for the viscous potential for turbulent flow over a KVLCC2 hull

5.2.1 Viscous Potential

Before applying the velocity decomposition coupled solver, the viscous potential calculated based on the fully developed NSL solutions (after 20 flow-overs) is studied in this section. The perspective, side and bottom views of the contour plot of the vorticity thickness δ are shown in Fig. 5.10. As shown in Fig. 5.10, δ is small over most of the body, except the area close to the stern region. In the wake region, δ takes a much larger value ($\delta \sim B/2$), which is of the order of magnitude of half ship beam B. As can be seen in the bottom view of Fig. 5.11, the viscous-potential velocity introduces influx at the stern region to model viscous effects.

The perspective view of the viscous-potential velocity on the body and the wake surfaces is shown in Fig. 5.11 (a). The side view of the viscous-potential velocity, the inviscid-potential velocity and the transpiration velocity contour plots are shown in Fig. 5.11 (b)-(d). As can be seen, the transpiration velocity is mainly noticeable around the propeller shaft and at the wake surface. The bottom view of theses three contour plots are shown in Fig. 5.12. The transpiration velocity becomes large around the stern region to help the viscous-potential velocity account for the vorticity shed the propeller shaft.

The contour of the vortical velocity magnitude, and the vorticity magnitude are compared in Fig. 5.13. Both the slices at the calm-water plane and the center plane of the ship hull are shown. As can be seen, the contour plots for the vortical-velocity magnitude correlate well with the ones for the vorticity magnitude, which means the viscous-potential velocity matches well with the NSL solution.

For a more detailed comparison, the viscous-potential velocity profile at four differ-

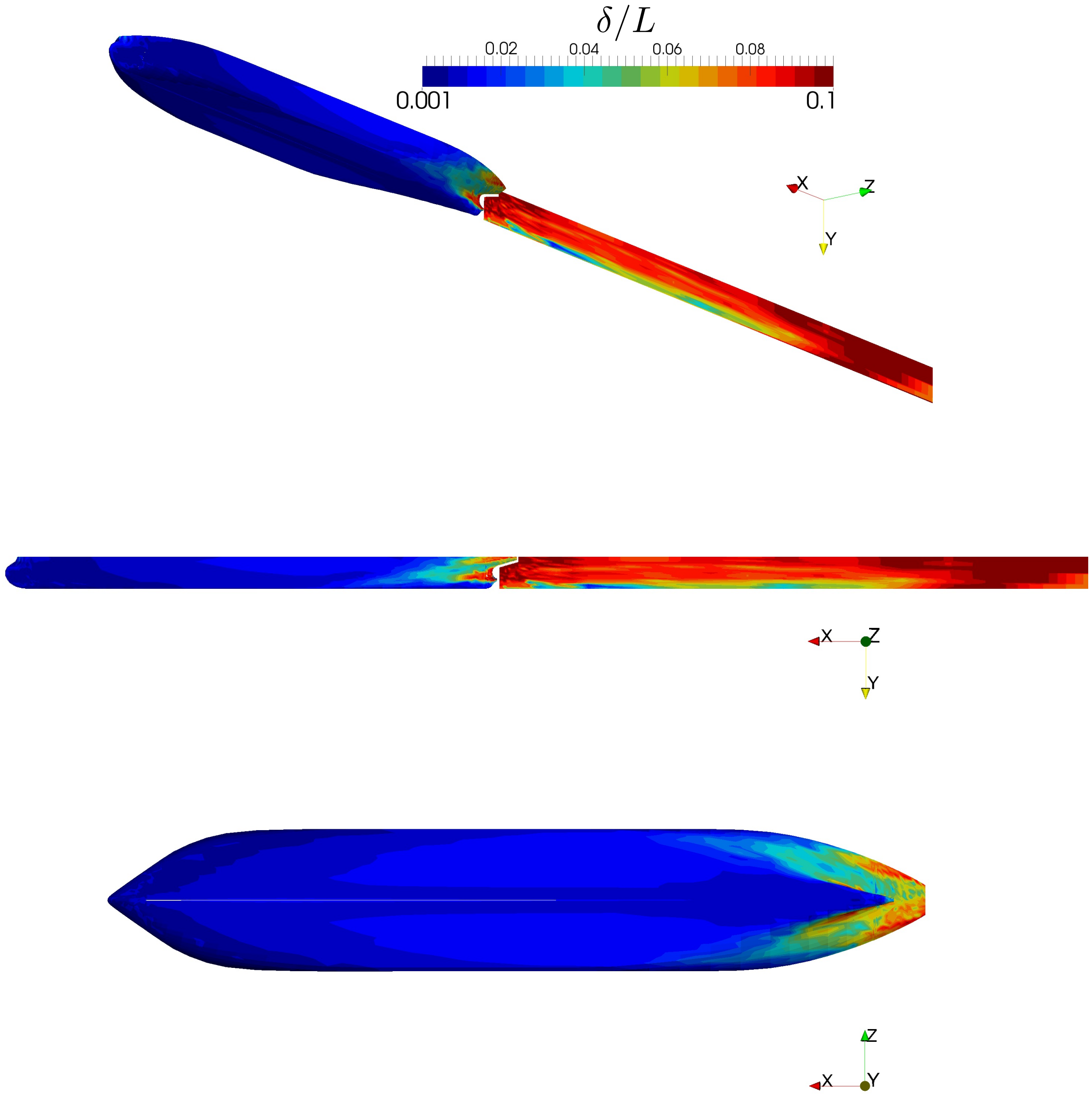
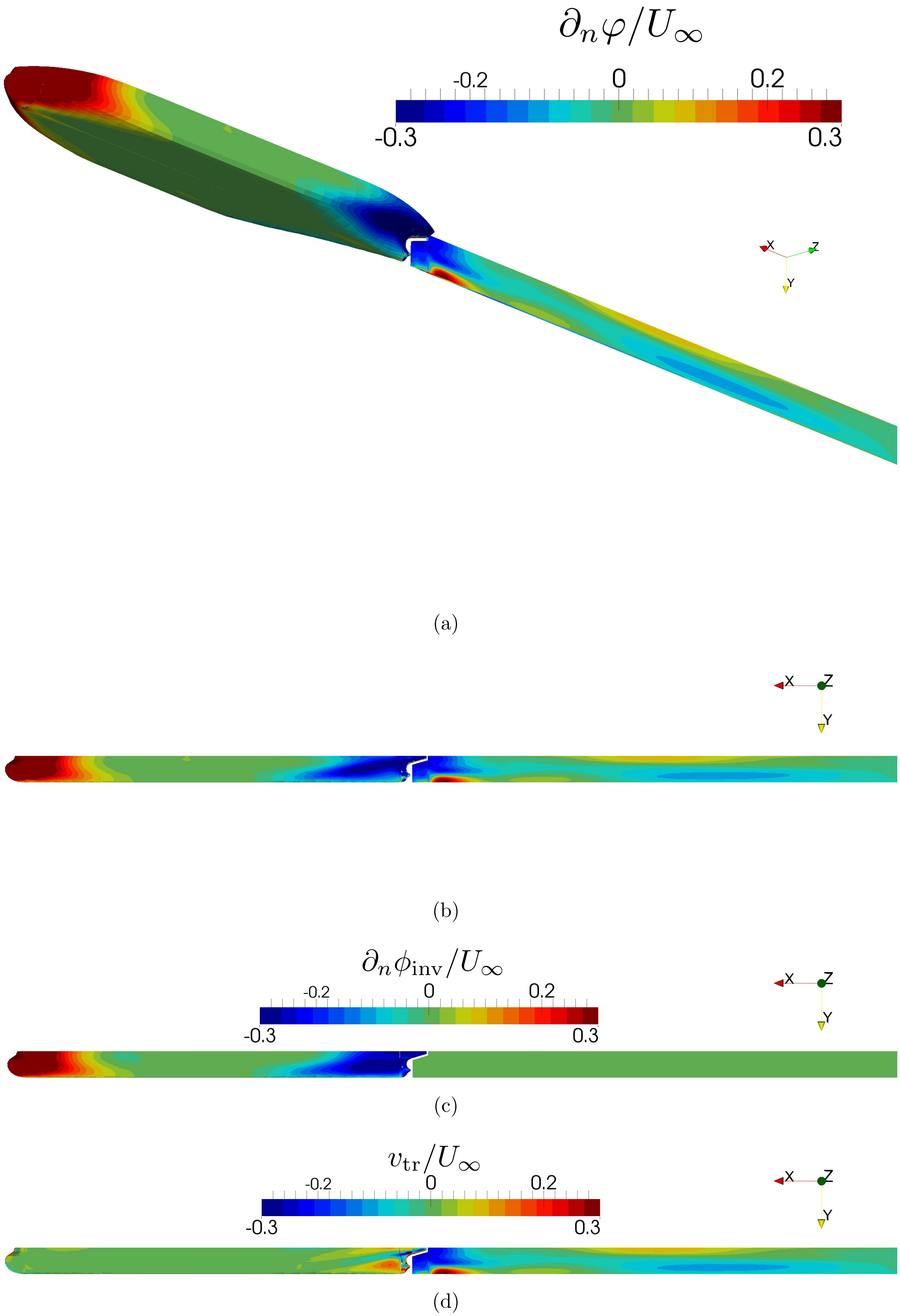


Figure 5.10: Vorticity thickness δ for turbulent flow over a KVLCC2 ship hull (from top to bottom: perspective, side and bottom view)



(a)

(b)

(c)

(d)

Figure 5.11: Viscous-potential velocity for turbulent flow over a KVLCC2 ship hull

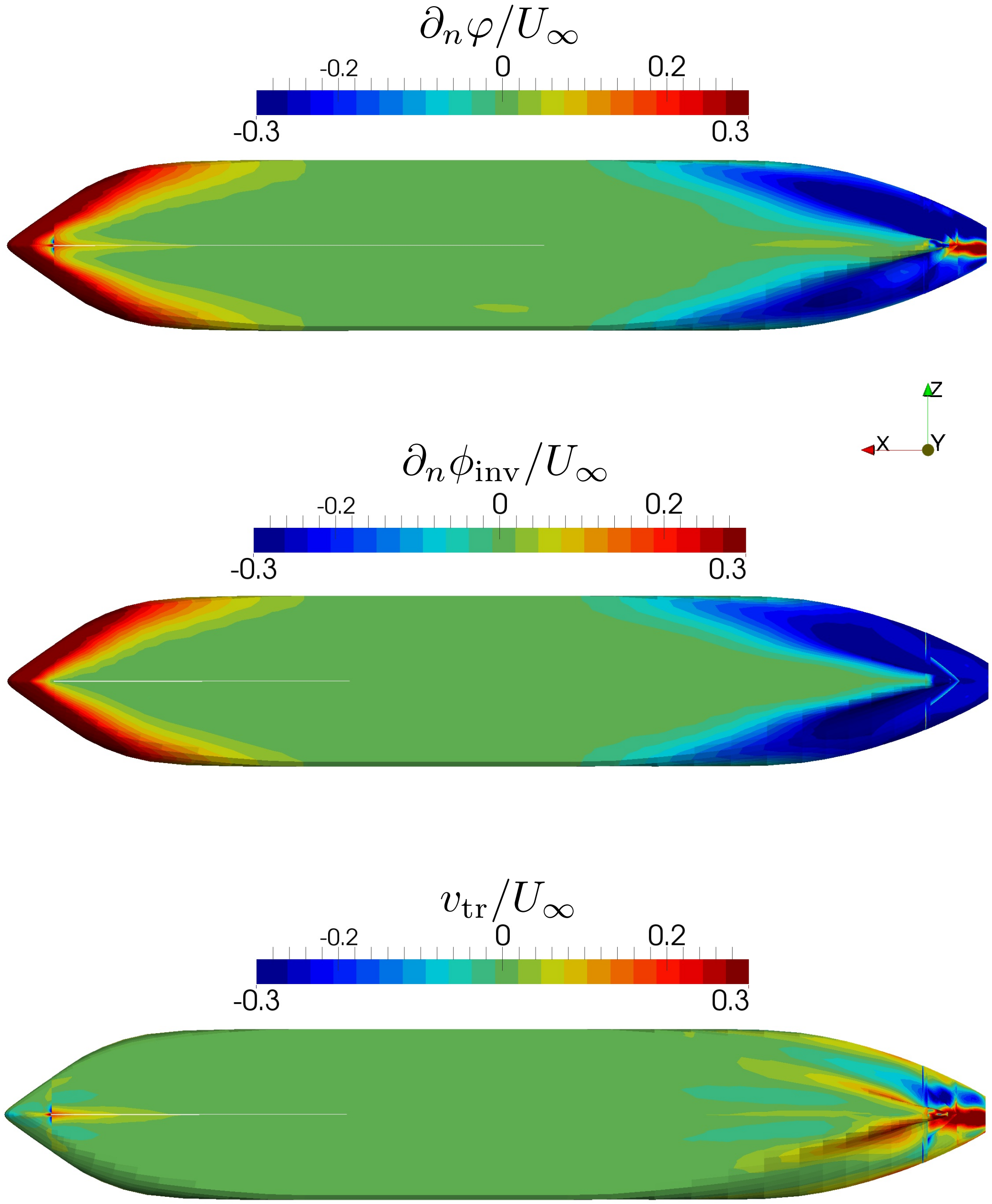


Figure 5.12: Viscous-potential velocity for turbulent flow over a KVLCC2 ship hull

