# Prediction of Forces and Moments in Multiphase Naval Simulations

by

Chunhui Liu

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Mechanical & Mechatronics Engineering

Waterloo, Ontario, Canada, 2019

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

# Statement of Contributions

I hereby declare that this thesis incorporates two manuscripts resulting from the collaborative work with Prof. Jean-Pierre Hickey from the University of Waterloo, Prof. Xiaohua Wu from the Royal Military College of Canada, and Mr. Kevin McTaggart from DRDC. Part of Chapter 3, 4, 5 and the Appendix A have been submitted to the Journal of Fluid Engineering.

In all works, the numerical simulations, data analysis, and writing were performed by the author. The contributions of co-authors were primarily through guidance and review of the text.

I certify that all the manuscripts only contain work completed during my registration as a graduate student at the University of Waterloo.

# Abstract

This thesis investigates the prediction of forces and moments in the static and dynamic planar motion mechanism (PMM) simulations of ship manoeuvring problems using the open-source Computational Fluid Dynamics (CFD) software package, OpenFOAM. Three well-established ships are used to investigate the static and dynamic PMM simulations: the Wigley hull (an analytical geometry), the DTC hull (a commercial ship), and the DTMB 5415 (a warship). The numerical simulations are performed using the InterFoam (for static cases) and InterDyMFoam (for dynamic cases) packages in OpenFOAM, which are incompressible multiphysics CFD solvers based on the volume of fluid (VOF) method to account for the multiple phases in a continuous regime (without needing to track the discrete air-water interface); the turbulence is modelled using Reynolds Averaged Navier-Stokes (RANS) approximations. The pressure and velocity are decoupled using the PIMPLE algorithm. The semi-implicit variant of multi-dimensional limiter for explicit solution (MULES) is used to solve the VOF Equation. Implicit schemes are used for time marching: local Euler scheme for steady cases; Euler scheme for unsteady cases. Two types of meshes are used in the current simulation: body-conforming mesh (generated by Pointwise) and castellated mesh (generated by snappyHexMesh tool). The castellated mesh is non-body conforming and generated via successive mesh refinement and adaptation using open-source software. A dynamic mesh technique is used in dynamic PMM simulations. Despite the lower accuracy of the castellated mesh generation method and the mesh deformation in dynamic PMM simulations, the results show an overall good agreement with experimental data and published numerical results. The relative errors remain small–for most of the static and dynamic cases under consideration–after a grid convergence study. There are three specific cases in which the error is significant: (1) at a high Froude number; (2) at a large drift angle; (3) for the specific combined yaw and drift case. In conclusion, the numerical method is capable of predicting force and moment coefficients for static and dynamic PMM simulations with good accuracy outside the three specific cases when the error is non-negligible.

# Acknowledgements

I would like to thank my supervisor Prof. Jean-Pierre Hickey for your invaluable help and guidance in this master research project. From the very beginning of the background knowledge of the ship maneuvering problem and the usage of OpenFOAM, to the thesis, you have helped me too much and taught me too many lessons in both of research and life. I would like to thank my co-supervisor Prof. Xiaohua Wu, your trust has made this thesis possible. I am also thankful to DRDC and the University of Waterloo for their financial support, allowing me to finish this research project. Thanks Mr. Kevin McTaggart for many constructive suggestions in the thesis.

Many thanks to my committee members for constructive suggestions. I would also like to thank all the people in our MPIL research group, your help in my research and in my life has made everything different. Thanks Nich for editing my thesis. I am deeply indebted to Mr. Meng Wei of the Memorial University of Newfoundland, thanks for your suggestions in the simulation of Wigley hull and Stokes $1^{st}$ wave cases. Thanks Jin Sing Sia for your four month cooperation in this project, and your help in the revision of the technique report. I am very grateful for the friendship of the past and present students in the Prof. Lien group. Thanks Xueqing (Caddie) Zhang, and Dr. Xiao Ren, your optimistic attitudes always remind me the hope for tomorrow. Thanks Dorothy, Michael, Laura, Bill and other volunteers in FOCUS, for continuously helping international students (including me) and scholars improve oral speaking and adapt to the life in Canada better.

Finally, I would like to thank my parents and my younger sister. Your support and trust keeps me progressing in my career.

# Table of Contents

# List of Figures

# List of Tables

# Abbreviations

**6DoF** Six Degrees of Freedom

**CFD** Computational Fluid Dynamics

**DES** Detached Eddy Simulation

**DTC** Duisburg Test Case

**DTMB** David Taylor Model Basin

**FFP** Fixed-Flux Pressure

**FV** Fixed-Value

**IIHR** Iowa Institute of Hydraulic Research

**INSEAN** Italian Ship Model Basin

**JBC** Japan Bulk Carrier

**KCS** KRISO Container Ship

**kqRWF** Wall Function for Turbulence Kinetic Energy

**MWD** Moving Wall Displacement

**MWV** Moving Wall Velocity

**nutkRWF** Rough Wall Function for Turbulence Kinetic Eddy Viscosity

**omegaWF** Wall Function for Specific Rate of Dissipation

**ONRT** ONR Tumblehome Ship

**OpenFOAM** Open-source Field Operation and Manipulation

**PIOV** Pressure-Inlet/Outlet Velocity

**PMM** Planar Motion Mechanism

**RANS** Reynolds-Averaged Navier-Stokes

**SSRC** Ship Safety Research Center

**TP** Total Pressure

**VHFR** Variable Height Flow Rate

**VOF** Volume of Fluid

**ZG** Zero-Gradient

# Nomenclature

| | |
|---|---|
| $\alpha$ | volume fraction |
| $\beta$ | drift angle |
| $\ddot{\theta}$ | angular acceleration |
| $\dot{\theta}$ | angular velocity |
| $\dot{v}$ | sway acceleration |
| $\gamma$ | diffusivity coefficients |
| $\mu$ | dynamic viscosity |
| $\mu_t$ | turbulent viscosity |
| $\nu$ | kinematic viscosity |
| $\omega$ | turbulent dissipation rate |
| $\omega_0$ | angular frequency |
| $\omega_{log}$ | turbulent dissipation based on standard Reynolds number wall function |
| $\omega_{vis}$ | turbulent dissipation based on low Reynolds number wall function |
| $\rho$ | density |
| $\rho_{air}$ | density of air |
| $\rho_{water}$ | density of water |
| $\tau_{ij}$ | viscous stresses |

| | |
|---|---|
| $\tau_{t_{ij}}$ | turbulent stresses |
| $\theta$ | yaw angle |
| $\theta_{max}$ | maximum yaw angle |
| $B$ | breadth of Wigley hull |
| $B_{wl}$ | wet surface under rest waterline without appendages |
| $C_B$ | block coefficient |
| $CFL$ | courant number |
| $Fr$ | Froude number |
| $g_i$ | gravitational acceleration |
| $h$ | height of wave |
| $k$ | turbulent kinetic energy |
| $L_{pp}$ | ship length between perpendicular |
| $L_x$ | computational domain size in spanwise direction |
| $L_x$ | computational domain size in streamwise direction |
| $L_x$ | computational domain size in vertical direction |
| $N$ | yaw moment |
| $N'$ | yaw moment coefficient |
| $p$ | pressure |
| $p_{rgh}$ | hydrodynamic pressure |
| $r'$ | non-dimensional angular velocity |
| $Re$ | Reynolds number |
| $S_{ij}$ | strain rate |
| $T$ | draft of Wigley hull |

| | |
|---|---|
| $T_m$ | mid-ship draught |
| $u$ | streamwise velocity |
| $u'$ | non-dimensional forward velocity |
| $V$ | volume displacement |
| $v$ | sway velocity |
| $v'$ | non-dimensional sway velocity |
| $X$ | streamwise force |
| $x$ | streamwise direction |
| $X'$ | streamwise force coefficient |
| $x_g$ | movement of mesh point |
| $Y$ | side force |
| $y$ | spanwise direction |
| $y^+$ | non dimensional first layer distance to the wall |
| $Y'$ | side force coefficient |
| $y_{max}$ | maximum sway amplitude |
| $z$ | vertical direction |

# Chapter 1

# Introduction

Manoeuvrability is one of the most important performance indicators of ship navigation. There are several ways to estimate the manoeuvrability characteristics of a ship, namely: theoretical approaches, experimental/empirical models and numerically via Computational Fluid Dynamics (CFD). The theoretical approach is limited to slender bodies and does not consider the interaction between the hull and the appendages, nor the more complex non-linear effects. Whereas the theory-based approaches are best suited for order-of-magnitude manoeuvrability approximations. Empirical modelling driven by experimental data is the traditional means for ship maneuverability estimation. This approach typically relies on the measurement of forces and moments on a model ship undergoing static and dynamic Planar Motion Mechanism (PMM) tests. These experimental tests are expensive and time consuming, as special experimental platforms are required. Thanks to increasing computational power, CFD is providing researchers with the ability to numerically compute static and dynamic stability derivatives on arbitrarily complex ship geometries. The ship manoeuvring simulations represent a truly multi-physics/multi-scale problem in which many modelling assumptions must be made. As a result of the complexity of the physics, the reliability of CFD represents the main challenge to a broader adoption of this technology. The present work seeks to assess the accuracy in the prediction of forces and moments in ship maneuvring simulations, using the open-source CFD package, OpenFOAM (Open-source Field Operation and Manipulation, version 17.12 [39]). To assess OpenFOAM's ability to compute ship maneuvers, well-defined test cases with experimental data are selected as benchmarks. In particular, this work deals with pure drift, pure sway, pure yaw, combined yaw and drift simulations.

The four types of motions in the current simulations are presented in Figure 1.1. The carriage speed $U$ is fixed based on the Froude number. Figure 1.1 (a) presents the static drift test, in which the drift angle $\beta$ keeps constant and the velocity in ship coordinate reduces to $u = U\cos\beta$, $v = U\sin\beta$. As shown in Figure 1.1 (b), in the pure sway test, the ship axis is parallel to the direction of the forward velocity $U$ and the ship is prescribed the periodic sway position $y$, the sway velocity $v$, and the sway acceleration $\dot{v}$ as following:

$$y = -y_{max}\sin(\omega_0 t) \tag{1.1}$$

$$v = -\omega_0 y_{max}\cos(\omega_0 t) \tag{1.2}$$

$$\dot{v} = \omega_0^2 y_{max}\sin(\omega_0 t) \tag{1.3}$$

The pure yaw motion is defined as:

$$\theta = \theta_{max}\cos(\omega_0 t) \tag{1.4}$$

$$\dot{\theta} = -\theta_{max}\omega_0\sin(\omega_0 t) \tag{1.5}$$

$$\ddot{\theta} = -\theta_{max}\omega_0^2\cos(\omega_0 t) \tag{1.6}$$

where $\theta_{max} = \arctan(\frac{v}{U})$ is set to keep the ship axis always tangent to its path, as shown in Figure 1.1 (c). The towing path of the combined yaw and drift test is presented in Figure 1.1 (d). The ship is prescribed yaw angular velocity $\dot{\theta}$ and yaw angular acceleration $\ddot{\theta}$ as a function of time by Equation 1.6 with constant drift angle $\beta$.

The objective of this study is to investigate the accuracy of the prediction of ship manoeuvring forces and moments in multiphase naval simulations. Three ships are used as benchmark cases: the Wigley hull, the DTC hull, and the DTMB 5415. Static and dynamic simulations are performed using the interFoam solver and the interDyMFoam solver, respectively. Dynamic PMM simulations are performed using the dynamic mesh technique [41], in which the ship motion is prescribed in advance and the mesh is deformed without topological changes. The errors of forces and moments in the simulations are investigated by comparison to the model test data [45, 19, 77, 95].

The thesis is organized as follows. A literature review of maneuvring experiments and simulations is provided in Chapter 2. In Chapter 3, modelling and numerical details of the OpenFOAM solver related with static and dynamic PMM simulations are presented. Chapter 4 summarizes the numerical simulation design, including the coordinate system, non-dimensionalization, computation domain, boundary conditions, geometries, mesh and computational resources. The results of static and dynamic PMM simulations of the Wigley, DTC and DTMB 5415 are discussed in Chapter 5. In Chapter 6, a final conclusion is given.