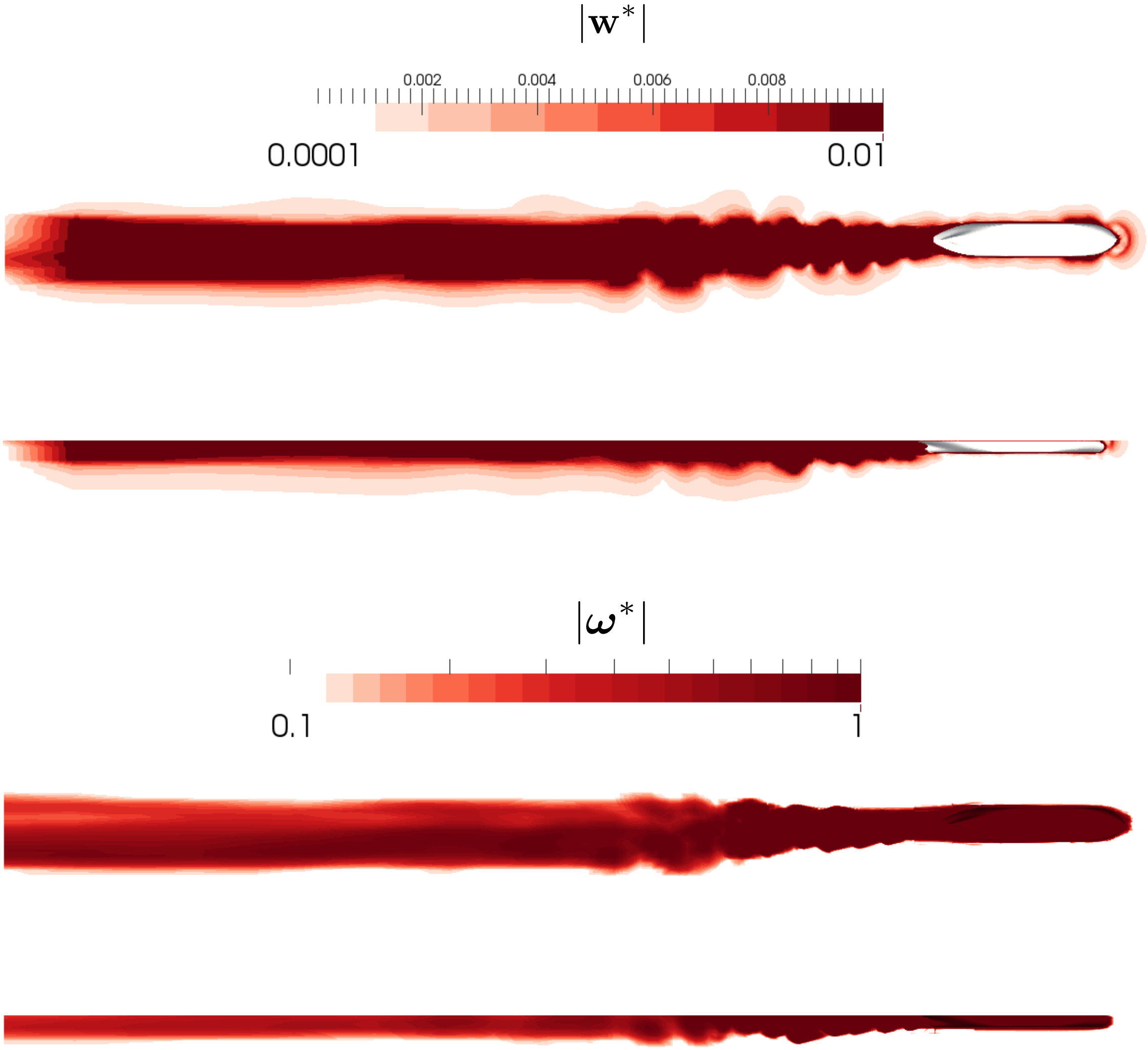
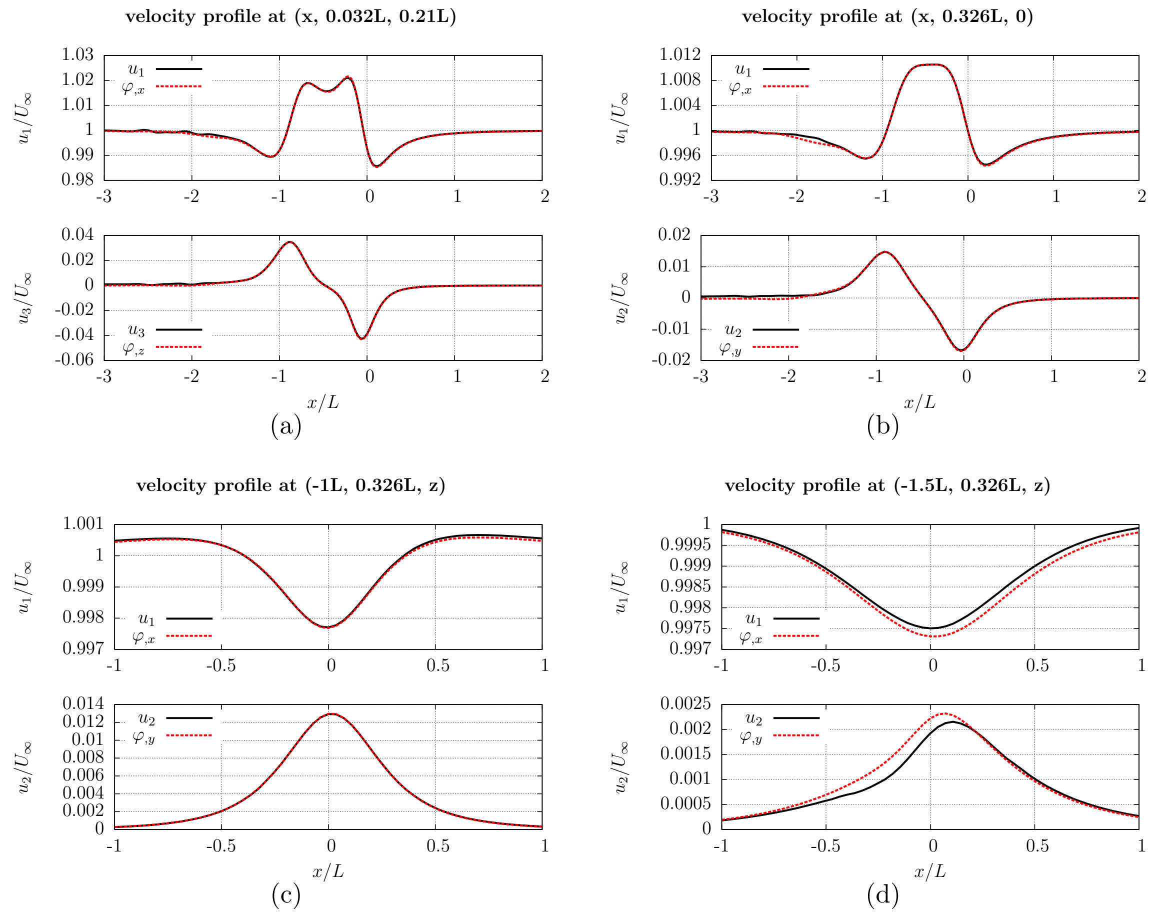


Figure 5.13: Contour plots of the magnitude of the vortical velocities $\mathbf{w}^*$ and vorticity $\boldsymbol{\omega}^*$ (superscript “$*$” denotes non-dimensionalized values)



(a) (b) (c) (d)

Figure 5.14: Streamwise and vertical velocity profile comparison

ent locations are plotted with the total velocity from the NSL solution in Fig. 5.14. As can be seen, both the streamwise and transverse components of the viscous-potential velocity agree well with the total velocity.

5.2.2 Velocity Decomposition for Transient Flow Simulation

In this section the velocity decomposition solver is applied to solve the same case on a greatly reduced domain. The mesh for the reduced domain is identical to the one of the NSL solutions in their common regions. The details of the mesh for the reduced domain are shown in Table 5.7. The inlet boundary condition is updated with the viscous-potential velocity every 0.1 flow-over ($\Delta t_{\text{update}} = 0.1t^* = 0.1L/U_\infty$).

The time histories of the streamwise and vertical velocity at three different loca-

x_{inlet}	x_{outlet}	# of cells
$0.25L$	$5L$	$2,435,542$

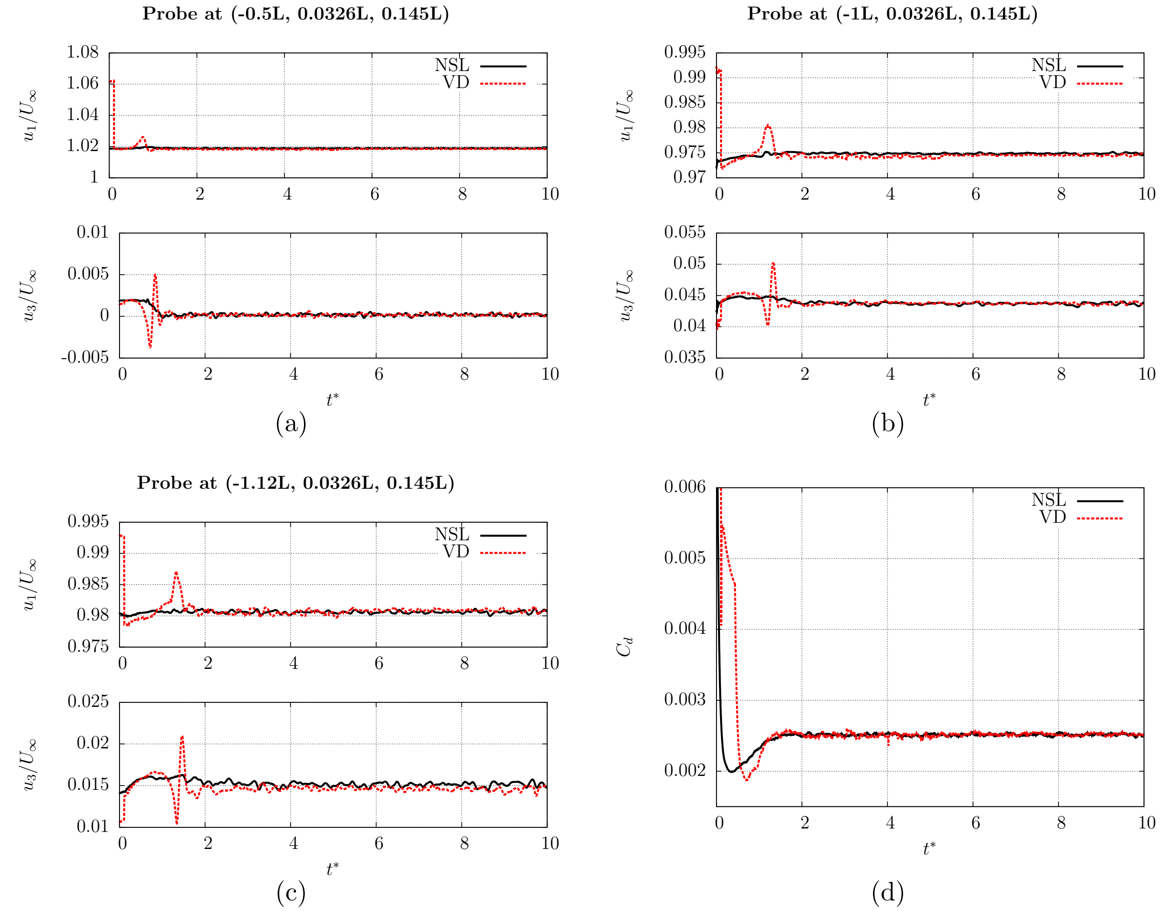
Table 5.7: Discretization in OpenFOAM for the reduced domain for turbulent flow over a KVLCC2 ship hull

tions, and the drag coefficient, are compared with the NSL solutions in Fig. 5.15. The velocity time histories oscillate around $t^* = 1$, due to the perturbations introduced by updating the boundary condition. The oscillations disappear after a few updates are applied. As can be seen, the VD solutions agree well with the NSL solutions after several updates are applied.

NSL-total	VD-total	avg(VD-update)
$1,253$	964	0.034

Table 5.8: The comparison of the CPU hours for turbulent flow over a KVLCC2 ship hull

The total CPU time for the NSL solution equals roughly 1,253 hours, while the one for the VD solution with 50 updates is about 964 hours (Table 5.8). The CPU time is calculated by multiplying the walltime with the number of processors used to compute the solution for 10 flow-overs. The averaged time for each update (solving the viscous potential and applying it to update the inlet boundary condition) is about 0.034 hours. The improvement of the computational efficiency for the VD solver is currently not optimized. The improvement can be potentially improved by reducing the amount of file I/O in the numerical implementation of the coupling. Another way to achieve further computational saving is to apply fewer updates, since after 10 updates the viscous potential has converged to a steady state. Currently, the coefficient matrix for evaluating the velocity at the inlet boundary is re-formulated at the beginning of each update. When the distance between the body and the inlet boundary is not changed, this coefficient matrix can be stored and reused for the following updates and this allows additional computational savings.



(a)

(b)

(c)

(d)

Figure 5.15: The comparison of streamwise, vertical velocity and drag coefficient time histories

CHAPTER VI

Lifting Flow

Flows with lift are important for a wide range of engineering applications. Ships rely lifting surfaces to be propelled forward and to maneuver. When naval vessels executing a turn, the ship hull essentially act as a low-aspect ratio wing generating a drift force. The maneuverability and stability of airplanes also depend on the performance of the lifting surfaces. Racing cars use lifting surface to provide down-force to help drive around corners. So the ability to predict the lift force efficiently and accurately is very important. However, the phenomena of vortex shedding and separation that are strongly associated with lift (*Zhu et al.* (2015)), can create challenges for conventional potential flow solvers that do not account for viscous effects. As discussed in Section 2.4.2, the viscous potential is capable to handle lifting flows, even with separation.

In this section, the method proposed in Section 2.4.2 is tested by conducting calculation for flow over a 6:1 elongated spheroid at a angle of attack of five degrees. Flow over a 6:1 elongated spheroid at angle of attack is a well established case and has been studied experimentally and numerically by many researchers (*Chesnakas and Simpson* (1996), *Wetzel et al.* (1998),*Alin et al.* (2005), *Kim et al.* (2003)).

For this test case, the benchmark solutions (denoted as NSL) are calculated by using a conventional FVM Navier-Stokes solver on a large computational domain.

The viscous-potential velocity is first studied, then the velocity decomposition coupled solver is used to solve for the same case on a greatly reduced domain. The solutions from the coupled solver (denoted as VD) are compared with the benchmark solutions.

As discussed in Section 2.4.2, it is not known a priori how the flow leaves the trailing edge of the body when the body has continuous curvature or the flow is separated. To calculate a correct viscous potential, it is important to capture the viscous effects at the viscous wake. To study the effect of the wake-surface configurations on the solution of the viscous potential, the viscous potential is computed by using different forms of wake surfaces and compared with the NSL solution. The different forms of the wake surfaces are also used with the coupled VD solver and the solutions are compared with the NSL solutions.

6.1 Turbulent Flow over a 6:1 Prolate Spheroid at an Angle of Attack

In this section, turbulent flow (Re $= 4.6 \times 10^6$) over a 6:1 prolate spheroid is studied. Flow over a prolate spheroid at an angle of attack is a canonical flow problem for studying 3-D separation. It is also an important geometry to study for the engineering designs such as submarines and fuselages.

To investigate the effects of the wake-surface configuration on the calculation of the viscous potential, three different wake surfaces, shown in Fig. 6.1, are used in this study. The vortical region are indicated in red in Fig. 6.1. As discussed in Section 3.2.1, the wake surface should be topologically similar to the shape of the vortical wake and it allows the wake surface to better determine the extent of the vortical region and accurately account for the viscous effect at the wake. Hence, all three wake surfaces shown in Fig. 6.1 are selected as cylindrical to better represent the vortical region at the wake.

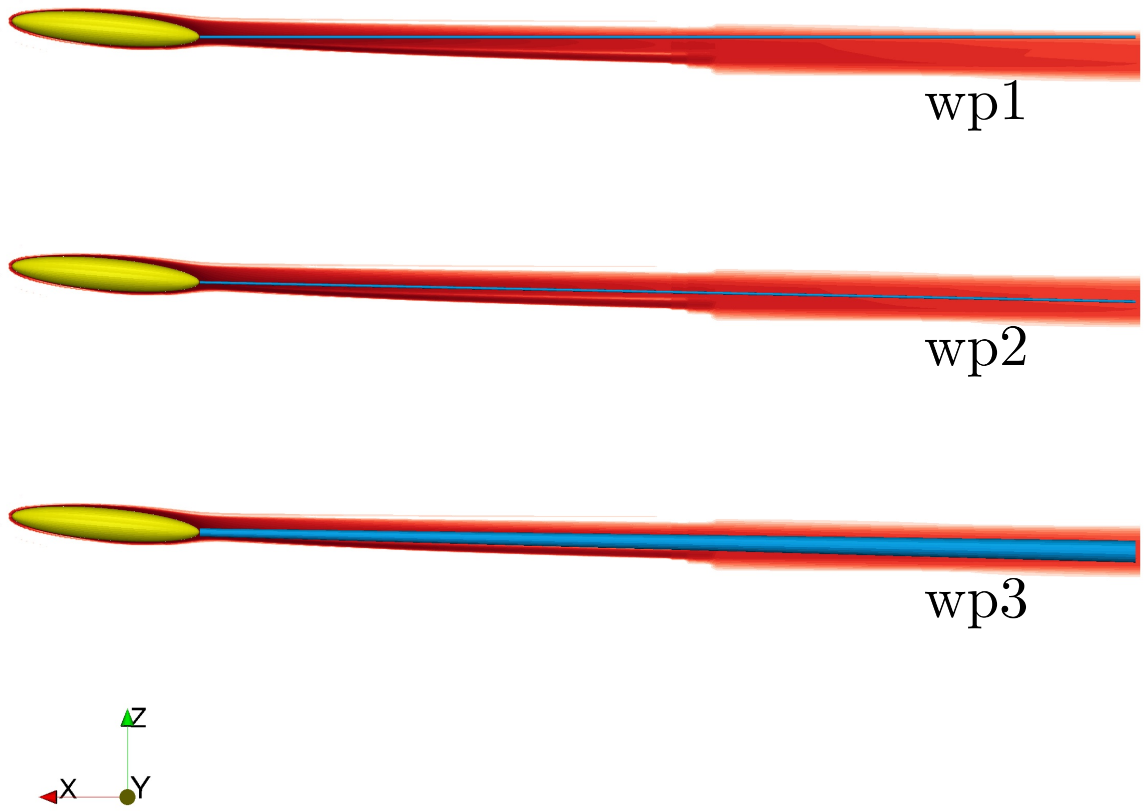


Figure 6.1: Three different wake surface configurations for turbulent flow over a prolate spheroid at $\alpha = 5°$ (body and wake surfaces are colored in yellow and blue respectively, the vortical region is indicated in red)

The specification of the wake surfaces are shown in Table 6.1, where radius 1 represents the radius at the end of the wake surface near the body, radius 2 represents the radius at the other end of the wake surface, which is five body length downstream to the body.

wake surface name	aligned with	radius 1	radius 2
wp1	free-stream	$0.043B$	$0.043B$
wp2	vortical wake	$0.043B$	$0.043B$
wp3	vortical wake	$0.043B$	$0.35B$

Table 6.1: Radius of the wake surfaces for turbulent flow over a prolate spheroid at $\alpha = 5°$ (B is the width of the body)

As shown in Fig. 6.1, wake surface "wp1" is aligned with the free-stream direction, stemming from the end of the spheroid. The wake surface "wp2" is with the same shape of "wp1", except that it is aligned approximately with the center of the vortical region. Comparison between the solution from using "wp1" and "wp2" can demonstrate the effect of the wake surface alignment. The wake surface "wp3" is also aligned approximately with the center of the vortical region, but its radius varies linearly from one end to the other, as shown in Table 6.1. Comparing the solution from using "wp3" and "wp2" should provide some understanding of the effect of the shape of the wake surface.

The computational domains in OpenFOAM is shown in Fig. 6.2(a). The origin of the coordinate system is located at the center of the spheroid. A symmetry boundary condition is applied at the center plane of the body due to the symmetry of the problem. S_{I} is the inlet boundary and S_{o} is the outlet boundary. The computational domain using the wake surface "wp3" in Aegir is shown in Fig. 6.2(b). The body panels are colored in yellow and the wake panels are in blue. The details of the discretization used in OpenFOAM for the NSL solution and in Aegir for the viscous potential are shown in Table 6.2 and Table 6.3. The number of panels on the wake surface is the same for all three wake surfaces ("wp1", "wp2", "wp3"). The mesh in

OpenFOAM is made such that the average y^+ is approximately 37. A wall function is applied on the body boundary. The k- ω SST model described in Section 3.1.1 is used for turbulence modeling.

x_{inlet}	x_{outlet}	# of cells	y^+_{avg}
$5L$	$5L$	$3,063,436$	37

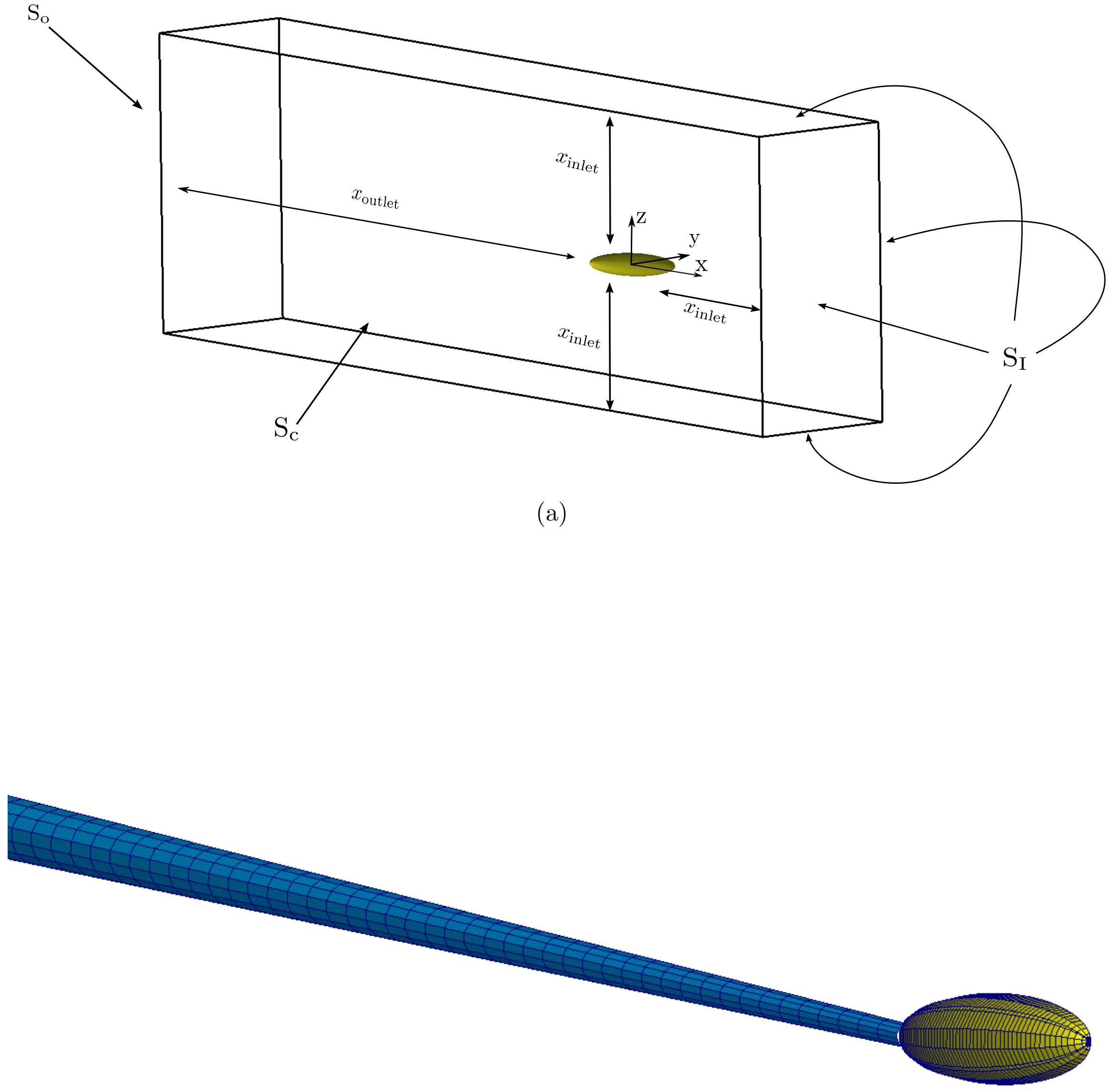
Table 6.2: Discretization in OpenFOAM for the NSL solution for turbulent flow over a prolate spheroid at $\alpha = 5°$

wake panel extent	body panels #	total # of panels
$5L$	152	200

Table 6.3: Discretization in Aegir for the viscous potential for turbulent flow over a prolate spheroid at $\alpha = 5°$

6.1.1 Viscous Potential

The viscous-potential velocity used in this section is calculated based on the fully developed NSL solutions (after 20 flow-overs). The contour of the vortical velocity magnitude and vorticity magnitude are compared in Fig. 6.3. Fig. 6.3 (a) shows the slices of the contours at the x-z symmetry plane, with vorticity magnitude contour plot shown on the left and the contours of the vortical velocity magnitude for each wake surface are shown on the right and indicated with there name (wp1, wp2, wp3). The slices of the contours approximately along the center of the vortical wake at the x-y plane are shown at Fig. 6.3 (b). As can be seen, the contour plots for the vortical-velocity magnitude generally correlate well with the ones for the vorticity magnitude, which means the viscous-potential velocity matches well with the NSL solution. However, the vortical velocity magnitude contour plot with “wp3” wake surface correlates best with the vorticity magnitude contour plot. The vortical-velocity magnitude contour plots between “wp1” and “wp2” are very similar. This seems to indicate that as long as the wake surface is located inside the vortical wake, the alignment of the wake surface is not as important as the shape of the wake surface.