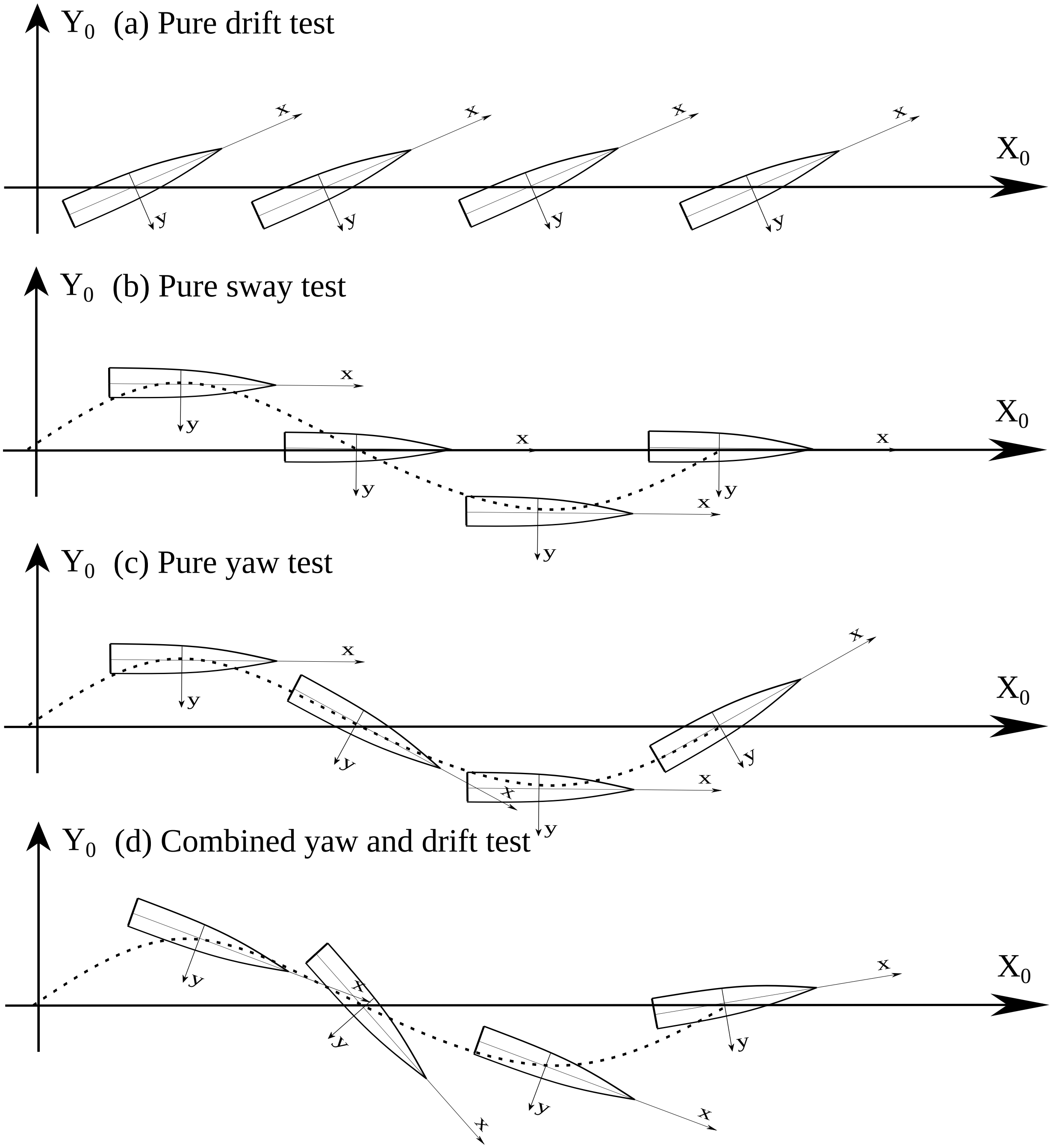


Figure 1.1: Towing paths of static and dynamic PMM tests: (a) pure drift, (b) pure sway, (c) pure yaw, (d) combined yaw and drift.

## 1.1 Contributions of the Thesis

The main contributions of the thesis can be summarized as follows:

- Validate the numerical method for static and dynamic PMM simulations
- Implement of the multi-motion package to the simulation of dynamic PMM cases
- Compare the effect of mesh topology (body-conforming/castellated mesh) on drag and wave predictions
- Investigate the prediction error of force and moment coefficients, and the wave profile in the simulation using experimental data
- Study the Froude number effect on the drag coefficient in static drift 0 cases
- Investigate the effect of drift angle on force and moment coefficients, and the hydrodynamic pressure distribution on the interface

# Chapter 2

# Literature Review

## 2.1 Manoeuvring Experiments

To assess the predictive ability of CFD solvers for ship manoeuvrability simulations, well-defined experimental test cases must be devised. Of the numerous ship manoeuvring test cases, the US Navy Combatant model 5415 (here denoted as DTMB 5415 [18] as an acronym for the David Taylor Model Basin) provides a well-studied, sub-scale geometry that was used as part of the Workshop on Verification and Validation of Ship Manoeuvring Simulation Methods since the 2000s [55]. This sub-scale ship was inspired by a preliminary design of a US Navy surface combatant ship from the early 1980s. An extensive experimental and parametric database in deep water, including PMM tests, serves as a well-defined benchmark against which numerical tools can be assessed. Other established test cases have also been reported in the literature [55]. Longo [64] presented results for DTMB 5512 in the Iowa Institute of Hydraulic Research (IIHR) towing-tank [34]. Their experimental results include resistance calculations, sinkage and trim, wave profiles and nominal wake tests and uncertainty assessment. Similarly, Olivieri *et al.* [77] carried out towing-tank experiments for the Italian Ship Model Basin (INSEAN) 2340 model in the range of Froude number between 0.05 and 0.45 for free model conditions. Comparative assessments between the various international towing-tanks were undertaken by Stern *et al.* [100]. Their work provides a comparative experimental database, at overlapping tests conditions, between three institutes: IIHR, INSEAN, and DTMB. The results include a detailed uncertainty assessment on the test data from these experiments. A similar uncertainty assessment methodology was presented by Longo [65] for towing-tank tests using DTMB 5512.

Of the many types of ship manoeuvring tests that can be conducted, the most common test is the static and dynamic planar motion mechanism (PMM) test. Simonsen [95] conducted experiments of pure drift, pure sway, pure yaw, yaw and drift to provide PMM data for comparison with the data from IIHR towing tank. The 1:35.48 scale model of DDG51 (DTMB 5415) was constrained in roll but free to heave and pitch, so trim and sinkage were also obtained. Forces were measured in the longitudinal direction of the ship and perpendicular to this direction. The yaw moment was taken with respect to the midship position at the center between forward perpendicular (Fp) and after perpendicular (Ap). Similarly, Benedetti *et al.* [5] conducted PMM tests for the DTMB 5415 with a scale ratio 1:24.83. Stern *et al.* [117, 119] summarized the benchmark data for CFD validation; in particular, they considered the force, moment, motion measurements for the DTMB 5512. For the free roll motion, Lee *et al.* [57, 58] performed free roll decay tests in calm water with both an intact and damaged ship. These six degrees of freedom (6DoF) motion tests were performed with regular waves for a passenger ship provided by the Ship Safety Research Centre (SSRC) at the University of Strathclyde.

## 2.2 Manoeuvring Simulations

### 2.2.1 International Workshops

International workshops on computational ship hydrodynamics have been held seven times since 1980 in order to standardize testing for naval hydrodynamics. The workshops were held in Gothenburg, Sweden in 1980, 1990, 2000 and 2010 [56], while they were held in Tokyo, Japan in 1994, 2005 and 2015 [106]. The first workshop in 1980 focused on simulations based on simplified boundary layer equations. While most of these simplified methods were capable of predicting ship boundary layer characteristics with reasonable accuracy, they failed in predicting the flow at the stern and in the wake of the vessels. In the second workshop in 1990, most simulations were conducted using Reynolds-Averaged Navier-Stokes (RANS) turbulence closure models. These numerical tools provided a better description of the flow near the propeller plane but still maintained an inaccurate prediction of the bilge vortex. In the prediction of viscous flow for the Series 60 cargo ship with free surface, multi-phase modelling was added in the workshop in Tokyo, 1994. The Series 60 wave profile was predicted accurately with the RANS simulation, while the damping of the wave pattern was observed due to the insufficient mesh resolution and numerical dissipation.

The increasing availability of computational power resulted in a drastic increase in the number and complexity of the ship models during the 2000 workshop [56] in Gothenburg, Sweden. The simulations included naval hydrodynamics with complex geometries, multi-physics and environment; self-propulsion and the numerical and modelling uncertainties were also presented by participants. Three modern ships were investigated: the KRISO Tanker Ship (KVLCC2), the KRISO Container Ship (KCS), and the U.S. Navy Combatant (DTMB 5415); these ship models have been used ever since. In the fifth workshop in Tokyo 2005, seakeeping and manoeuvring cases were shown and the number of test cases increased significantly, with the same three hulls as in the Gothenburg 2000 workshop. The sixth workshop in 2010 covered the areas such as resistance, local flow, self-propulsion, and seakeeping with the three previous hulls. In the workshop Tokyo 2015, two new hulls were introduced, namely: the Japan Bulk Carrier (JBC) and the ONR tumblehome ship (ONRT). Besides RANS and detached eddy simulation (DES) technique, energy saving devices were presented in the workshop.

### 2.2.2 Numerical Approaches

The Workshop on Verification and Validation has served as the gold standard for the assessment of the predictive ability of numerical simulations for ship maneuverability. These experimental data have been used by a number of researchers to assess the predictive ability of numerical tools. The steady drift cases of DTMB 5415 have been investigated by a number of research groups. Wood *et al.* [116], using the commercial CFD solver ANSYS-CFX [1], numerically investigated the steady drift case of the DTMB 5415 with RANS and Volume of Fluid VOF approach. Wave profiles, local wave elevations and global resistance were obtained; the comparison between experimental and computational results showed a better prediction using structured mesh than using unstructured mesh. Jones *et al.* [44] simulated the static drift case for DTMB 5415 using Fluent and noticed that the results were sensitive to the angle of the grid with respect to the free surface waterline. Ma *et al.* [66] investigated a static drift case for the DTMB 5415 using an in-house CFD code with a level-set method to capture the air-water interface. The total friction and wave profile along the ship were in good agreement with experimental data, although it was noted that a smaller amplitude of wave on the hull was the result of the dissipation due to the mesh and numerical scheme. More recently, Bhushan [7] summarized the CFD results for DTMB 5415 for static drift $\beta = 0$ and 20 degrees using experimental data [118]. Five solvers are used and evaluated: CFDShip-Iowa V4.5 [11], ReFRESCO [111], ISIS-CFD [85], NavyFOAM [26], and STAR-CCM+ [29]. For drift $\beta = 0°$ results, the relative error of resistance was smaller than 2% for all solvers, compared with experimental data; the wave profiles were predicted well without significant difference between VOF and level-set methods. The predicted data of forces and moments at drift $\beta = 20^\circ$ was within 4.5% deviation from experimental data. The wave elevation near the hull was predicted well, while large errors were observed away from the hull. RANS showed excessive decay of vortices, while DES presented grid sensitivity and shear stress depletion.

The small-amplitude ship motion was investigated numerically using the solid-body motion method and dynamic mesh method, which are implemented as open-source tools in the OpenFOAM solver. Vuko *et al.* [112] investigated pure sway motion for DTMB 5512 and MOERI container ship hulls using the OpenFOAM extension: foam-extend [42]. The mesh motion was modelled as the rigid body motion and the cells moved at each time step while updating the convective mesh flux. More recently, Islam *et al.* [36] investigated static drift, pure yaw and pure sway motion for KCS by solid body motion solver in OpenFOAM. The predicted forces, yaw moment and hydrodynamic derivatives were compared with the experimental data. The results showed good agreement except for some of the pure yaw cases. Using a dynamic mesh technique, Henry [82] performed roll decay simulations of DTMB 5415 and DTC hull in OpenFOAM. The three dimensional forced roll motion was simulated by sliding interface method in OpenFOAM. Shen *et al.* [89] investigated added resistance, heave and pitch motions of DTMB 5512 in head waves, using the naoe-FOAM-SJTU solver [114] based on the OpenFOAM platform. The wave is generated by setting a time dependent inlet boundary condition, and a sponge layer is setup at the outlet of the computational domain to avoid the wave reflection; a dynamic mesh method is used to capture the 6DoF motion. The simulation result was validated against experimental data [27, 28, 35].

For ship simulations with large amplitude motion, the overset method is generally the preferred approach. Sakmoto *et al.* [87] verified and validated the forces and yaw moment coefficients, and hydrodynamic derivatives in static and dynamic PMM simulations for DTMB 5415 using the in-house code CFDShip-Iowa [11]. This code is an unsteady RANS solver with overset grid technique to deal with dynamic ship motions and local grid refinements. Carrica *et al.* [9] investigated steady turn and zig-zag motions for DTMB 5415 using a hierarchy of body technique. More recently, Shen [91] developed a dynamic overset grid within OpenFOAM (although not publicly available) and performed simulations of zig-zag motions with self-propulsion for a full dynamic simulation case. Besides the open source and in-house code, commercial tools have also been used for dynamic PMM simulations. Oldfield *et al.* [76] presented static and dynamic PMM simulation results for DTMB 5415 using STAR-CCM+.

Besides DTMB 5415, the Wigley hull [45], as an analytically defined geometry, are used to benchmark the modelling and numerical errors. These errors arise for one of the most simplest ship hull cases. This canonical geometry permits a simple analysis of predictive naval simulations abstracted from the geometric complexities of the more realistic hulls. For this geometry, Chen [12] compared a number of theoretical predictions of the wave-resistance coefficient of the Wigley hull with experimental data. While most theoretical predictions show larger discrepancies compared with the experimental data at high Froude numbers, the first-order slender-ship approximation [13] is in good agreement with the experimental data at different Froude numbers. Ciortan [14] simulated a steady zero drift angle case of the Wigley hull using a slightly compressible code. The resistance and the wave profile were predicted well with higher computational efficiency compared to other incompressible solvers. More recently, Heredero [79] investigated the vortical structures and associated instabilities for flows around the Wigley hull at large drift angles (from 10° to 60°) using RANS and DES solver in CFDShip-Iowa-V4. A strong correlation between the vortical structures and the surface wave pattern is observed. When the drift angle is smaller than 45°, the flow remains steady; the increase of drift angle causes the increase of the strength and the complexity of vortices. The flow becomes unsteady at drift 45°, while the flow pattern remains the same. A large recirculation region is generated from the aft to the fore end, as the drift angle increases from 45° to 60°. The numerical simulation extends engineers' understanding of the interaction among the flow separation, the vortex structure and the free surface, especially at extreme drift angles, in which cases no experimental data available.