(a)

(b)

Figure 6.2: Sketch of the computational domain: (a) in OpenFOAM; (b) in Aegir

The vorticity thickness δ and the viscous-potential velocity on the body and wake surfaces for the case with "wp3" are plotted in Fig. 6.4 and Fig. 6.5. The viscous-potential velocity is compared with the inviscid-potential velocity on the body in Fig. 6.6. Their difference, the tranpiration velocity, is also shown in Fig. 6.6. As can be seen, the transpiration velocity becomes large close to the trailing edge at the suction side (top) to model the vortical region.

For a more detailed comparison, the viscous-potential velocity profile at four different locations are plotted with the total velocity from the NSL solution in Fig. 6.7 and Fig. 6.8. The locations where the velocity profiles are sampled are indicated with blue lines in Fig. 6.7 (a) and Fig. 6.8 (b). As can be seen, both the streamwise and transverse components of the viscous-potential velocity generally agree well with the total velocity for all three wake configurations. As indicated in Fig. 6.3, the viscous potential by using wake surface "wp3" is more accurate than the ones using wake surfaces "wp1" and "wp2". However, the differences appear in Fig. 6.8(c) and (d) are very small, about 0.05% of the free-stream value.

6.1.2 Velocity Decomposition for Transient Flow Simulation

In this section the velocity decomposition solver is applied to solve the same case on a greatly reduced domain. The mesh for the reduced domain is identical to the one of the NSL solutions in their common regions. The details of the mesh for the reduced domain are shown in Table 6.4. The inlet boundary condition is updated with the viscous-potential velocity every 0.1 flow-over ($\Delta t_{\text{update}} = 0.1t^* = 0.1L/U_\infty$). The update process is started after about two flow-overs ($t_{\text{update}}U_\infty/L = t^*_{\text{update}} \approx 2$).

x_{inlet}	x_{outlet}	# of cells
$0.33L$	$5L$	2,328,262

Table 6.4: Discretization in OpenFOAM for the reduced domain

The time histories of the streamwise and vertical velocity at three different lo-

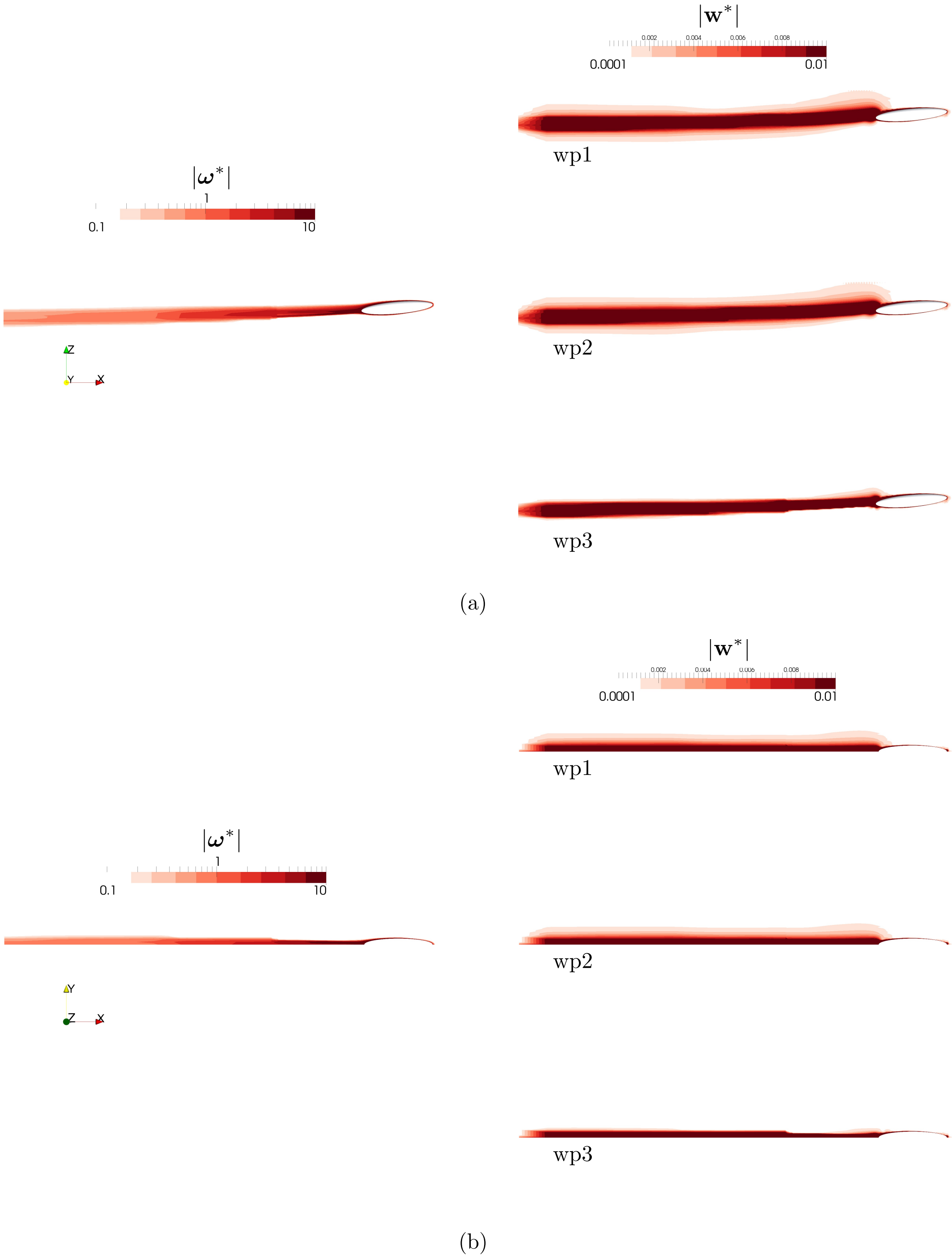
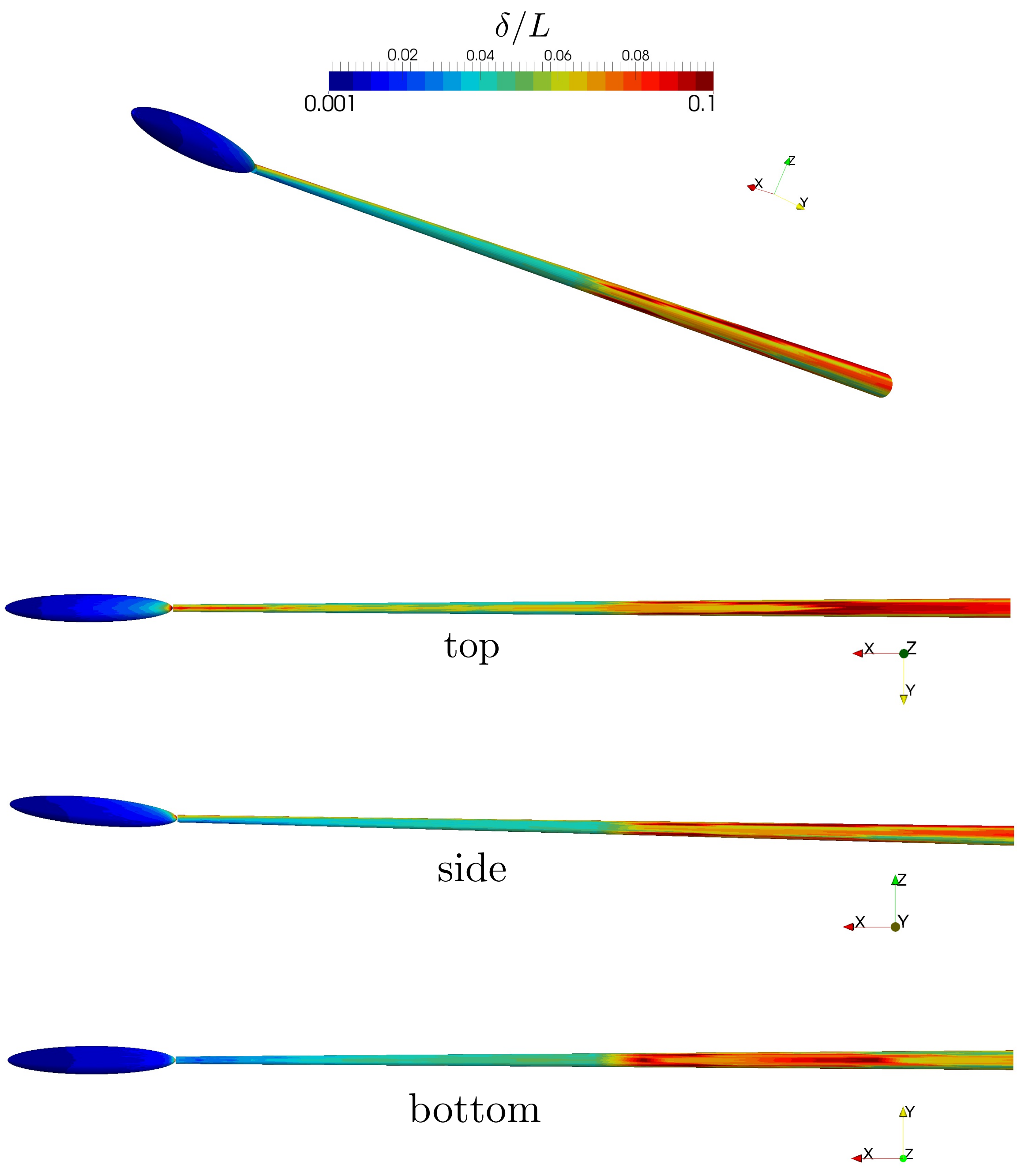


Figure 6.3: Contour plots of the magnitude of the vortical velocities $\mathbf{w}^*$ and vorticity $\boldsymbol{\omega}^*$ (superscript "$*$" denotes non-dimensionalized values)



top

side

bottom

Figure 6.4: Vorticity thickness δ for turbulent flow over a elongated spheroid at $\alpha = 5^\circ$ (using wake surface “wp3”)

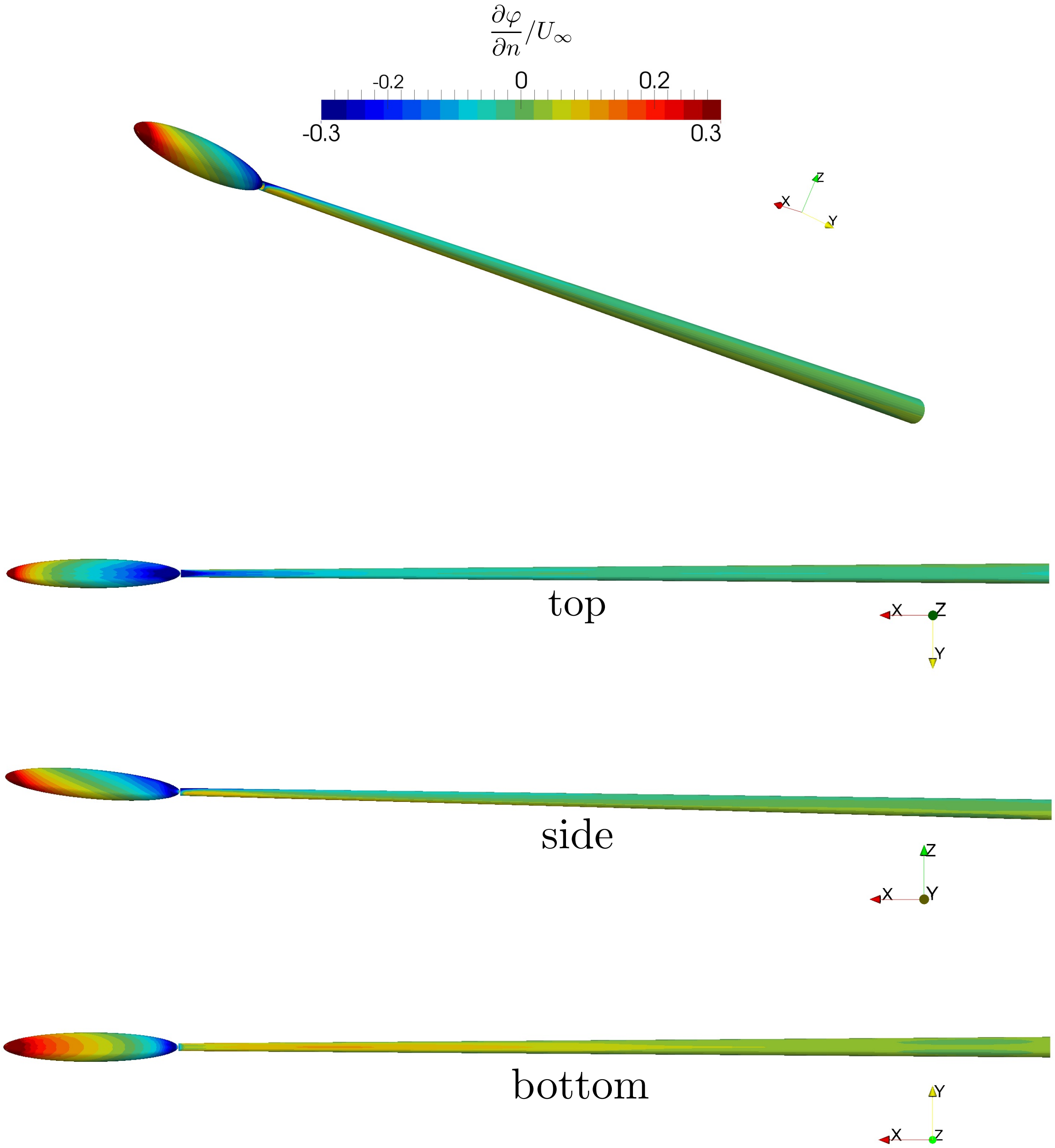


Figure 6.5: Viscous-potential velocity for turbulent flow over a elongated spheroid at $\alpha = 5°$ (using wake surface "wp3")

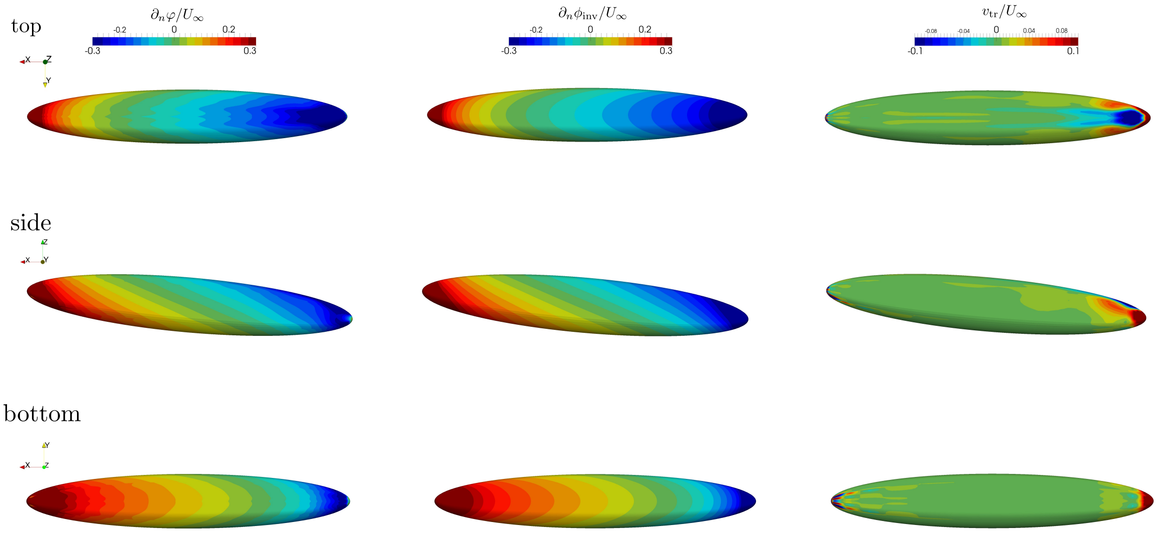
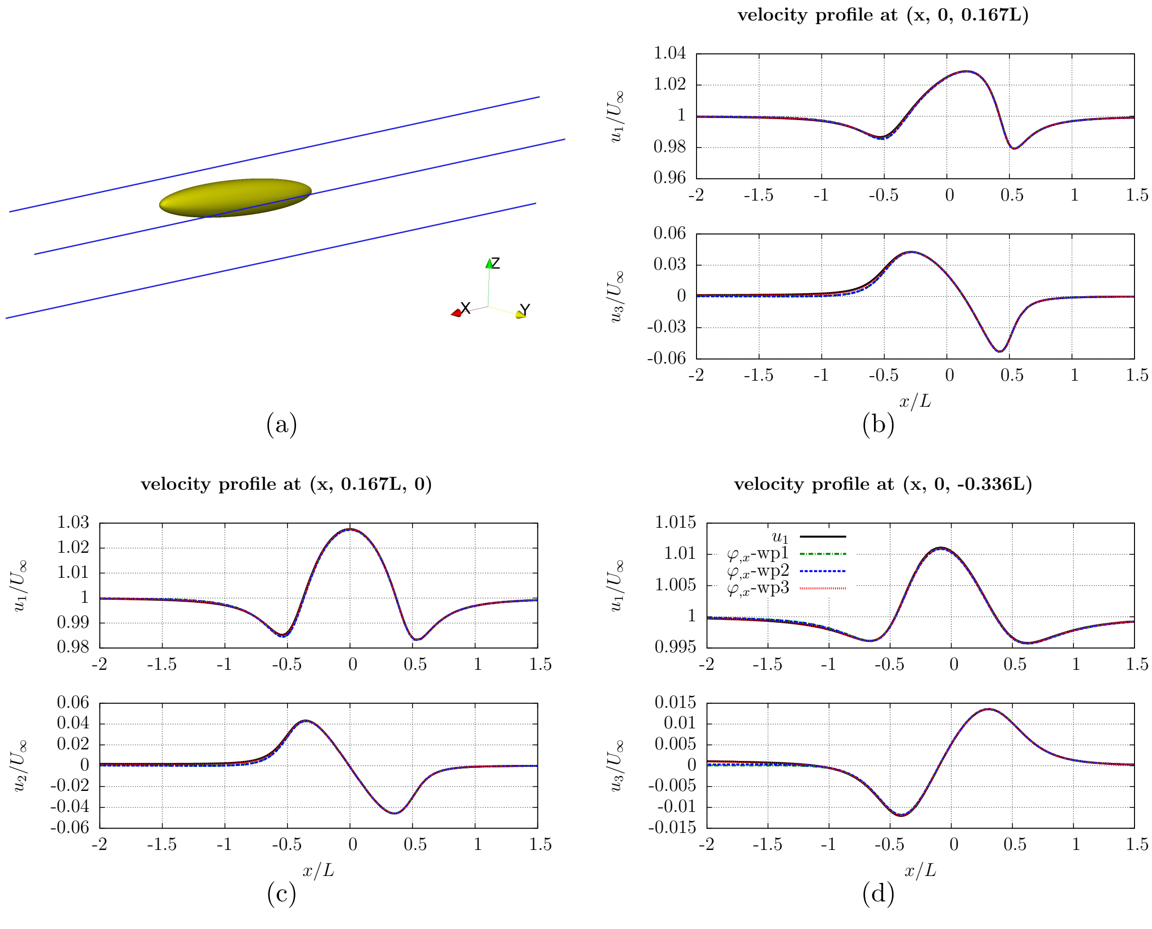

top
$\partial_n\varphi/U_\infty$
-0.2
0
0.2
-0.3
0.3
$\partial_n\phi_{\mathrm{inv}}/U_\infty$
-0.2
0
0.2
-0.3
0.3
v_{tr}/U_∞
-0.08
-0.04
0
0.04
0.08
-0.1
0.1
X
Z
Y
side
bottom

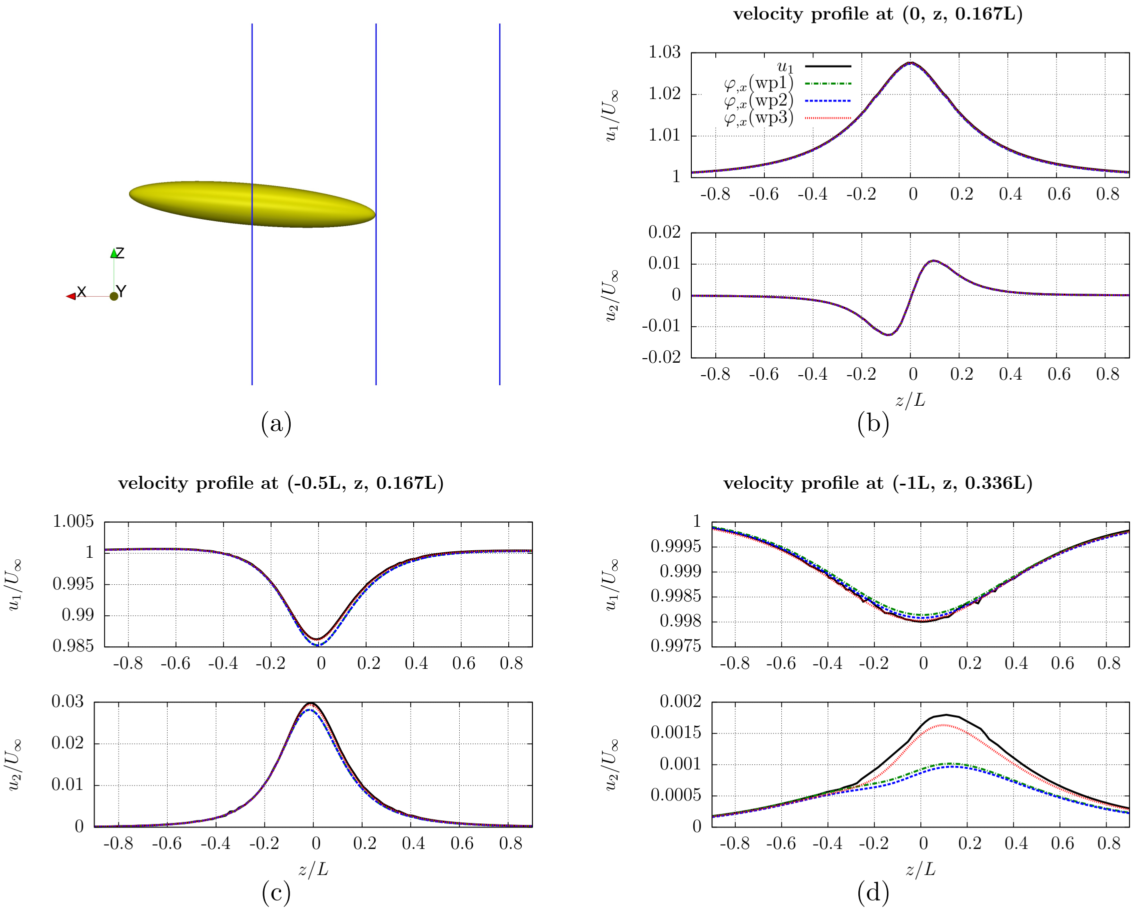
Figure 6.6: Viscous-potential velocity, invisicd potential velocity and transpiration velocity on the body for turbulent flow over a elongated spheroid at $\alpha = 5°$



(a) (b)

(c) (d)

Figure 6.7: Velocity profile comparison between the NSL solution and the viscous-potential velocity



(a)

(b)

(c)

(d)