Another well-established ship manoeuvring case is the Duisburg Test Case (DTC), which is used as a benchmark case for modern container vessels. This geometry provides an additional validation case on the force estimates and it is receiving increasing attention for benchmarking CFD solvers. For this case, el Moctar *et al.* [19] presented both experimental and numerical simulation data for the DTC model with resistance test and roll decay test. Kinaci *et al.* [52] researched ship propeller interaction problem of DTC by numerical simulations whereas Liu *et al.* [63] presented a three-dimensional nonlinear time domain method for the simulation of 6DoF motion of the case. In the same year, Ley *et al.* [59] predicted the added resistance of the DTC in waves using the commercial software Comet and the open-source alternative OpenFOAM. More recently, el Moctar *et al.* [20] further investigated the effects of waves on manoeuvring in both experiments and CFD. The full-nonlinear effects were investigated by He *et al.* [31] who implemented an overset grid approach for a three degrees of freedom simulation, which is capable of simulating larger ship movements as well as the bottom hull interaction. Recently, Terziev *et al.* [105] investigated the behavior of trim, sinkage and resistance for DTC in shallow waters at varying channel cross-sections and speeds. The numerical results in Star-CCM+ simulations were compared to those predicted by the slender body theory and it showed good agreement in the low speed range.

| **Software** | **Type** | **Numerical techniques** | **Relevant literature** |
|---|---|---|---|
| OpenFOAM | open source | FVM, VOF, RANS, dynamic mesh | [36, 59, 82, 112] |
| naoe-Foam-SJTU | in-house | FVM, VOF, RANS, DES, overset | [88, 89, 91, 113, 114] |
| Navy-Foam | in-house | FVM, VOF, RANS, overset | [26, 50, 51] |
| CFDShip-Iowa | in-house | FDM, level set, RANS, DES, overset | [9, 86, 87] |
| ShipMotion | in-house | FVM, density function [48], RANS, overset | [49] |
| WISDAM-X | in-house | FVM, density function [48], RANS, overset | [78] |
| ANSYS-CFX | commercial | FVM, VOF, RANS, dynamic mesh | [15, 44, 61, 81, 108, 116] |
| Star-CCM+ | commercial | FVM, VOF, RANS, overset | [30, 68, 76, 96] |

Table 2.1: Summary of CFD software for ship manoeuvring simulations.

In Table 2.1, a comprehensive summary of the known CFD software used for ship manoeuvring simulations is presented as an overview. FVM represents the finite volume method, and FDM is the finite difference method. Both the in-house codes and commercial software are capable of simulating static and dynamic PMM cases, however, they are either not available publicly or need a licence which is expensive. The current work is based on the open source software OpenFOAM, in which the overset method is not available. There has been a limited number of research of PMM simulations. Most of the research was based on in-house codes or commercial. To the author's knowledge, there has been no dynamic PMM simulation using dynamic mesh technique in the OpenFOAM platform. This is the gap that this paper aims to address. Using the numerical simulation, a deeper understanding of the numerical error of forces and moments in static and dynamic PMM simulations can be obtained, which would provide extensions for ship engineers.

# Chapter 3

# Computational Method

## 3.1 Physical Modelling

### 3.1.1 Governing Equations

The interFoam solver in the OpenFOAM package is used for multiphase ship simulation. The governing equations in interFoam are the incompressible Navier-Stokes equations within a VOF framework [33]. The continuity and momentum equations, are written as:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0 \tag{3.1}$$

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

where $u_i$ represents the velocity in the $i^{th}$ direction, $\rho$ is the density, $p$ is the pressure, and $g_i$ is the gravitational acceleration. The terms $\tau_{ij}$ and $\tau_{t_{ij}}$ are respectively the viscous and turbulent stresses. The velocity, density, and pressure in the above equations corresponds to the Reynolds-averaged values. The fluctuating components account for the turbulent stress terms.

The key of VOF approach is the estimation of the volume fraction, $\alpha$, to capture the free surface. A convective transport equation for the volume fraction is solved:

$$\frac{\partial \alpha}{\partial t} + \frac{\partial (\alpha u_i)}{\partial x_i} = 0 \tag{3.3}$$

This transport equation is analogous to a scalar transport equation and allows a time-dependent evaluation of the volume fraction at each spatial location in the computational domain. Generally, $\alpha$ has a value of 0 (in pure air) and 1 (in pure water) except near the interface where $\alpha$ can take any value between 0 and 1. The volume fraction is needed to compute the local density using Equation 3.4, which, in turn, is required to solve the mass Equation 3.1, and momentum Equations 3.2. As the conservation equations are valid for both the liquid and gaseous states of the fluid, the mixture density in governing equations is computed based on a weighted-average density of the liquid and gas phases:

$$\rho = \alpha\rho_{water} + (1-\alpha)\rho_{air} \tag{3.4}$$

where $\rho_{water}$ and $\rho_{air}$ represent the constant densities of water and air.

### 3.1.2 Turbulence Modelling

In the context of RANS, the influence of the fluctuating component of turbulence is modelled using standard turbulence closure models. Two-equation closure models represent a good compromise amongst robustness, computational efficiency, and accuracy. The present work focuses on the $k-\omega$ SST model, which represents one of the standard turbulence models for engineering simulations. The $k-\omega$ model was first proposed by Menter in 1994 [69]. OpenFOAM implements the extension of this classical model with a shear stress transport formulation [70]. This model is a combination of two common two-equation turbulent models, namely the $k-\omega$ and $k-\epsilon$ models with a blending function. In fact, the $k-\omega$ and $k-\epsilon$ models are used in near wall and far field, respectively, as the $k-\omega$ model predicts flow separation with adverse pressure gradient better than $k-\epsilon$ model in near wall boundary layer. Meanwhile, away from solid walls, the $k-\epsilon$ model is favoured.

The turbulent kinetic energy $k$ and turbulent dissipation rate $\omega$ equations are:

$$\frac{\partial k}{\partial t} + \frac{\partial \rho u_i k}{\partial x_i} = \tilde{G} - \beta^* k\omega + \frac{\partial[(\nu + \alpha_k \nu_t)\frac{\partial k}{\partial x_i}]}{\partial x_i} \tag{3.5}$$

$$\frac{\partial \omega}{\partial t} + \frac{\partial \rho u_i \omega}{\partial x_i} = \gamma S^2 - \beta\omega^2 + \frac{\partial[(\nu + \alpha_\omega \nu_t)\frac{\partial \omega}{\partial x_i}]}{\partial x_i} + (1-F_1)CD_{k\omega} \tag{3.6}$$

The coefficients are listed in Table 3.1. $F_1$ is a hybrid function combining $k-\epsilon$ and $k-\omega$ models. With the hybrid function $F_1$, $k-\omega$ model is activated near the wall and $k-\epsilon$ model is utilized in the far field. $F_1$ is defined as:

$$F_1 = \tanh(\min[\max(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}), \frac{4\alpha_{\omega 2}k}{CD^*_{k\omega}y^2}]^4) \tag{3.7}$$

where $CD^*_{k\omega}$ is defined as:

$$CD^*_{k\omega} = \max(CD_{k\omega}, 10^{-10}) \tag{3.8}$$

where $CD_{k\omega}$ is defined as:

$$CD_{k\omega} = 2\sigma_{\omega 2}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i}\frac{1}{\omega} \tag{3.9}$$

and $\tilde{G}$ is defined as:

$$\tilde{G} = min(G, c_1\beta^* k\omega) \tag{3.10}$$

where $c_1 = 10$; G is the production term:

$$G = \nu_t S^2 \tag{3.11}$$

where S is the invariant measure of the strain rate:

$$S = \sqrt{2S_{ij}S_{ij}} \tag{3.12}$$

$$S_{ij} = \frac{1}{2}(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}) \tag{3.13}$$

After solving Equations 3.5 and 3.6, turbulent viscosity $\nu_t$ is calculated by:

$$\nu_t = \frac{\alpha_1 k}{\max(\alpha_1\omega, SF_2)} \tag{3.14}$$

where $\alpha_1 = 0.31$, and $F_2$ is a hybrid function:

$$F_2 = \tanh[\max(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega})]^2 \tag{3.15}$$

In addition, all the coefficients in Equations 3.5 and 3.6 are calculated by the hybrid function $F_1$:

$$\phi = \phi_1 F_1 + \phi_2(1 - F_1) \tag{3.16}$$

where $\phi$ represents the coefficients mixed in Equation 3.5 and 3.6; $\phi_1$ and $\phi_2$ represent coefficients in $k-\omega$ and $k-\epsilon$ equations, respectively. All coefficients are listed in Table 3.1.

| $\alpha_{k1}$ | $\alpha_{k2}$ | $\alpha_{\omega1}$ | $\alpha_{\omega2}$ | $\beta_1$ | $\beta_2$ | $\gamma_1$ | $\gamma_2$ | $\beta^*$ | $\alpha_1$ | $c_1$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 0.85034 | 1.0 | 0.5 | 0.85616 | 0.075 | 0.0828 | 0.5532 | 0.4403 | 0.09 | 0.31 | 10.0 |

Table 3.1: Coefficients for $k-\omega$ SST turbulence model.

### 3.1.3 Near Wall Function

Since the typical requirement of the first grid near the wall is $y^+ \leq 1$ to use wall-resolved RANS, it is expensive and difficult in engineering problems utilizing the castellated mesh. Wall functions relax the requirement by permitting the first grid near the wall to be set in the log law region of the turbulent boundary layer. The current OpenFOAM solver in the $k-\omega$ SST turbulent model implements the wall modelling by changing the first layer's turbulent viscosity $\nu_t$, turbulent kinetic energy $k$, and turbulent dissipation rate $\omega$. $\nu_t$ is calculated by the following formula:

$$\nu_t = \begin{cases} 0, y^+ < 11.53 \\ \nu \frac{y^+ k}{\ln(Ey^+)}, y^+ \geq 11.53 \end{cases} \tag{3.17}$$

Based on the wall functioned RANS model, the zero gradient boundary condition is used for turbulent kinetic energy. On the first layer, $\omega$ is calculated based on a combination of the low Reynolds number wall function and the standard wall function:

$$\begin{cases} \omega = \sqrt{\omega_{vis}^2 + \omega_{log}^2} \\ \omega_{vis} = \frac{6.0\nu}{\beta_1 y^2} \\ \omega_{log} = \frac{k^{1/2}}{C_\nu^{1/4} \kappa y} \end{cases} \tag{3.18}$$

where $\beta_1 = 0.075$, $\kappa = 0.41$, and $C_\nu = 0.09$. $\omega_{vis}$ represents the low Reynolds number wall function when the first layer is located in the viscous layer. $\omega_{log}$ means the standard wall function method when the first layer is located in the log law region. The current hybrid function can switch between viscous and logarithmic regions according to the values of $y^+$.

## 3.2 Numerical Methods and Schemes

### 3.2.1 Discretization

The three-dimensional Navier-Stokes equations and VOF framework are solved via a finite volume method [40], as shown in Figure 3.1. As the governing equations are incompressible (we do not use a state equation to relate the thermodynamic variables), the pressure and velocity are decoupled using the PIMPLE algorithm. The PIMPLE algorithm is the OpenFOAM specific approach, which combines the classical PISO (Pressure Implicit with Splitting of Operator) and the SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) schemes.

The semi-implicit variant of multi-dimensional limiter for explicit solution (MULES) [110] solver is utilized to solve the VOF Equation 3.3. The cell limited Gauss linear method is applied for gradient schemes to improve boundedness and stability. In this way the face values do not fall outside the bounds of values in surrounding cells. The convection terms are computed by summing the fluxes on all faces which are determined through linear interpolation from the cell centers. This method is referred to as Gaussian integration within the OpenFOAM framework. Similarly, the Gaussian integration scheme is utilized for the discretization of Laplacian terms. In order to maintain second-order accuracy, an explicit non-orthogonal correction method is used as the surface normal gradient scheme. A local Euler scheme is applied in steady simulations, which is a first-order implicit method using local-time stepping. This low-order pseudo-time integration is satisfactory for steady-state simulations. A first-order implicit Euler scheme is used for the time marching in unsteady simulations.

### 3.2.2 Dynamic Mesh Method

For dynamic cases (pure sway, pure yaw, yaw and drift motion), the interDyMFoam solver is used with a dynamic deforming mesh [41]. The mesh deforms based on the movement of the ship without topological change. The displacement of each mesh point is calculated by solving the following Laplace equation at each time step:

$$\nabla.(\gamma \nabla x_g) = 0 \tag{3.19}$$

where $x_g$ is the movement of the mesh point; $\gamma$ is the diffusivity coefficients based on the square inverse of the distance between the ship and the mesh point. Figure 3.2 shows an example of mesh deformation due to the motion of a box (red) inside the computational domain.

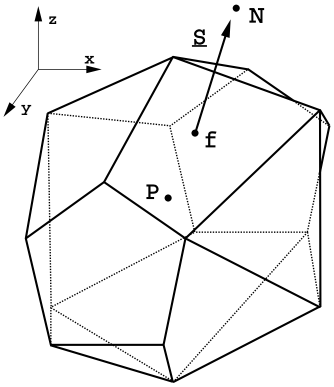


Figure 3.1: Control volume. Source [40]

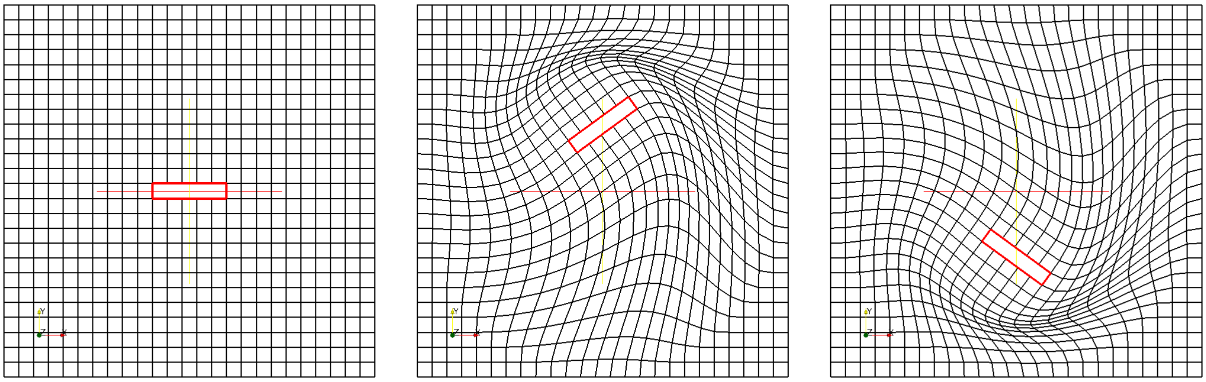


Figure 3.2: Dynamic mesh. Source [43]

### 3.2.3 Code Validation

In ship maneuvring simulations, the forces and moments are integral results, which are influenced by the complex flow phenomena. Therefore, modelling requirements including the mesh resolution criteria and turbulent models for a multiphase ship simulation were investigated systematically by decomposing the current multiphase complex ship simulation using benchmark cases with simple geometries, which is presented in the Appendix A. These simple geometry cases not only serve as code validation but also provide guidelines for ship maneuvring simulations.