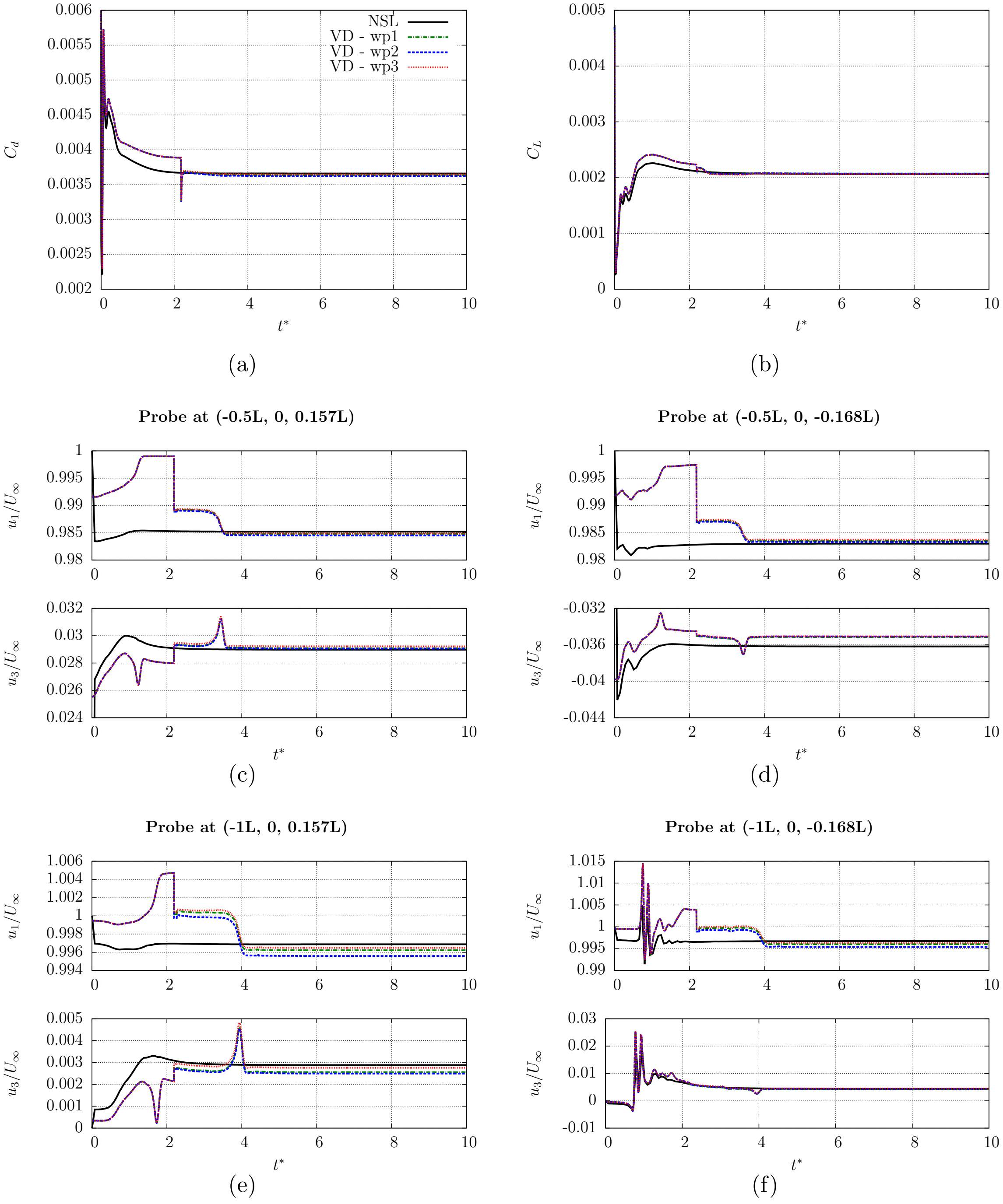
Figure 6.8: Velocity profile comparison between the NSL solution and the viscous-potential velocity

cations, and the drag coefficient, are compared with the NSL solutions in Fig. 6.9. Solutions using all three wake surface shown in Fig. 6.1 are compared. As can be seen, the VD solutions initially do not match with the NSL solution due to the free-stream inlet boundary condition. Then, at around $t^* \approx 2$, the update process started, the VD solutions improve and agree well the NSL solutions after several updates are applied. The VD solutions by using all three wake surface configurations provide very close prediction for lift, drag coefficients and velocities. Small differences in the velocity time histories at the wake half-body lenght downstream, shown in Fig. 6.1 (e) and (f) can be spotted. As indicated in the viscous potential calculation in Section 6.1.1, the solutions by using "wp3" provide slightly better prediction than the other two. In general, the wake surface configuration does not affect the solutions significantly as long as it is placed inside the vortical region and is topologically similar to the vortical wake.

NSL-total	VD-total	avg(VD-update)
691	690	0.055

Table 6.5: The comparison of the CPU hours for turbulent flow over a prolate spheroid at $\alpha = 5°$

The total CPU time for the NSL solution equals roughly 691 hours. The total CPU times for the VD solution with different Wake-surface configurations with 50 updates, are similar and equal about 690 hours (Table 6.5). The CPU time is calculated by multiplying the walltime with the number of processors used to compute the solution for 10 flow-overs. The averaged time for each update (solving the viscous potential and applying it to update the inlet boundary condition) is about 0.055 hours. It should be noted that the improvement of the computational efficiency for the VD solver is currently not optimized. The improvement can be potentially improved by reducing the amount of file I/O in the numerical implementation of the coupling. Another way to achieve further computational saving is to apply fewer updates, since after 20 updates the viscous potential has converged to a steady state. Currently, the coefficient matrix for evaluating the velocity at the inlet boundary is re-formulated at the beginning of each update. When the distance between the body and the inlet boundary is not changed, this coefficient matrix can be stored and reused for the following updates and this allows additional computational savings.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 6.9: The comparison of streamwise, vertical velocity and force coefficients time histories for turbulent flow over a elongated spheroid at $\alpha = 5°$

CHAPTER VII

Summary, Contributions and Future work

7.1 Summary and Contributions

A velocity decomposition approach for 3D, unsteady flows is developed. The decomposition algorithm is able to calculate a 3D, time accurate viscous-potential velocity that fully describes the flow field outside the vortical region. By coupling with the high-order BEM (Boundary-Element-Method) solver, Aegir, the algorithm is able to solve for a accurate viscous-potential, even for cases with complicated geometries (*i.e.* DTMB 5415 and KVLCC2 ship hulls) and with state-of-art turbulence modeling techniques (*i.e.* URANS, LES).

The velocity decomposition for 3D, unsteady flows are formulated and the decomposition is shown to be unique. The viscous boundary condition applying on the wake surface for solving the viscous potential is shown to be equivalent to its body surface counterpart. The feasibility to apply the viscous potential on lifting flows are discussed and the proposed methodology is tested on a test case with a elongated spheroid at angle of attack of five degrees.

The stability of the iterative viscous boundary condition used for the solution of the viscous potential is studied and related to the physics of the flow. It is found that the convergence of the iterative viscous boundary condition is directly related to the vorticity thickness, which is highly dependent to the shape of the body and the Re of the flow field. For the our targeted applications, the bodies are of streamline shapes to mitigate the resistance and the Re is at the range of 10^6 to 10^9. So the iteration should converged rapidly and this is consistent with what is found for all the test cases. This convergence analysis should bring some confident to the application of the velocity decomposition algorithm.

The 3D viscous potential is solved with the higher-order BEM solver, Aegir, instead of the 2D panel method used in previous studies. The discretizations of the BEM and FVM (Finite-Volume-Method) solvers are allowed to be different. This allows the users to have a better control of the resolution, since for FVM, especially high-fidelity simulations (*e.g.* LES) the mesh are required to be dense while BEM requires much coarser discretization. Various important elements for extending the velocity decomposition algorithm for 3D flow simulations with realistic geometries and unstructured meshes are discussed. One of the important elements for accurate solutions for the viscous potential is the determination of the vorticity thickness δ. The various vorticity thresholds used to obtain δ are analyzed based on boundary layer theory and asymptotic solutions. Improvements have been made to the sampling algorithm used to determine δ, so that it can be robustly applied on cases with unstructured meshes and complicated geometries.

The unsteady velocity algorithm is formulated. The time step sizes of the Navier-Stokes and viscous-potential sub-problems are allowed to be different. While it is necessary to use a small time-step size to resolve the high frequency variation due to turbulence in the Navier-Stokes sub-problem, the viscous-potential requires a much large time-step size.

The 3D, unsteady velocity algorithm is tested on a series of test cases. For each test case, the viscous potential is studied and is shown to be able to accurately describe the flow field outside of the vortical region. The difference between the inviscid potential and viscous potential is demonstrated on all test cases. The iterative viscous boundary condition is demonstrated to converge mostly after 3 iterations on the canonical problems (*i.e.* laminar and turbulent flow over a finite flat plate, turbulent flow over a Wigley hull). By utilizing the viscous potential to update the boundary condition, the VD (Velocity-Decomposition) solver is able to solve the same case in a greatly reduced domain, while using the inviscid potential only provides limited improvement and the solutions by using free-stream boundary conditions diverge from the correct solutions. The velocity decomposition algorithm is also applied with LES to solve turbulent flow over a DTMB 5415 and a KVLCC2 ship hull. The velocity decomposition algorithm is applied with LES to solve turbulent flow over a DTMB 5415 and a KVLCC2 ship hull.

Preliminary results of applying the velocity decomposition algorithm for lifting flow with a elongated spheroid at an angle of attack of five degrees are presented. Three different wake surface configurations have been studied on the effect of alignment direction and shape. The velocity decomposition algorithm is shown to be able to solve lifting flow with a reduced domain and the solution matches well with the one using a conventional FVM solver with a large domain. By comparing the velocity time histories and force coefficients, it is demonstrated that the wake surface configurations should not affect the solutions significantly as long as it is place inside the vortical region and has a shape that is topologically similar with the vortical wake.

The improvement of computational efficiency when using the VD solver for 3D, single phase flow is not significant. The mesh reduction can be around $50\% - 15\%$ of the large domain, due to a large portion of the cells are concentrated around the body and wake. However, the ultimate application of the VD solver is for marine free-surface ship flows. For those cases, much finer resolution is required away from the body to capture the free-surface wave accurately. Hence, the mesh reduction by using the VD solver for this type of application can be expected to be significant.

7.2 Future work

The overarching goal of the velocity decomposition approach is to solve for 3-D, unsteady, free-surface ship flows. The apparent next step is to extend this method toward free-surface flows, which is currently underway. A possible first step of extending the algorithm for the free-surface problem can be applying the velocity decomposition algorithm with some linear formations of the free-surface boundary conditions, such as the the inviscid potential of the Neumann-Kelvin problem or the one described in *Woolliscroft and Maki* (2016).

To further improved the computational saving gained by using the VD-coupled solver with a greatly reduced domain, an improved interface to communicate between the BEM and FVM solver needs to be developed to avoid unnecessary file I/O.

In this research, only cases with fixed bodies are considered. Extension toward cases with bodies under forced motion or eventually free-running bodies would be another exciting research topic. An algorithm to dynamically change the wake surface according to the vortical region is important for applications with moving bodies.

The application of the velocity decomposition on lifting flows explored in this work is only an initial step. Further study to problems with a larger angle of attack and massive separations should be conducted. A means to track an unsteady wake is necessary for realistic applications.

Simulations involving multiple bodies, such as vessel to vessel interaction, can be a beneficial from the velocity decomposition approach, since the disturbance caused by the bodies only changes the farfield boundary condition which can be accounted by the viscous potential. The N-S solver can then solve for the flow around each body within its own domain.

APPENDICES

APPENDIX A

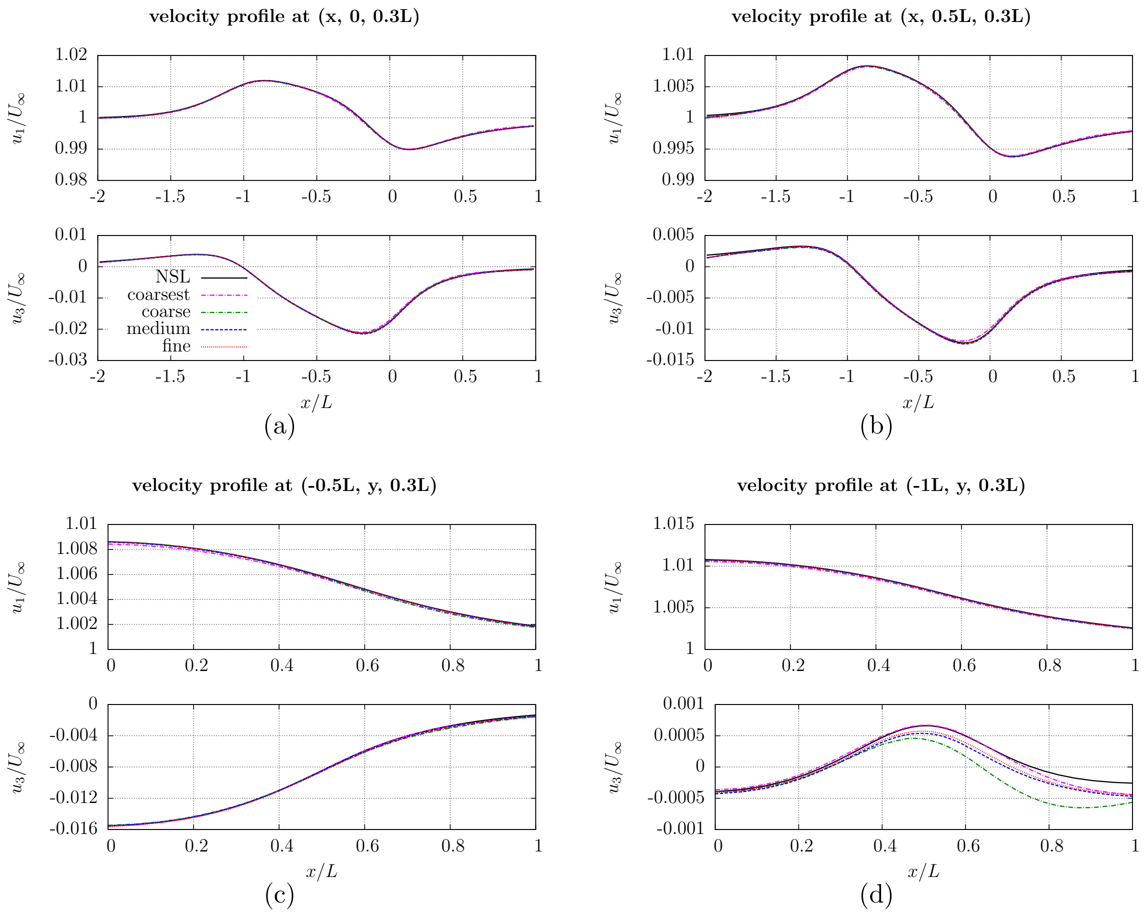
Grid Convergence Study for the BEM Solution

The grid convergence study for the solution of the viscous potential for the laminar flow over a finite flat plate is presented in this appendix. The details of the different sets of the discretization in the BEM solver (Aegir) is shown in Table A.1. The discretization of the viscous potential in BEM is highly related to both the vorticity thickness δ and the transpiration velocity v_{tr}, shown in Fig. 4.2. Hence, cosine spacing are used to enable a better resolution at the leading, trailing and side edge of the flat plate.

The viscous potential is calculated based on a fully developed NSL solution and 20 iterations are used for the iterative viscous boundary condition to minimize the iteration error. The velocity profiles calculated using the grids shown in Table A.1. are compared with the NSL solution. Comparison for both the streamwise and vertical component of the velocity profile are shown. As can bee seen, all viscous potential

grid	total # of panels	# of body panels	# of wake panels	# of side panels
coarsest	48	4	4	40
coarse	77	6	6	65
medium	160	8	8	144
fine	360	18	18	324

Table A.1: Discretization in Aegir for laminar flow over a finite flat plate



(a)

(b)

(c)

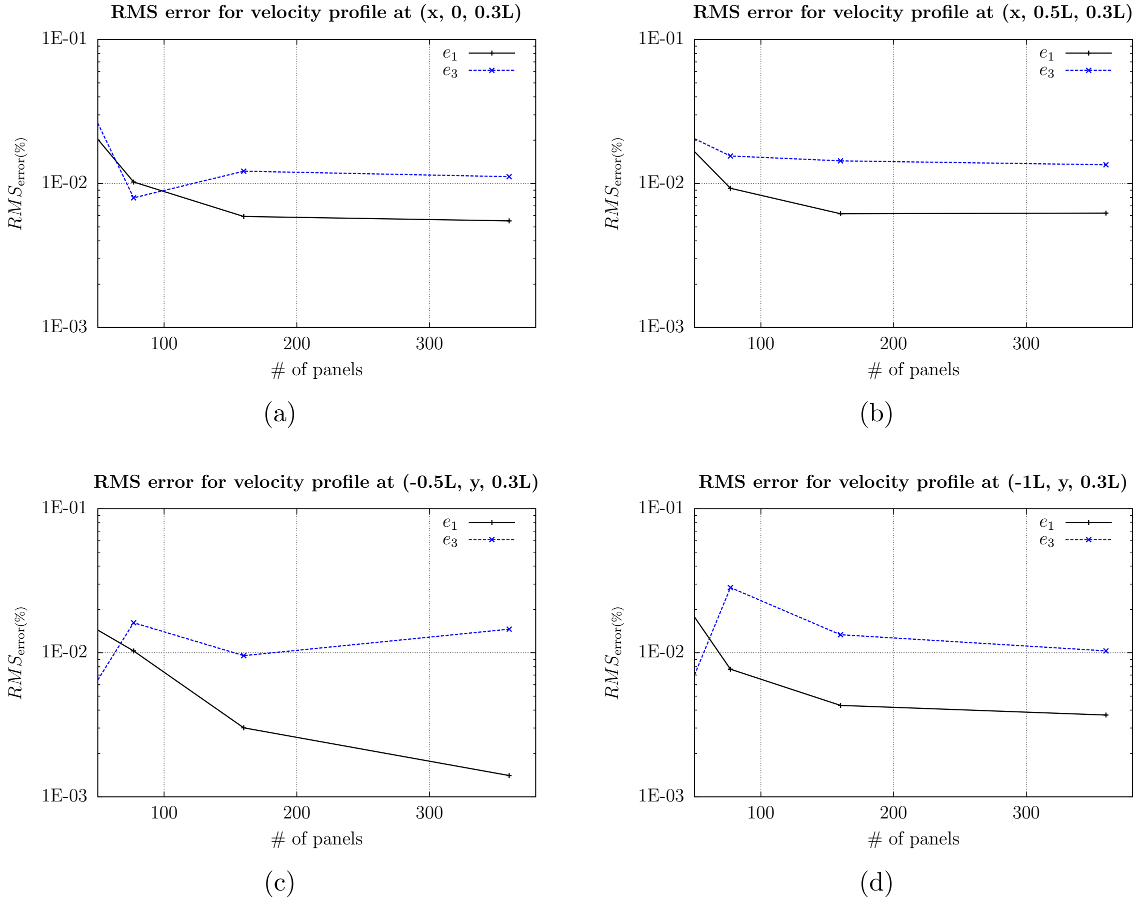
(d)

Figure A.1: Streamwise and vertical velocity profile comparison for different discretization for laminar flow over a finite flat plate

velocity profiles agrees well with the NSL solution. Small differences (0.05% of U_∞) for the vertical velocity in Fig. A.1(d) can be seen, but as the grid refined, the differences decrease.

The RMS error for the velocity profiles in Fig. A.1 is shown in Fig. A.2. The RMS error for the streamwise velocity is calculated using Eq. (4.1), while the one for the vertical velocity is calculated using Eq. (A.1), where the free-stream velocity is used for normalization instead.

$$RMS_{\text{error}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{u_i - u_{\text{NSL,i}}}{U_\infty}\right)^2} \times 100\% \tag{A.1}$$



(a) (b) (c) (d)

Figure A.2: RMS error of velocity profiles as a function of the number of panels

As can be seen in Fig. A.2, all the RMS error for the streamwise velocity profiles (e_1) decrease as the grid is refined. The RMS error for the vertical velocity profiles in Fig. A.2(c) shows oscillatory behavior as the grid refined. However, the RMS error is already very small, around 0.01% and it can be affected by other sources of error, such as the linear interpolation for the calculating the value of δ or the error in the NSL solution.

APPENDIX B

Grid Convergence Study for FVM Solution

The uncertainty due to FVM discretization is estimated following the procedure presented in *Celik et al.* (2008). The drag coefficient for the case of laminar flow over a finite flat plate is used for the analysis with a large domain ($x_{\text{inlet}} = 50L$).

Three sets of systematically refined meshes are used for this study. The details of the meshes are shown in Table B.1. The representative mesh size h is defined by

$$h = \left[\frac{1}{N} \sum_{i=1}^{N} (\Delta V_i) \right]^{1/3} \tag{B.1}$$

where ΔV_i is the volume of the ith cell, and N is the total number of cells.

The observed order of accuracy, p, is calculated by solving Eq. (B.2)-B.4. using fixed-point iteration,

$$p = \frac{1}{\ln(r_{21})} \left| \ln \left| \epsilon_{21} / \epsilon_{21} \right| + q(p) \right| \tag{B.2}$$

grid	total # of cells	h	$r_j = h_j / h_{\text{fine}}$
fine	2,855,160	0.358	1.00
medium	795,156	0.549	1.53
coarse	360,294	0.714	1.99

Table B.1: Mesh details for laminar flow over a finite flat plate uncertainty analysis

r_{21}	1.531
r_{32}	1.302
Cd_1	0.03377
Cd_2	0.03398
Cd_3	0.03423
p	1.415
Cd_{ext}	0.03352
e^{21}_{rel}	0.6164%
e^{21}_{ext}	0.7506%
GCI^{21}_{fine}	0.9313%

Table B.2: Discretization uncertainty values for the FVM solver based on the results of laminar flow over a finite flat plate

$$q(p) = \ln(\frac{r^p_{21} - s}{r^p_{32} - s}) \tag{B.3}$$

$$s = 1 \cdot \text{sgn}(\epsilon_{32}/\epsilon_{21}) \tag{B.4}$$

where $\epsilon_{32} = \phi_3 - \phi_2, \epsilon_{21} = \phi_2 - \phi_1$ and ϕ_k denotes the drag coefficient on kth grid, $r_{21} = h_2/h_1, r_{32} = h_3/h_2$, subscripts $1, 2, 3$ represent the values from the fine, medium and coarse mesh respectively.

The extrapolated drag coefficient can be calculated from

$$\phi^{21}_{\text{ext}} = \frac{(r^p_{21}\phi_1 - \phi_2)}{(r^p_{21} - 1)}. \tag{B.5}$$

Then, the relative error e^{21}_{rel}, extrapolated relative error e^{21}_{ext} and the fine-grid convergence index GCI^{21}_{fine} can be calculated from

$$e^{21}_{\text{rel}} = |\frac{\phi_1 - \phi_2}{\phi_1}| \tag{B.6}$$

$$e^{21}_{\text{ext}} = |\frac{\phi^{21}_{\text{ext}} - \phi_2}{\phi^{21}_{\text{ext}}}| \tag{B.7}$$

$$GCI^{21}_{\text{fine}} = \frac{1.25 e^{21}_{\text{rel}}}{r^p_{21} - 1} \tag{B.8}$$

The previous mentioned variables are calculated and reported in Table B.2. The find grid convergence index GCI^{21}_{fine} represents the numerical uncertainty in the solution of the fine grid.

BIBLIOGRAPHY

BIBLIOGRAPHY

Alin, N., C. Fureby, S. Svennberg, W. Sandberg, R. Ramamuti, R. Bensow, and T. Persson (2005), 3D unsteady computations for submarine-like bodies, in *43rd AIAA Aerospace Sciences Meeting and Exhibit*, pp. 10–13.

Bassanini, P., C. M. Casciola, M. Lancia, and R. Piva (1999), Edge singularities and Kutta condition in 3D aerodynamics, *Meccanica*, *34*(3), 199–229.

Batchelor, G. K. (2000), *An introduction to fluid dynamics*, Cambridge University Press.

Beck, R. F. (1971), The wave resistance of a thin ship with a rotational wake, *Journal of Ship Research*, *15*(3).

Campana, E., A. Di Mascio, P. Esposito, and F. Lalli (1995), Viscous-inviscid coupling in free surface ship flows, *International Journal for Numerical Methods in Fluids*, *21*(9), 699–722.

Celik, I. B., U. Ghia, P. J. Roache, et al. (2008), Procedure for estimation and reporting of uncertainty due to discretization in CFD applications, *Journal of fluids Engineering-Transactions of the ASME*, *130*(7).

Chen, H.-C., and S.-K. Lee (1996), Interactive RANS/Laplace method for nonlinear free surface flows, *Journal of Engineering Mechanics*, *122*(2), 153–162.

Chen, H.-C., and S.-K. Lee (1999), RANS/Laplace calculations of nonlinear waves induced by surface-piercing bodies, *Journal of Engineering Mechanics*, *125*(11), 1231–1242.

Chen, Y. (2014), A velocity decomposition approach for unsteady flow, Master thesis.

Chen, Y., and K. J. Maki (2016), A velocity decomposition approach for three-dimensional unsteady flow, *European Journal of Mechanics-B/Fluids*.

Chen, Y., K. J. Maki, and W. J. Rosemurgy (2015), A velocity decomposition approach for unsteady external flow, in *ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering*, p. V011T12A029, American Society of Mechanical Engineers.

Chesnakas, C., and R. Simpson (1996), Measurements of the turbulence structure in the vicinity of a 3-d separation, *Journal of Fluids Engineering*, *118*(2), 268–275.