# Chapter 4

# Simulation Setup

## 4.1 Coordinate System and Non-Dimensionalization

For ship manoeuvring simulations, there are two coordinate systems of interest: the earth coordinate system and the ship-fixed coordinate system. Both coordinate systems follow the right-hand rule, as shown in Figure 4.1. In the ship fixed coordinate, the $x$-axis is positive from the stern to the bow and the $z$-axis is negative in the gravity direction. The origin of the earth coordinate system is generally located at the start point of manoeuvring simulation, while the origin of the ship-fixed coordinate is taken with respect to the mid-ship position at $L_{pp}/2$. The drift angle $\beta$ is the horizontal angle between the $x$-axis of the ship and the tangent to its path. All results in the current simulations are reported in the ship coordinate system. The forward velocity $u$, sway velocity $v$, and clockwise angular velocity $r$ are non-dimensionalized by the magnitude of the ship velocity $U$ and the ship length between perpendiculars $L_{pp}$:

$$u' = \frac{u}{U}, \qquad v' = \frac{v}{U}, \qquad r' = \frac{rL_{pp}}{U} \tag{4.1}$$

The Reynolds number and the Froude number are defined as:

$$Re = \frac{UL_{pp}}{\nu}, \qquad Fr = \frac{U}{\sqrt{gL_{pp}}} \tag{4.2}$$

where $\nu$ represents fluid kinematic viscosity.

The forces and moment are non-dimensionalized as follows:

$$X' = \frac{X}{0.5\rho U^2 L_{pp} T_m}, \qquad Y' = \frac{Y}{0.5\rho U^2 L_{pp} T_m}, \qquad N' = \frac{N}{0.5\rho U^2 L_{pp}^2 T_m} \tag{4.3}$$

where $T_m$[m] is the mid-ship draught.

## 4.2 Domain Definition and Boundary Conditions

The computation domain is shown in Figure 4.2, where a deep water condition is represented. For all cases, a full domain is selected even in the cases where a symmetry plane can be defined. Seven patches are defined as boundaries in the current simulation: inlet, outlet, bottom, atmosphere, front, back and the hull, where front patch is defined at the starboard side and back patch is defined at port side of the hull. The boundary condition at the inlet, outlet and atmosphere are defined based on the velocity of the ship. The velocities for the air portion and the water portion are the same, based on the towing tank experiments. A no-slip wall boundary condition is used on the hull in both the water and air. The boundary condition of the bottom is set as symmetry plane. The boundary conditions on the front and back are defined based on the drift angle in the simulation, as shown in Figure 4.3. For cases $\beta = 0°$, the symmetry boundary condition is set on both of the front and the back patches; for cases with a non-zero drift angle, the velocity inlet boundary condition is defined for the back patch, the outlet boundary condition is defined for the front plane. The inlet flow velocity is parallel to the center plane of the domain in cases of $\beta = 0°$, while the angle between the inlet flow velocity and the center plane of domain equals to the drift angle in cases $\beta \neq 0°$. The current method to deal with drift cases can improve the accuracy of the simulation, compared to the method that rotating the hull model in the computational domain since the mesh quality would decrease with the rotation of the hull. These aspects will be discussed in Section 4.4.

The detailed information about the boundary conditions of the simulations and turbulent parameters is presented in Table 4.1. FV represents a fixed value, which is a first-type boundary condition (Dirichlet); the boundary value for the variable is explicitly prescribed. ZG is zero-gradient (second-type boundary condition, Neumann); PIOV is the pressure-inlet/outlet velocity. MWV is the moving wall velocity, which is set as a fixed type zero in static simulation and a calculated type in dynamic simulations. FFP is an abbreviation for fixed-flux pressure, which adjusts the pressure gradient to match the velocity boundary condition while TP is the total pressure. VHFR represents variable height flow rate, providing a phase fraction condition based on the local flow conditions. IO is the inlet and

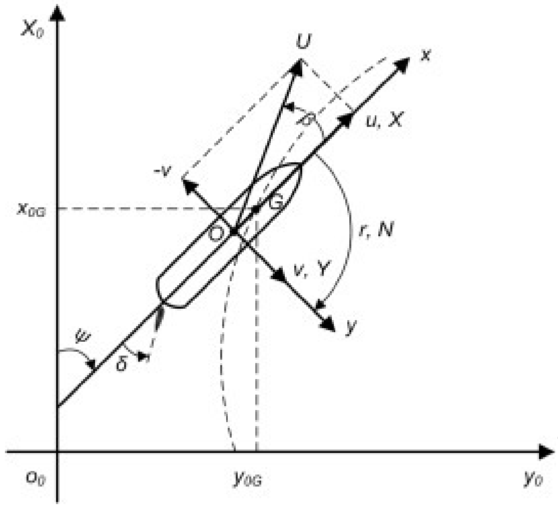


Figure 4.1: Coordinate system [62].

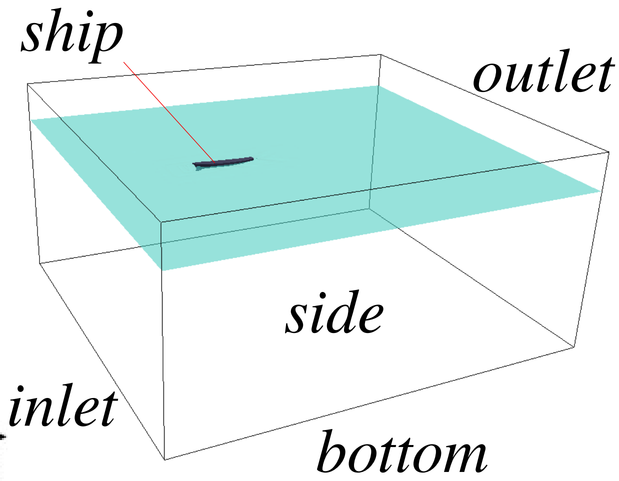


Figure 4.2: Computation boundaries.

outlet boundary condition; it provides zero gradient outflow conditions normally while it switches to fixed value if there is back flow, as shown in Figure 4.4. kqRWF is the wall function for the turbulent kinetic energy $k$. nutkRWF is the rough wall function for the turbulence kinetic eddy viscosity $\nu_t$. omegaWF is the wall function for the specific rate of dissipation $\omega$. *pointDisplacement* means the point motion displacement field and it is only prescribed in the dynamic cases. MWD represents a moving wall displacement defined by user for specific type of motion.

(a) $drift = 0°$ (b) $drift = \beta$

Figure 4.3: Boundary condition for inlet, outlet, front, and back.

|  | Inlet | Outlet | Atmosphere | Ship |
|---|---|---|---|---|
| $U$ | FV | ZG | PIOV | MWV |
| $p_{rgh}$ | FFP | ZG | TP | FFP |
| $alpha.water$ | FV | VHFR | IO | ZG |
| $k$ | FV | IO | IO | kqRWF |
| $\nu_t$ | FV | ZG | ZG | nutkRWF |
| $\omega$ | FV | IO | IO | omegaWF |
| $pointDisplacement$ | FV | FV | FV | MWD |

Table 4.1: Boundary conditions for fluid and turbulence parameters.

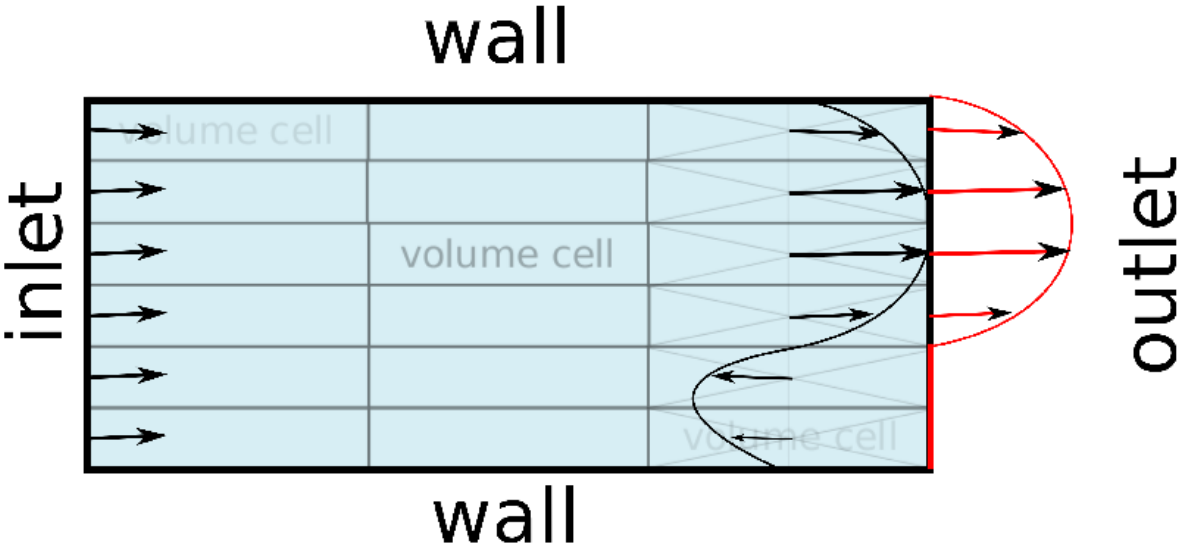


Figure 4.4: InletOutlet boundary condition. Source [3]

## 4.3 Ship Models

The Wigley hull is an analytically defined ship geometry which is used to assess the modelling requirements for multiphase ship simulation. The geometry of the Wigley hull is defined as:

$$y = \frac{B}{2}\left(1 - \left(\frac{2x}{L_{pp}}\right)^2\right)\left(1 - \left(\frac{z}{T}\right)^2\right) \tag{4.4}$$

where $L_{pp} = 2$, $B = 0.25$, and $T = 0.125$ are respectively the length, breadth, and draft of the hull. The $x$-axis is positive from stern to bow direction, the $y$-axis is positive from the port side to the starboard side, and the $z$-axis is negative in the gravity direction. The geometry of the hull above the free surface ($z \in [0, 0.04L_{pp}]$) is defined by an extrusion of the hull plane at $z = 0$ in the vertical direction. To create the Wigley hull geometry STL file, a Python code was developed, as shown in Appendix B.1. This simple analytical ship geometry is selected as it offers an easily repeatable simulation without any geometric complexity. Furthermore, it allows for a systematic study of the effect of the mesh resolution and the mesh topology (castellated mesh and body-conforming mesh) on the force prediction.

Besides the Wigley hull, another two ship models: the DTC and the DTMB 5415 (as previously discussed), are used for benchmarking OpenFOAM, as shown in Figure 4.5. The geometric files defining the outer hull of the vessels are available on the website of the Workshop on Verification and Validation of Ship Manoeuvring Simulation Methods [92]. The main parameters of the DTC and DTMB 5415 are presented in Table 4.2 and Table 4.3 respectively, where $B_{wl}$ is waterline breadth, $T_m$ is midship draught, $V$ is volume displacement, $C_B$ is block coefficient, $S_w$ is the wet surface under rest waterline without appendages. There is no bilge keel in the DTMB 5415 model, and the sinkage and trim are set based on the experimental data [95] in the current simulations.

## 4.4 Computational Domain and Mesh

The computational domain and mesh for the ships are discussed in the section. The domain size was set to be large enough for domain-independent solutions [79], especially for large drift cases. The details of the domain size followed ITTC guidelines [84]. The distance of between the waterline and the bottom plane is larger than the length of ship, in order to satisfy deep water condition [75]. Body-conforming mesh and castellated mesh is generated for the Wigley hull. Only castellated mesh is generated for DTC and DTMB due to the complexity of geometries.

Figure 4.5: The geometry of the ship model. Top: DTC; Bottom: DTMB 5415.

| DTC | Model | Full Scale |
|---|---|---|
| $L_{pp}[m]$ | 5.976 | 355.0 |
| $B_{wl}[m]$ | 0.859 | 51.0 |
| $T_m[m]$ | 0.244 | 14.5 |
| $V[m^3]$ | 0.827 | 173467.0 |
| $C_B$[-] | 0.661 | 0.661 |
| $S_w[m^2]$ | 6.243 | 22032.0 |

Table 4.2: Main parameters for DTC.

| DTMB 5415 | INSEAN | MARIN |
|---|---|---|
| Scale ratio | 24.83 | 35.48 |
| $L_{pp}[m]$ | 5.72 | 4.002 |
| $B_{wl}[m]$ | 0.768 | 0.538 |
| $L_{wl}[m]$ | 5.726 | 4.0083 |
| $T_m[m]$ | 0.248 | 0.173 |
| $V[m^3]$ | 0.554 | 0.190 |
| $C_B$[-] | 0.5060 | 0.507 |

Table 4.3: Main parameters for DTMB 5415.

### 4.4.1 Wigley Hull

The extent of the computational domain is optimized to reduce computational cost while allowing a domain size independent solution. For the present cases, the computational domain is set as: $5L_{pp}$ ($-3.5L_{pp}$ to $1.5L_{pp}$), $4L_{pp}$ ($-2L_{pp}$ to $2L_{pp}$) and $1.04L_{pp}$ ($-L_{pp}$ to $0.04L_{pp}$) respectively in the $x$, $y$, and $z$ directions. For these cases without drift angles ($\beta = 0$), a half domain $y$ ($0{,}2L_{pp}$) is used with a symmetric center plane boundary condition. Velocity inlet boundary condition is set at the inlet, zero gradient boundary condition is set at the outlet, bottom, and atmosphere.

Two meshing strategies were adopted and compared. First, an open source automatic hexahedra and split-hexahedra tool (SnappyHexMesh) was used to generate an approximate body-conforming mesh on the Wigley hull geometry. Additional layers of hexahedral cells were then aligned to the hull surface and automatic refinement guaranteed approximately uniform mesh distribution, in all three-dimensions, near the solid walls. This meshing strategy is often used for more complex hull simulations [2, 36, 76, 89, 91] and is denoted as the ***castellated mesh*** herein. The second approach consists of a user generated, true body-conforming mesh using commercial software (Pointwise). We assure a smooth expansion of the mesh from the solid walls and an optimal orientation of the near wall mesh relative to the flow. This second approach is termed ***body-conforming mesh***. The general view of the half computational domain and the mesh on the hull surface is shown in Figure 4.6 (body-conforming mesh) and Figure 4.7 (castellated mesh). For each of these meshing strategies, three different mesh refinement levels are investigated (named as coarse, medium, and fine, respectively); the first layer distance is set to satisfy $30 < y^+ < 150$ in all cases; the details of the mesh are found in Table 5.3. With the similar first layer distance from the wall, the number of grid points in castellated mesh are far more than body-conforming mesh, which is due to the Cartesian alignment of the castellated mesh.

### 4.4.2 DTC and DTMB 5415

Due to the complex geometry of DTC and DTMB 5415, two types of castellated meshes are generated by snappyHexMesh utility in OpenFOAM in the current simulation: a coarse mesh and a high-resolution mesh. The coarse mesh for DTMB 5415 is shown in Figure 4.8 and the high-resolution mesh is shown Figure 4.9. For the coarse mesh, we selected the computational domain to be $L_x$,$L_y$,$L_z$=42, 38, 20 for a unitary ship size. For the high resolution mesh, we selected a smaller computational domain with $L_x$,$L_y$,$L_z$=40, 18, 9, which also satisfies the ITTC guidelines [84]. In the first step, a multi-grading functionality

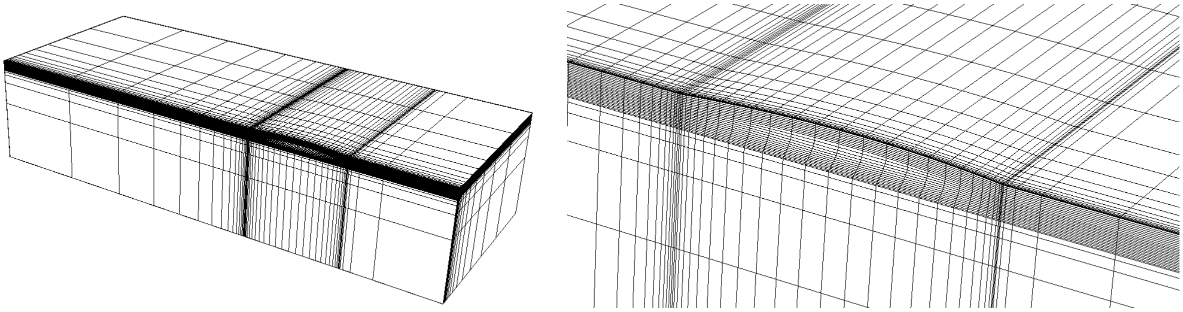


Figure 4.6: Body-conforming mesh of the Wigley hull (half domain). Left: general view, right: mesh detail near the hull.

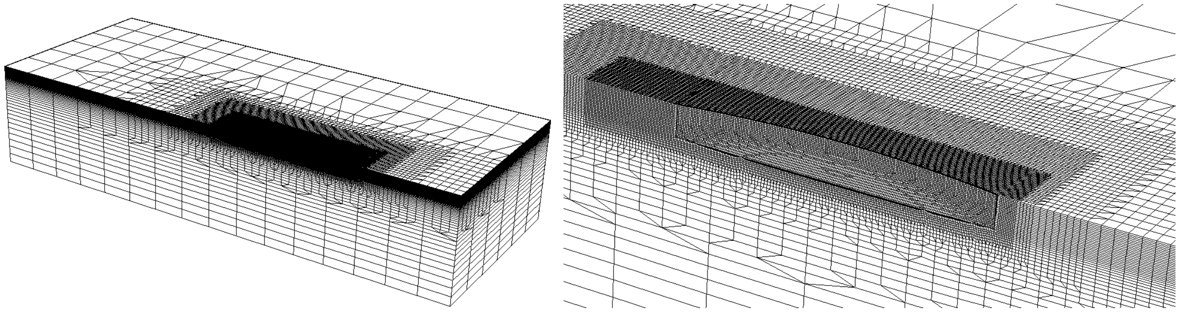
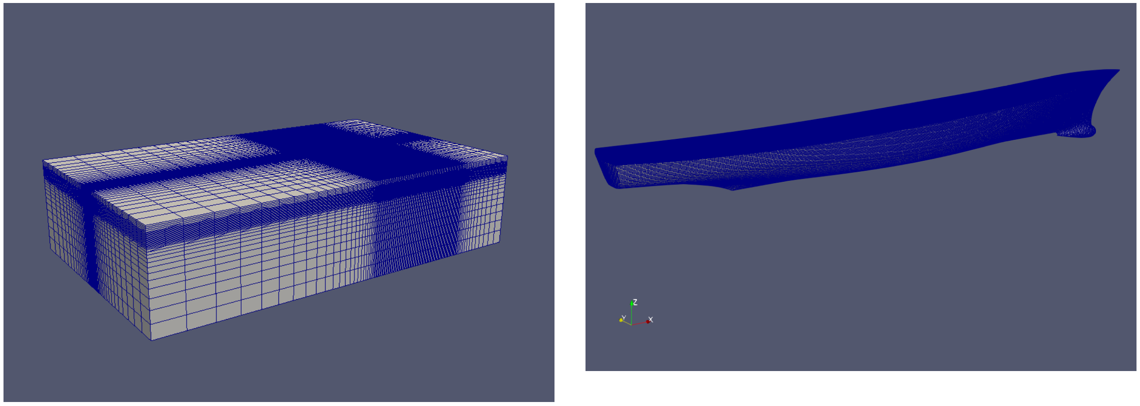


Figure 4.7: Castellated mesh for Wigley hull (half domain). Left: general view, right: mesh detail near the hull.

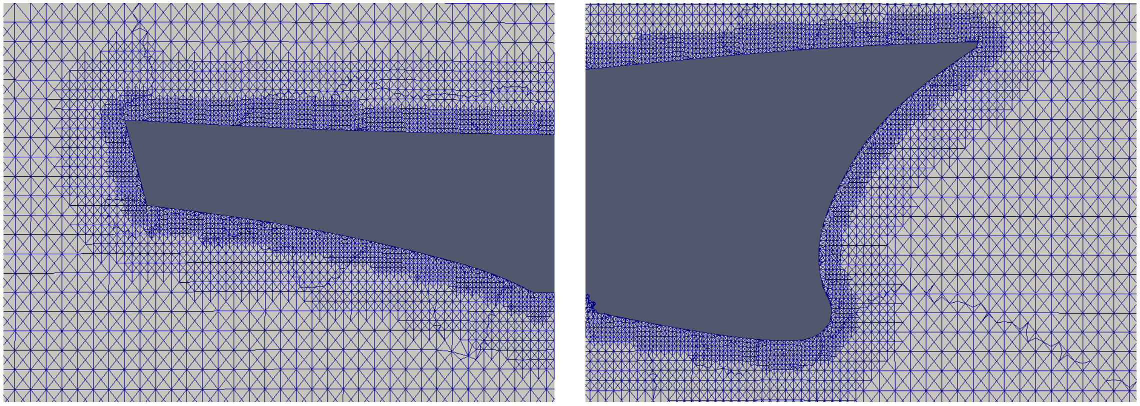
is used in blockMesh to generate background mesh: where a gradual refinement in three direction is achieved, with uniform refined three-dimensional mesh near the hull. In the second step, refineRegion and refineSurface functionality in the snappyHexMesh tool are successively used, each refinement level splits the cell size in half over the defined air-water interface region and the region near the hull surface. Several Python and C codes were developed to scale, to rotate the ship geometry, and to generate the blockMeshDict file for non-uniform background mesh, as shown in the Appendix B.

## 4.5 Computational Resources

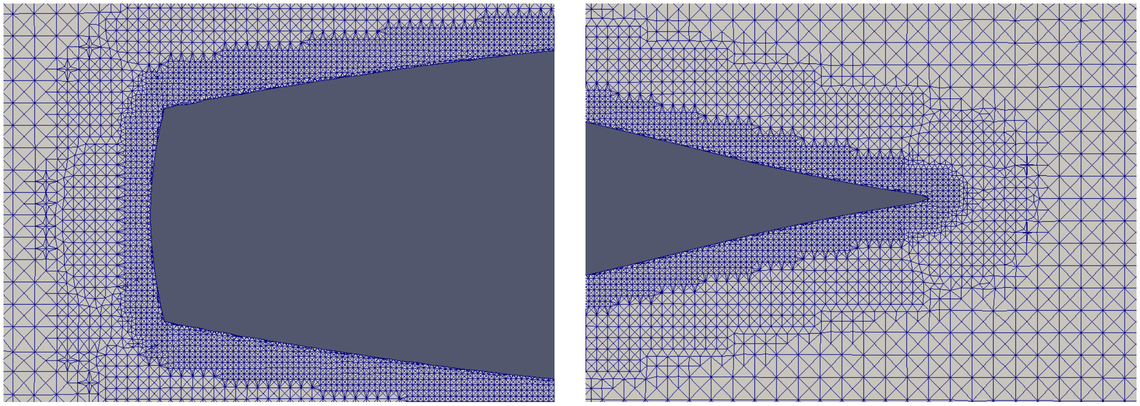
The simulations were performed on the Niagara supercomputer at the SciNet HPC Consortium with 40 cores, a clock speed of 2.4 GHz and 202 GB of Random Access Memory (RAM). The wall clock time was approximately 3000s per static simulation. The static simulations were run up to 4000s with a time step of 1s for attaining converged results. In the dynamic cases, the wall clock time was approximately 20 hours per simulation, with an average time step of 0.001s to satisfy $CFL \leq 0.5$ and simulation time of 20s to obtain stable results.


(a) whole domain

(b) DTMB 5415

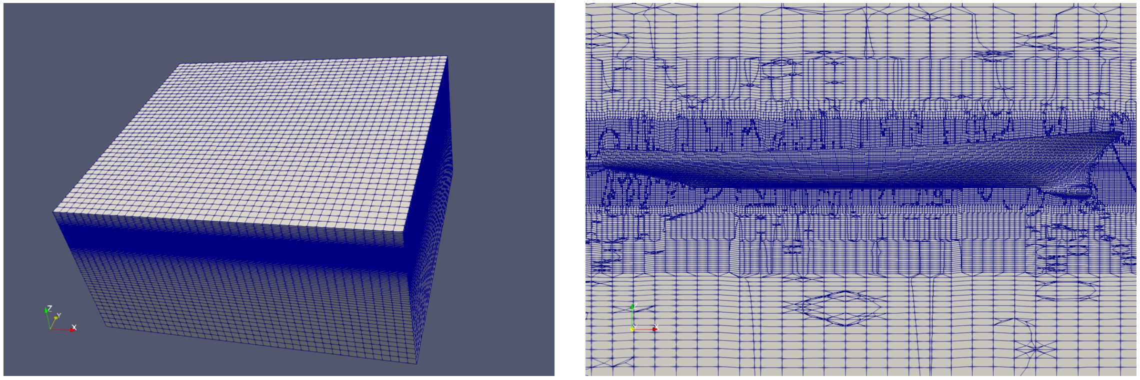

(c) center plane: enlarged rear

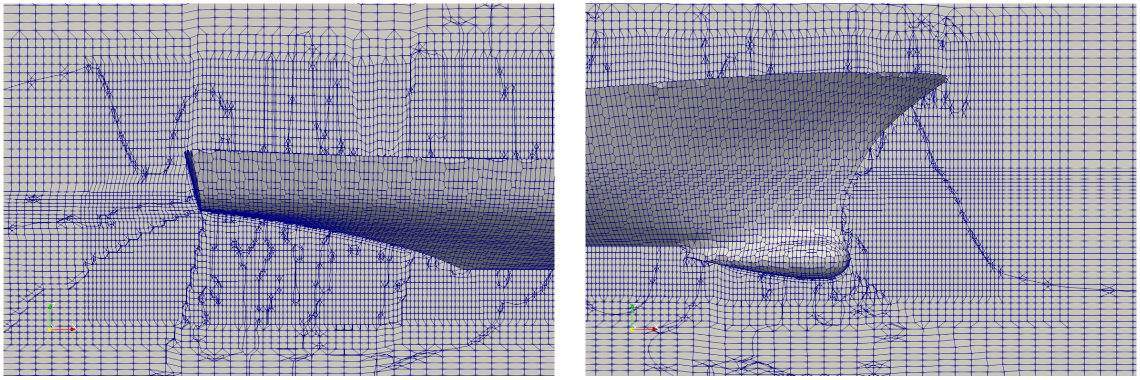
(d) center plane: enlarged front


(e) top view: enlarged rear

(f) top view: enlarged front

Figure 4.8: Mesh for DTMB 5415 by refineSurface ( number of grid points: 2.37 million).