Colicchio, G., C. Lugni, M. Greco, and O. M. Faltinsen (2015), Dynamic Domain Decomposition Strategy Coupling Lattice Boltzmann Methods With Finite Differences Approximations of the Navier-Stokes Equations to Study Bodies in Current, in *ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering*, p. V011T12A030, American Society of Mechanical Engineers.

Connell, B. S., W. M. Milewski, B. Goldman, and D. C. Kring (2011), Single and multi-body surface effect ship simulation for T-Craft design evaluation, in *Proceedings of 11th International Conference on Fast Sea Transportation*.

Dias, F., A. Dyachenko, and V. Zakharov (2008), Theory of weakly damped free-surface flows: A new formulation based on potential flow solutions, *Physics Letters A*, *372*, 1297–1302.

Doering, C. R., and J. D. Gibbon (1995), *Applied analysis of the Navier-Stokes equations*, vol. 12, Cambridge University Press.

Drela, M., and M. B. Giles (1987), Viscous-inviscid analysis of transonic and low reynolds number airfoils, *AIAA Journal*, *25*(10), 1347–1355.

Eça, L., and M. Hoekstra (2009), On the numerical accuracy of the prediction of resistance coefficients in ship stern flow calculations, *Journal of Marine Science and Technology*, *14*(1), 2–18.

Edmund, D. O. (2012a), A velocity decomposition method for efficient numerical computation of steady external flows, Ph.D. thesis, The University of Michigan.

Edmund, D. O. (2012b), A velocity decomposition method for unsteady flows demonstrated on flow over a circular cylinder at Re = 140, (Unpublished report).

Edmund, D. O., K. J. Maki, and R. F. Beck (2013), A velocity-decomposition formulation for the incompressible Navier–Stokes equations, *Computational Mechanics*, *52*(3), 669–680.

Ferrant, P., L. Gentaz, B. Alessandrini, R. Luquet, C. Monroy, G. Ducrozet, E. Jacquin, and A. Drouet (2008), Fully nonlinear potential/ranse simulation of wave interaction with ships and marine structures, in *ASME 2008 27th International Conference on Offshore Mechanics and Arctic Engineering*, pp. 379–387, American Society of Mechanical Engineers.

Filip, G., and K. Maki (2015), Evaluation of Advanced Turbulence Models for High-Reynolds Number External Flow.

Filip, G. P. (2013), High-resolution numerical simulation of turbulent interfacial marine flows, Ph.D. thesis, University of Michigan.

Fureby, C., S. Toxopeus, M. Johansson, M. Tormalm, and K. Petterson (2016), A computational study of the flow around the KVLCC2 model hull at straight ahead conditions and at drift, *Ocean Engineering*, *118*, 1–16.

Germano, M., U. Piomelli, P. Moin, and W. H. Cabot (1991), A dynamic subgrid-scale eddy viscosity model, *Physics of Fluids A: Fluid Dynamics (1989-1993)*, *3*(7), 1760–1765.

Guillerm, P.-E., and B. Alessandrini (2003), 3D free-surface flow computation using a RANSE/Fourier–Kochin coupling, *International Journal for Numerical Methods in Fluids*, *43*(3), 301–318.

Hafez, M., A. Shatalov, and E. Wahba (2006), Numerical simulations of incompressible aerodynamic flows using viscous/inviscid interaction procedures, *Computer Methods in Applied Mechanics and Engineering*, *195*(23), 3110–3127.

Hafez, M., A. Shatalov, and M. Nakajima (2007), Improved numerical simulations of incompressible flows based on viscous/inviscid interaction procedures, *Computers & Fluids*, *36*(10), 1588–1591.

Hamilton, J. A., and R. W. Yeung (2011), Viscous and inviscid matching of three-dimensional free-surface flows utilizing shell functions, *Journal of Engineering Mathematics*, *70*(1-3), 43–66.

Hess, J. L. (1974), The problem of three-dimensional lifting potential flow and its solution by means of surface singularity distribution, *Computer Methods in Applied Mechanics and Engineering*, *4*(3), 283–319.

Iafrati, A., and E. Campana (2003), A domain decomposition approach to compute wave breaking (wave-breaking flows), *International Journal for Numerical Methods in Fluids*, *41*(4), 419–445.

Jasak, H. (1996), Error analysis and estimation for finite volume method with applications to fluid flow, Ph.D. thesis, Imperial College of Science, Technology and Medicine.

Joncquez, S., P. Andersen, and H. Bingham (2012), A comparison of methods for computing the added resistance, *Journal of Ship Research*, *56*(2), 106–119.

Joseph, D. D. (2006), Potential flow of viscous fluids: Historical notes, *International Journal of Multiphase Flow*, *32*(3), 285–310.

Katz, J., and A. Plotkin (2001), *Low-Speed Aerodynamics*, vol. 13, Cambridge University Press.

Kendon, T., S. Sherwin, and J. Graham (2003), An irrotational/vortical split-flow approach to viscous free surface flow, in *Proceedings of the 2nd M.I.T. Conference on Computational Fluid and Solid Mechanics, Cambridge, USA*.

Kim, K., A. I. Sirviente, and R. F. Beck (2005), The complementary RANS equations for the simulation of viscous flows, *International Journal for Numerical Methods in Fluids*, *48*(2), 199–229.

Kim, S.-E., S. Rhee, and D. Cokljat (2003), Application of modern turbulence models to vortical flow around a 6: 1 prolate spheroid at incidence, in *AIAA 41st Aerospace Sciences Meeting and Exhibit, Reno, Paper No. AIAA-2003-0429.*

Kim, W., S. Van, and D. Kim (2001), Measurement of flows around modern commercial ship models, *Experiments in Fluids*, *31*(5), 567–578.

Kim, W.-W., and S. Menon (1995), A new dynamic one-equation subgrid-scale model for large eddy simulations, in *AIAA, Aerospace Sciences Meeting and Exhibit, 33 rd, Reno, NV.*

Kring, D., F. T. Korsmeyer, J. Singer, D. Danmeier, and J. K. White (1999), Accelerated nonlinear wave simulations for large structures, in *7th Int'l Conference on Numerical Ship Hydrodynamics, Nantes , France.*

Kring, D. C., M. K. Parish, W. M. Milewski, and B. Connell (2011), Simulation of maneuvering in waves for a high-speed surface effect ship, in *Proc. 11th International Conference on Fast Sea Transportation, FAST.*

Larsson, L., L. Broberg, and K. Kim (1990), New viscous and inviscid cfd techniques for ship flows, in *International Conference on Numerical Ship Hydrodynamics, 5th.*

Larsson, L., L. Broberg, and C.-E. Janson (1991), A zonal method for predicting external automobile aerodynamics.

Larsson, L., F. Stern, and M. Visonneau (2010), Numerical ship hydrodynamics, in *Gothenburg 2010 a Workshop on Numerical Ship Hydrodynamics, Gothenburg*, pp. 1–32, Springer.

Lee, S.-J., H.-R. Kim, W.-J. Kim, and S.-H. Van (2003), Wind tunnel tests on flow characteristics of the kriso 3,600 teu containership and 300k vlcc double-deck ship models, *Journal of Ship Research*, *47*(1), 24–38.

Lemmerman, L., and V. Sonnad (1979), Three-dimensional viscous-inviscid coupling using surface transpiration, *Journal of Aircraft*, *16*(6), 353–358.

Lighthill, M. (1958), On displacement thickness, *Journal of Fluid Mechanics*, *4*(4), 383–392.

Longo, J., J. Shao, M. Irvine, and F. Stern (2007), Phase-averaged piv for the nominal wake of a surface ship in regular head waves, *Journal of fluids engineering*, *129*(5), 524–540.

Maniar, H. D. (1995), A three dimensional higher order panel method based on B-splines, Ph.D. thesis, Massachusetts Institute of Technology.

Menter, F., and T. Esch (2001), Elements of industrial heat transfer predictions, in *16th Brazilian Congress of Mechanical Engineering (COBEM)*, vol. 109.

Menter, F. R. (1994), Two-equation eddy-viscosity turbulence models for engineering applications, *AIAA Journal*, *32*(8), 1598–1605.

Morino, L. (1986), Helmholtz decomposition revisited: vorticity generation and trailing edge condition, *Computational Mechanics*, *1*(1), 65–90.

Morino, L. (1990), Helmholtz and Poincaré potential-vorticity decompositions for the analysis of unsteady compressible viscous flows, *Developments in Boundary Element Methods*, *6*.

Morino, L., and G. Bernardini (2001), Singularities in BIEs for the Laplace equation; Joukowski trailing-edge conjecture revisited, *Engineering Analysis with Boundary Elements*, *25*(9), 805–818.

Morino, L., G. Caputi Gennaro, R. Camussi, and U. Iemma (2007), On the vorticity generated sound: a transpiration-velocity/power-spectral-density approach, *AIAA Paper*, *3400*, 21–23.

Moukalled, F., L. Mangani, and M. Darwish (2015), *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM® and Matlab*, vol. 113, Springer.

OpenFOAM (2016), The OpenFOAM Foundation, http://www.openfoam.org.

Paulsen, B. T., H. Bredmose, and H. B. Bingham (2014), An efficient domain decomposition strategy for wave loads on surface piercing circular cylinders, *Coastal Engineering*, *86*, 57–76.

Pope, S. B. (2001), *Turbulent flows*, IOP Publishing.

Rosemurgy, W., D. Edmund, K. Maki, and R. Beck (2012), A velocity decomposition approach for steady free-surface flow, in *29th Symposium on Naval Hydrodynamics, Gothenburg, Sweden*.

Rosemurgy, W. J. (2014), A velocity decomposition approach for lifting and free-surface flow, Ph.D. thesis, The University of Michigan.

Rosemurgy, W. J., R. F. Beck, and K. J. Maki (2016), A velocity decomposition formulation for 2D steady incompressible lifting problems, *European Journal of Mechanics-B/Fluids*, *58*, 70–84.

Rosenhead, L. (1963), *Laminar Boundary Layers*, Clarendon Press.

Sagaut, P. (2006), *Large eddy simulation for incompressible flows: an introduction*, Springer Science & Business Media.

Schlichting, H., and K. Gersten (2000), *Boundary-Layer Theory*, Springer.

Sclavounos, P. (1987), A note on a galerkin technique for integral equations in potential flows, *Journal of Engineering Mathematics*, *21*(2), 101–114.

Serrin, J. (1959), Mathematical principles of classical fluid mechanics, in *Fluid Dynamics I/Strömungsmechanik I*, pp. 125–263, Springer.

Sweby, P. K. (1984), High resolution schemes using flux limiters for hyperbolic conservation laws, *SIAM Journal on Numerical Analysis*, *21*(5), 995–1011.

Vukčević, V., H. Jasak, and Š. Malenica (2016a), Decomposition Model for Naval Hydrodynamic Applications, Part I: Computational Method, *Ocean Engineering*, *121*, 37–46.

Vukčević, V., H. Jasak, and Š. Malenica (2016b), Decomposition model for naval hydrodynamic applications, part ii: Verification and validation, *Ocean Engineering*, *121*, 76–88.

Wehausen, J. V. (1973), The wave resistance of ships, *Advances in Applied Mechanics*, *13*, 93–245.

Weller, H. G., G. Tabor, H. Jasak, and C. Fureby (1998), A tensorial approach to computational continuum mechanics using object-oriented techniques, *Computers in Physics*, *12*(6), 620–631.

Wetzel, T. G., R. L. Simpson, and C. J. Chesnakas (1998), Measurement of three-dimensional crossflow separation, *AIAA Journal*, *36*(4), 557–564.

Wilcox, D. C. (1998), *Turbulence modeling for CFD*, vol. 2, DCW industries La Canada, CA.

Woolliscroft, M., and K. Maki (2016), A fast-running cfd formulation for unsteady ship maneuvering performance prediction, *Ocean Engineering*, *117*, 154–162.

Yoshizawa, A., and K. Horiuti (1985), A statistically-derived subgrid-scale kinetic energy model for the large-eddy simulation of turbulent flows, *Journal of the Physical Society of Japan*, *54*(8), 2834–2839.

Zhang, Y., M. Peszynska, and S. Yim (2013), Coupling of viscous and potential flow models with free surface for near and far field wave propagation, *International Journal of Numerical Analysis and Modeling, Series B*, *4*(3), 256–282.

Zhang, Y., S. C. Yim, and F. Del Pin (2015), A nonoverlapping heterogeneous domain decomposition method for three-dimensional gravity wave impact problems, *Computers & Fluids*, *106*, 154–170.

Zhu, J., T. Liu, L. Liu, S. Zou, and J. Wu (2015), Causal mechanisms in airfoil-circulation formation, *Physics of Fluids*, *27*(12), 123,601.