(a) whole domain (b) DTMB 5415


(c) center plane: enlarged rear (d) center plane: enlarged front

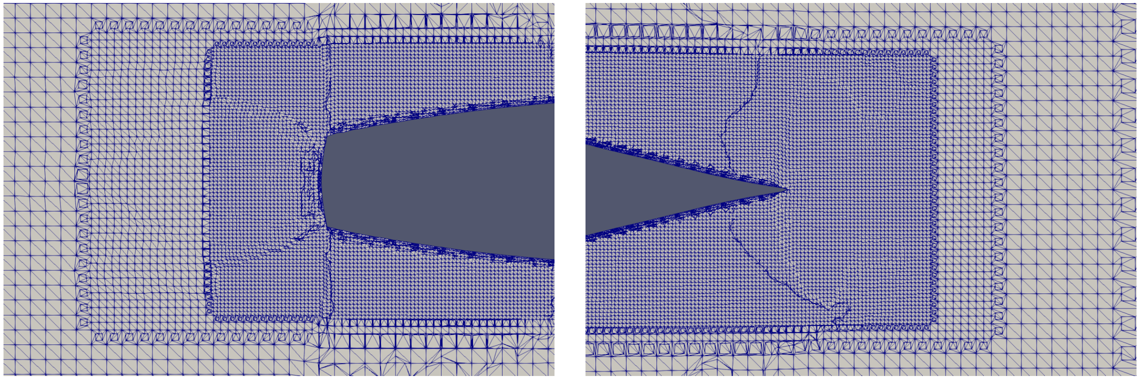

(e) top view: enlarged rear (f) top view: enlarged front

Figure 4.9: Mesh for DTMB 5415 by topoSet (number of grid points: 2.99 million).

# Chapter 5

# Results and Discussion

This paper focuses on the static and dynamic PMM simulation. All test cases are summarized in Table 5.1. A mesh-independence study was performed with five different resolution for both the DTC and DTMB 5415 at drift 0°. Different Froude number simulations were conducted at drift angle of 0° for validation of both the DTC and DTMB 5415. Further steady and unsteady PMM cases at $Fr = 0.28$ (for DTMB 5415) were investigated. For unsteady PMM simulations, the motions were set by the PMM model tests in the experiment [95]. From the resulting numerical simulations, drag, lateral force, and yaw moments were calculated and compared against experimental data. The animations of the mesh movement and hydrodynamic pressure distribution in dynamic PMM simulation is presented in `https://www.youtube.com/channel/UCKjJx2ACHCDmguEGGf6eEKQ`.

## 5.1 Wigley Hull

The simulations are performed at static drift conditions with a drift angle of $\beta \in [0°, 10°]$, which correspond to the experimentally studied test cases by Kajitani *et al.* [45] and by Kashiwagi [46]. A summary of all test cases is shown in Table 5.2.

### 5.1.1 Static Drift

To start with, a mesh independence test at a zero drift angle ($\beta = 0°$) was performed with the castellated mesh and body-conforming mesh and the drag coefficient is investigated. As shown in Table 5.3, the errors on the drag coefficients are less than 5% in both of the

| **Cases** | **Conditions** | **Grid** | **Solver** |
|---|---|---|---|
| Static drift: Wigley hull | $\beta \in [0°, 10°]$ | refineSurface, body-conforming | interFoam |
| Static drift: DTC | $\beta = 0°, Fr \in [0.174, 0.218]$ | topoSet | interFoam |
| Static drift: INSEAN | $\beta = 0°, Fr \in [0.05, 0.45]$ | topoSet | interFoam |
| Static drift: MARIN | $\beta \in [0°, 20°], Fr = 0.28$ | refineSurface | interFoam |
| Pure sway: MARIN | $\beta = 0°, Fr = 0.28$ | refineSurface | interDyMFoam |
| Pure yaw: MARIN | $\beta = 0°, Fr = 0.28$ | topoSet | interDyMFoam |
| Yaw and drift: MARIN | $\beta = 10°, Fr = 0.28$ | topoSet | interDyMFoam |

Table 5.1: Test cases of ships for the static and dynamic PMM simulations.

| $\beta$ [deg] | **Re** [$\times 10^6$] | $Fr$ | **Experimental data** |
|---|---|---|---|
| 0 | 2.86 | 0.25 | Kajitani *et al.* [45] |
| 10 | 2.86 | 0.267 | Heredero *et al.* [79] |
| 2-10 | 1.6 | 0.181 | Kashiwagi *et al.* [46] |

Table 5.2: Case description of the Wigley hull experiments

cases at medium mesh and reach 2.0% and 3.1% in the cases with fine castellated mesh and body-conforming mesh, respectively. Besides the force coefficients, the wave profile around the hull is presented in Figure 5.1. Compared with experimental data [45], the peak and trough of the wave around the hull can be predicted accurately in medium and fine both types of meshes. Non-physical oscillation is observed with castellated mesh, which could be due to the poorer near wall mesh alignment. The body conforming mesh (fine) is used for investigation of the Froude number effect on drag coefficient in the zero drift cases, and the side force and yaw moment coefficients in the non-zero drift cases.

Zero drift cases at different Froude numbers (from 0.268 to 0.408) are further simulated in the body-conforming mesh (refine). As presented in Figure 5.2, the current simulation can capture the trend of resistance with different Froude numbers well. Further, the relative error of resistance coefficients between the current simulation and experimental data [45] at different Froude numbers are shown in Table 5.4, where the relative error reaches a maximum value of 5.89% at $Fr = 0.267$ and a minimum value of 0.76% at $Fr = 0.408$, which could be due to the difficulty to capture the shorter length of wave at small Froude number, as observed in the simulation by Ma [66]. Besides the resistance coefficients, the wave profiles on the hull surface are further compared with experimental data, as shown in Figure 5.3. The peak of wave profiles are captured well, except a slight under prediction in the case of $Fr = 0.408$.

Side force coefficient $Y'$ and yaw moment coefficient $N'$ are further investigated in the non-zero drift cases ($\beta$ from $2^\circ$ to $10^\circ$) using the body-conforming mesh (fine), as presented in Figure 5.4. The simulations can capture the experimental trend at different drift angles: both $Y'$ and $N'$ increase as the drift angle increases. An increasing error with experimental data is observed for $N'$ at larger drift angles. The relative error of $Y'$ and $N'$ at different drift angles are shown in Table 5.5. The error of $Y'$ is around 10% at different drift angles, while the error of $N'$ increases from 20.7% at drift $\beta = 2^\circ$ and reaches 33.0% at drift $\beta = 8^\circ$, which could be due to the flow separation at the larger drift angles. Due to the limited experimental data available, wave profiles at windward and leeward around the hull at drift $\beta = 10^\circ$ and $Fr = 0.267$ are shown in Figure 5.5. Although the current simulation can capture the wave profile, the profile shows a greater discrepancy with experiments compared to the zero drift angle case, especially for the wave profile at the leeward. On the windward side, the peak of the waves is well captured, while the amplitudes of the first and second troughs are under predicted; on the windward side, the amplitude of the wave is under-predicted all along the hull. The current deviation at large drift angle indicates inaccuracy of modelling the turbulent multiphase flow, which should be investigated further.

| **Description** | **castellated mesh** | | | **body-conforming mesh** | | |
|---|---|---|---|---|---|---|
| Mesh resolution | coarse | medium | fine | coarse | medium | fine |
| Number of cells [$\times 10^5$] | 7.14 | 17.92 | 39.75 | 0.69 | 1.96 | 5.50 |
| Drag coefficient $X'$ [$\times 10^{-3}$] | 5.36 | 4.80 | 4.48 | 4.25 | 4.36 | 4.43 |
| Error $X'$ [%] | 17.3 | 5.0 | 2.0 | 7.0 | 4.6 | 3.1 |

Table 5.3: Mesh independence test (half domain) at drift $\beta = 0°$, $Fr = 0.25$. The error is evaluated relative to the experimental data: Kajitani *et al.* 1983 [45] which found a drag coefficient of $X' = 4.57 \times 10^{-3}$.

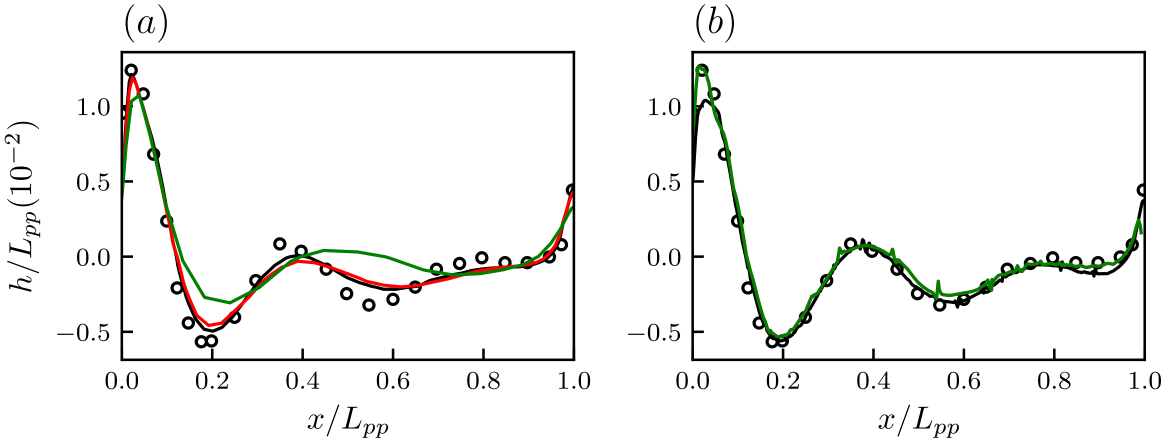


Figure 5.1: The wave profile along the hull at a drift angle of $\beta = 0$ degree ($Fr = 0.25$). ○: Experimental data [45], black line: fine mesh, red line: medium mesh, green line: coarse mesh. (a) body-conforming mesh, (b) castellated mesh.

| $Fr$ | 0.25 | 0.267 | 0.289 | 0.316 | 0.354 | 0.408 |
|---|---|---|---|---|---|---|
| $Re$ [$\times 10^5$] | 7.744 | 8.303 | 8.602 | 8.689 | 10.009 | 12.690 |
| **relative error** [%] | 3.10 | 5.89 | 5.16 | 4.95 | 0.81 | 0.76 |

Table 5.4: Relative error (%) of resistance coefficient for Wigley hull at different Froude numbers, drift 0°.

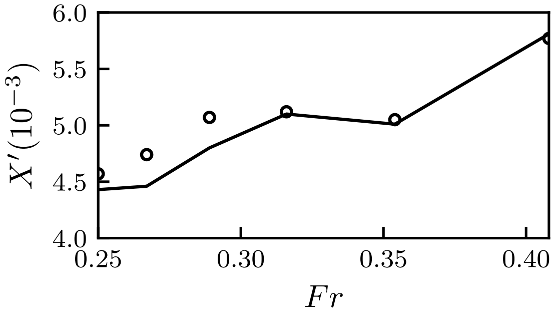


Figure 5.2: Drag coefficient $X'$ at different Froude numbers, drift 0 degree. $\circ$ Experimental data [45], $-$ OpenFOAM result.

| $\beta$ | 2° | 4° | 6° | 8° | 10° |
|---|---|---|---|---|---|
| $Y'$ | 11.8 | 14.4 | 11.8 | 9.7 | 10.2 |
| $N'$ | 20.7 | 25.8 | 29.6 | 33.0 | 29.3 |

Table 5.5: Relative error (%) of force and moment coefficients at different drift angles.

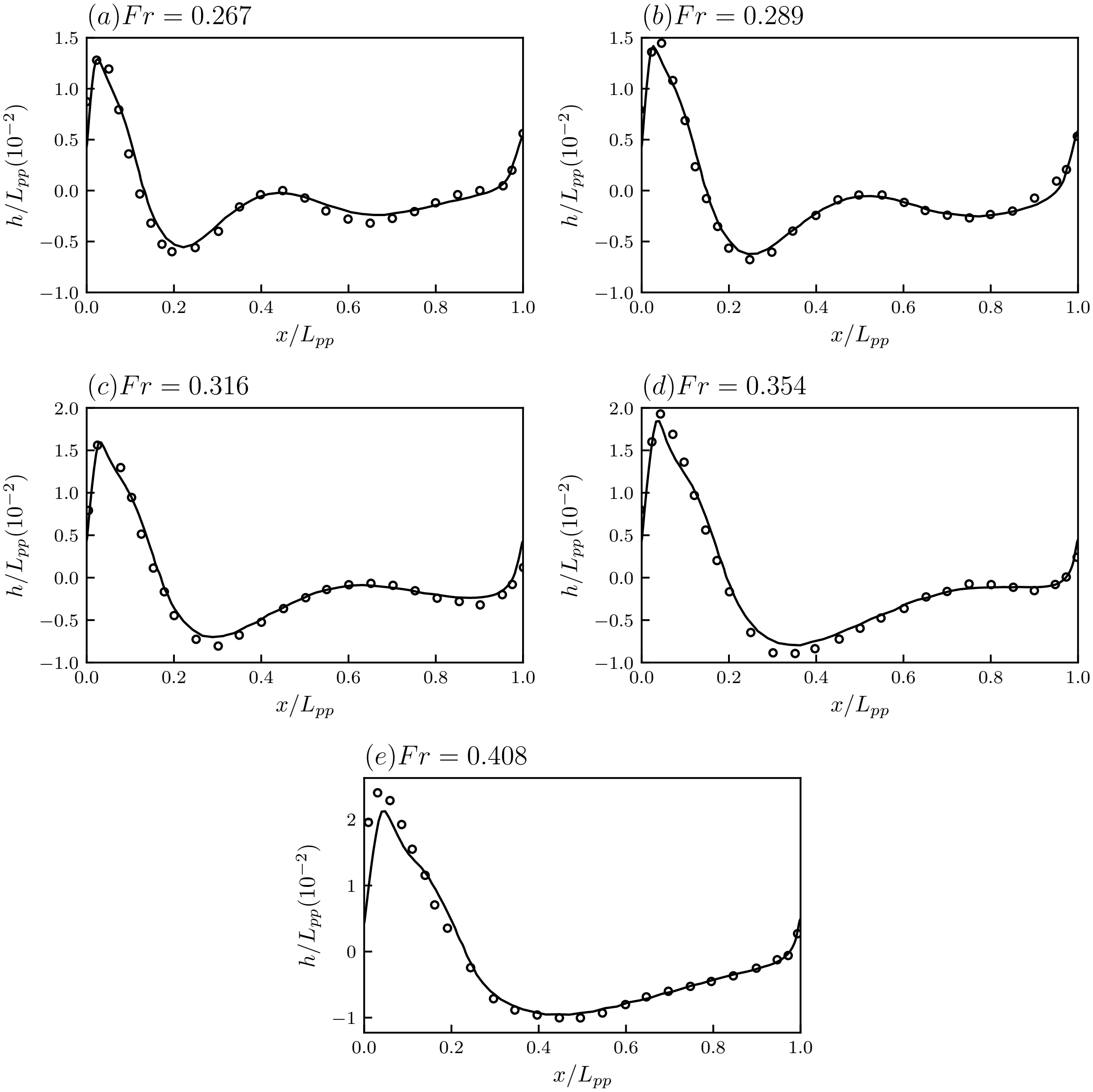


Figure 5.3: Wave profile along the hull at different Froude numbers, drift 0 degree. ○ Experimental data [45], – OpenFOAM result.

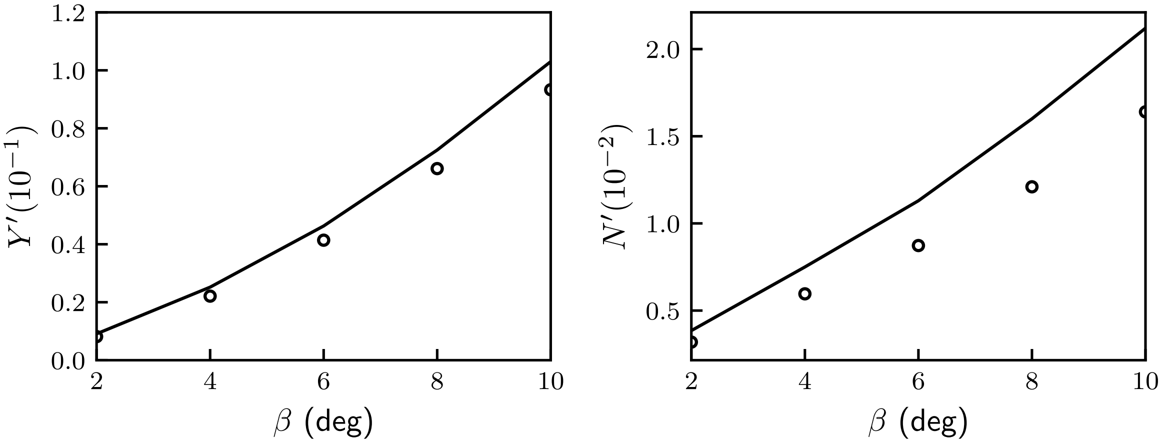


Figure 5.4: Force and moment coefficients at different drift angles, $Fr = 0.161$(body-conforming mesh). ○ Experimental data [46], – OpenFOAM result.

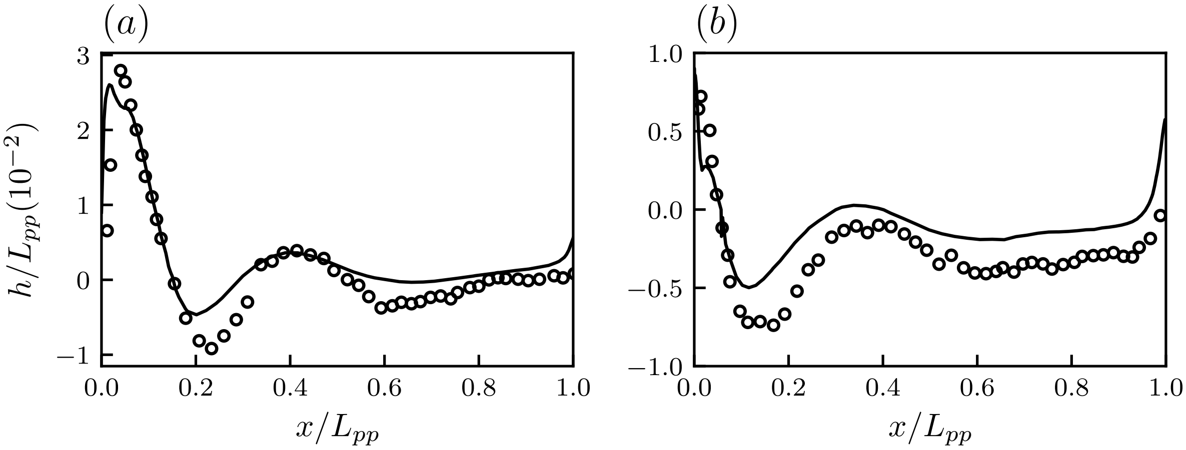


Figure 5.5: Wave profiles at drift $\beta = 10°$, $Fr = 0.267$. ○ Experimental data [79], – OpenFOAM result (a)windward; (b)leedward.

| $Fr$ | 0.174 | 0.183 | 0.192 | 0.200 | 0.210 | 0.218 |
|---|---|---|---|---|---|---|
| $Re$ $[\times 10^6]$ | 7.319 | 7.681 | 8.054 | 8.415 | 8.783 | 9.145 |

Table 5.6: Test cases for DTC hull at drift $\beta = 0°$.

## 5.2 DTC Hull

### 5.2.1 Static Drift

All test cases for DTC hull is presented in Table 5.6. The mesh independence study was performed at $Fr = 0.218$. The different meshes for the DTC case are based on the refinement times near the hull. For different mesh resolutions, the convergence of the relative error of resistance coefficient, compared with model test data [19] are shown in Table 5.7. $m2$ means twice the grid refinement near the hull and $m6$ six times refinement near the hull. The non-dimensional distance of the first layer from the hull, $y^+$, equals 140 for the medium mesh which is slightly superior to the suggested value of 100 for the use of wall models. We consider this value to be acceptable and consistent with the current wall function used in conjunction with the turbulence models used in OpenFOAM. The medium mesh was used for further investigation of the effect of Froude numbers on resistance coefficients.

With varying Froude numbers between 0.174 to 0.218 on the medium mesh, the relative error on resistance coefficient is tabulated in Table 5.8. The visual comparison of the resistance force with the experimental data [19] over a range of Froude numbers is shown in Figure 5.6. The results provide evidence of a good agreement for the predictive numerical simulations with the experiments for the DTC case. Hydrodynamic pressure distribution on the interface at different Froude numbers are presented in Figure 5.7. As the Froude number increases, the difference between peak and crest of becomes more obvious, which means the wave effect is stronger at higher Froude numbers.

## 5.3 DTMB 5415

### 5.3.1 Static Drift

For the DTMB 5415 case, mesh-independence study was performed at $Fr = 0.28$, $Re = 1.26 \times 10^7$ and drift $\beta = 0°$ which allows us to compare with the experimental data taken

| **Mesh** | $m2$ | $m3$ | $m4$ | $m5$ | $m6$ |
|---|---|---|---|---|---|
| **number of grid points** [$\times 10^6$] | 0.668 | 0.913 | 1.260 | 1.718 | 2.679 |
| **relative error** [%] | 54 | -18 | 0.87 | -3.3 | -0.56 |

Table 5.7: Grid independence study on the resistance coefficient of DTC ($Fr = 0.218$, $\beta = 0°$). The error is evaluated relative to the experimental data: el Moctar *et al.* 2012 [19] which found a drag coefficient of $X' = 3.67 \times 10^{-3}$.

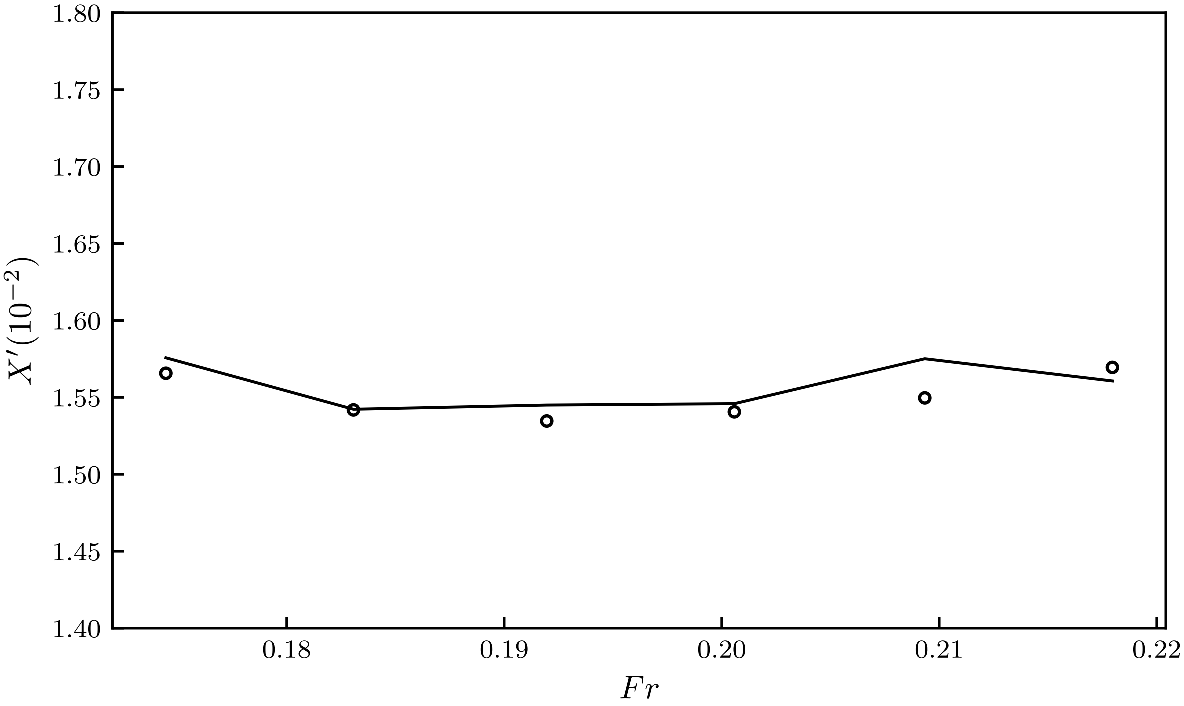


Figure 5.6: Resistance coefficients of DTC at drift 0°. ○ Experimental data [19], − OpenFOAM result.

| $Fr$ | 0.174 | 0.183 | 0.192 | 0.200 | 0.210 | 0.218 |
|---|---|---|---|---|---|---|
| **Error** ($X'$) [%] | 0.64 | 0.03 | 0.67 | 0.34 | 1.6 | 0.56 |

Table 5.8: Resistance coefficient for DTC at different Froude numbers, drift 0°.

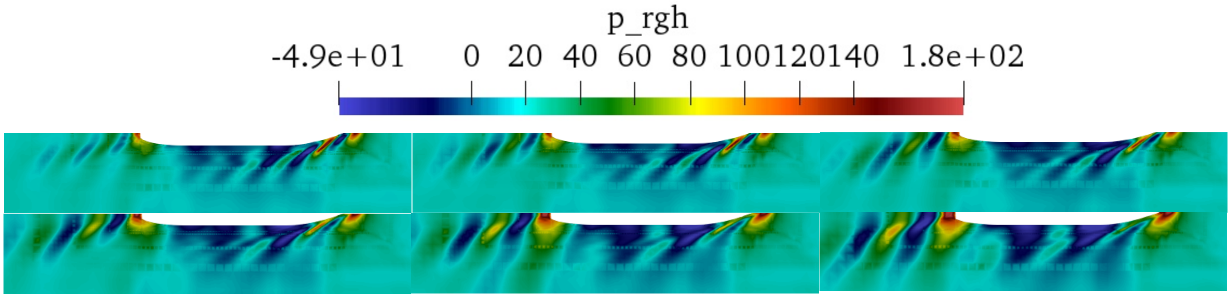


Figure 5.7: Hydrodynamic pressure distribution on the free surface of DTC at Fr= 0.174, 0.183, 0.192, 0.200, 0.210, 0.218 (from upper left to bottom right).

| mesh | $m2$ | $m3$ | $m4$ | $m5$ | $m6$ | refineSurface |
|---|---|---|---|---|---|---|
| number of cells [$10^6$] | 0.669 | 0.916 | 1.273 | 1.778 | 2.995 | 2.37 |
| relative error [%] | 49.6 | 14.0 | 28.3 | 1.21 | 4.13 | 3.04 |

Table 5.9: Resistance coefficients of DTMB 5415 ($Fr = 0.28$, $\beta = 0°$) for different mesh resolutions.

from [77]. Different refinement levels from $m2$ to $m6$ near the hull were used as a criteria for the different mesh resolutions. The non-dimensional distance of first grid layer from the hull $y^+$ equals 140 for the finest mesh: $m6$. The number of grid points and relative error on the resistance coefficient in different mesh resolutions are shown in Table 5.9. As the error is less than 4.13%, we deem the mesh resolution $m6$ refined by topoSet method and the mesh refined by refineSurface method to be acceptable for the parametric study.

The comparison of the free wave height around the ship hull (at $Fr = 0.28$) with published data from the open literature is shown in Figure 5.8; Figure 5.9 shows the experimental results [77]. The OpenFOAM results are consistent with both published experimental and numerical simulations, which supports the claim that the VOF method can capture the wave near the hull. Despite the overall good agreement, we do note that there is dissipation of the wave height due to the mesh stretching at the interface–this is especially true away from the ship hull. The air-water interface on the ship hull is shown in Figure 5.10. The trend of the wave height computed from CFD is consistent with the experimental results, but again we do note a slightly lower peak (0.17) in the simulation compared to experiment (0.18). Over the entire ship length, the wave height is slightly over-predicted. The cause of difference stems from limited mesh resolution in this region of the ship hull. Wave profile away from the ship at $y/L_{pp} = 0.082, 0.172, 0.301$ is also presented and compared with experimental data. At $y/L_{pp} = 0.082$, strong non-physical oscillations near the hull are observed in the simulation, which may due to the stretch ratio of mesh refinement near the hull. At $y/L_{pp} = 0.172$ and $y/L_{pp} = 0.301$, the wave trend is consistent with the experiment. The phase error of the wave at $y/L_{pp} = 0.301$ may result from the dispersion of the numerical scheme.

The wake of the paddle disk is important for the propeller design as it influences the propulsion efficiency. The correct prediction of the wake is critical for predictive ship manoeuvring simulations. Figure 5.11 presents the streamwise velocity contour among the current simulation results, experimental data [77] , and numerical simulation results using OpenFOAM as described by Shen [88]. The current simulation results are consistent with both experimental and numerical results from the open literature. Due to the presence of spanwise vortices, the contours of the streamwise velocity show humps away from the hull. The current simulation can capture key characteristics of the wake near the stern, which is useful for propeller design.

After the mesh independence test, the fine mesh $m6$ is used to investigate the effect of Froude number (from 0.05 to 0.45). The Froude number and the Reynolds number change simultaneously with the variation of velocity, as shown in Table 5.10. The resistance coefficient $X'$ is plotted against Froude number, as presented in Figure 5.12. As the increasing Froude number, the resistance coefficient $X'$ decreases in the range $Fr \in [0.05, 1]$; $X'$ increases slowly in the range $Fr \in [0.1, 0.35]$, and then $X'$ increases significantly in the range $Fr \in [0.35, 0.45]$. The comparison with experimentally determined values reveals an overall good agreement of the predictive simulations–especially to capture the trends. Table 5.11 shows the relative difference between model test [77] and numerical simulation results. The relative error is smaller than 6.55% at the Froude number ranging from 0.05 to 0.4 and it reaches 12.5% at $Fr = 0.45$. The larger error at $Fr = 0.45$ could be due to the combined effect of Froude number and Reynolds number. The high Froude number means high Reynolds number in the current case, in which the wave run up height and the velocity gradient near the wall increase. So a better mesh resolution in required in both of the vertical direction and the wall normal direction, compared with the mesh in the case $Fr = 0.28$. An error of 4.84% is observed at $Fr = 0.05$, which could be due to the incapability to capture the shorter wave. The shorter wave becomes significant at smaller Froude number, based on the ship Kelvin wake theory [75].

Pure drift simulations (recall Figure 1.1) at different drift angle $\beta$ from 0 to 20 degrees were conducted at a Froude number of 0.28. The prediction of forces and moment coefficients are presented in Figure 5.13 and show the overall good trend with increasing drift angles, compared with experimental data [95]. The relative error of the forces and drift moments are tabulated in Table 5.12. The current simulation results match well with the

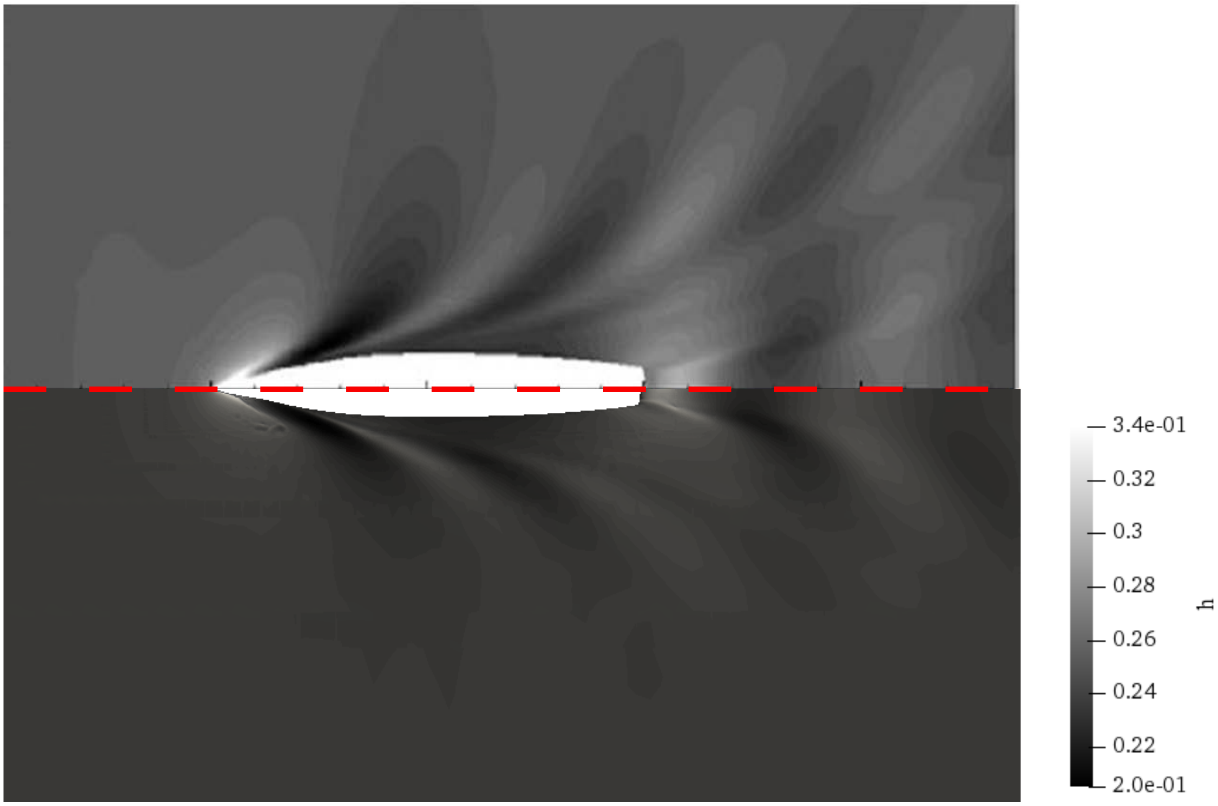


Figure 5.8: Comparison of the free surface wave height of the numerical simulations by Ma [66] (top half) and the OpenFOAM results of the present simulation (bottom half) in the DTMB 5415 case at $\beta = 0°$, $Fr = 0.28$.

| $Fr$ | 0.05 | 0.1 | 0.15 | 0.2 | 0.25 | 0.3 | 0.35 | 0.4 | 0.45 |
|---|---|---|---|---|---|---|---|---|---|
| $Re$ ($\times 10^6$) | 2.13 | 4.27 | 6.4 | 8.55 | 10.7 | 12.8 | 14.9 | 17.1 | 19.2 |

Table 5.10: Test cases for DTMB 5415, at drift $\beta = 0°$.

| $Fr$ | 0.05 | 0.1 | 0.15 | 0.2 | 0.25 | 0.3 | 0.35 | 0.4 | 0.45 |
|---|---|---|---|---|---|---|---|---|---|
| relative error [%] | 4.84 | 2.87 | 4.36 | 4.63 | 6.55 | 3.24 | 0.782 | -5.68 | -12.1 |

Table 5.11: Relative error of resistance coefficient for DTMB 5415, at drift 0°.

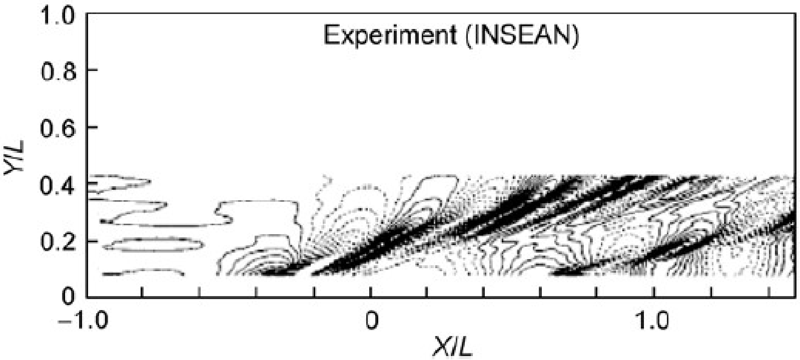


Figure 5.9: Free surface wave height from the experimental results by Olivieri [77] in the DTMB 5415 case.

| Drift angle(°) | 0 | 2 | 6 | 9 | 10 | 11 | 12 | 16 | 20 |
|---|---|---|---|---|---|---|---|---|---|
| $X'$ | 3 | 4.7 | 0.73 | 3.6 | 6.0 | 4.5 | 4.1 | 16 | 25 |
| $Y'$ | - | 3.2 | 8.8 | 13 | 6.1 | 5.7 | 0.81 | 2.5 | 4.0 |
| $N'$ | - | 59 | 19 | 15 | 7.2 | 2.5 | 5.3 | 3.2 | 10 |

Table 5.12: Relative error (%) of forces and moment coefficients at different drift angles (DTMB 5415).

experimental data. The only exception is the resistance coefficient $X'$ at drift angles of 16° and 20°. For $X'$, the relative error are smaller than 6% at drift angle range from $\beta = 0°$ to $\beta = 12°$; while it reaches 16% at $\beta = 16°$ and 25% at $\beta = 20°$.

Furthermore, the hydrodynamic pressure distribution on the free surface at different drift angles is shown in Figure 5.14. The pattern of hydrodynamic pressure shows stronger asymmetry due to the larger drift angles. The pressure shows an expected strong asymmetry and there are strong pressure fluctuations in the starboard side as the drift angle increases. The pressure pattern is not well-resolved at the high drift angles as the same mesh is used in all drift simulations. This may lead to some errors in the force and moment coefficients. In addition, these error may be compounded by the under-estimation of wave height on the ship surface.

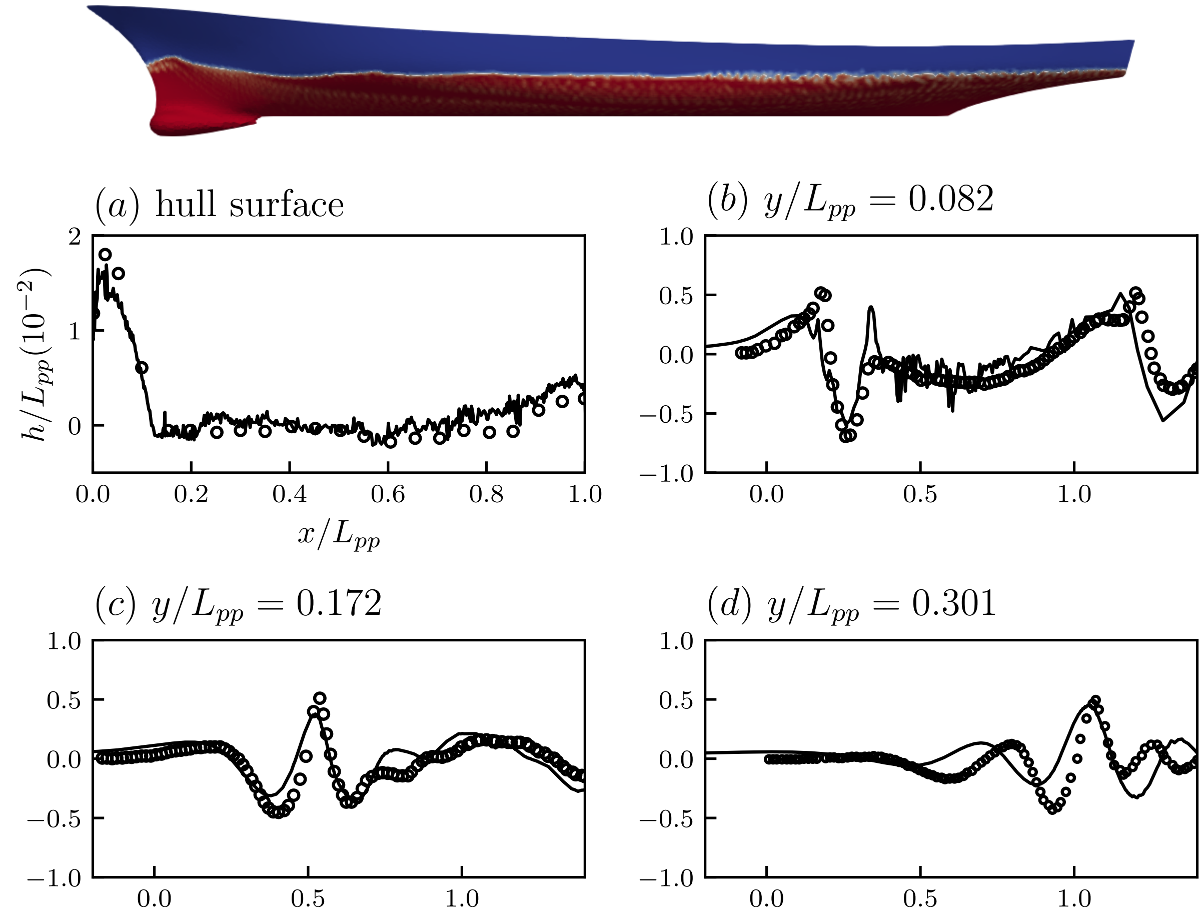


Figure 5.10: Wave on DTMB 5415 ship surface, Fr=0.28, drift=0°. ○ Experimental data [77]; − OpenFOAM result.

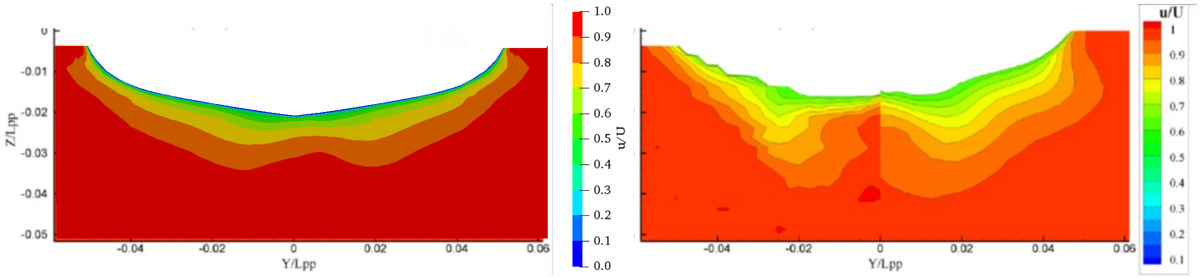


Figure 5.11: Streamwise velocity contour of the paddle disk $x/L_{pp} = 0.935$ (DTMB 5415). Left: current simulation; Right: experimental result (left half) and numerical simulation (right half) [88].

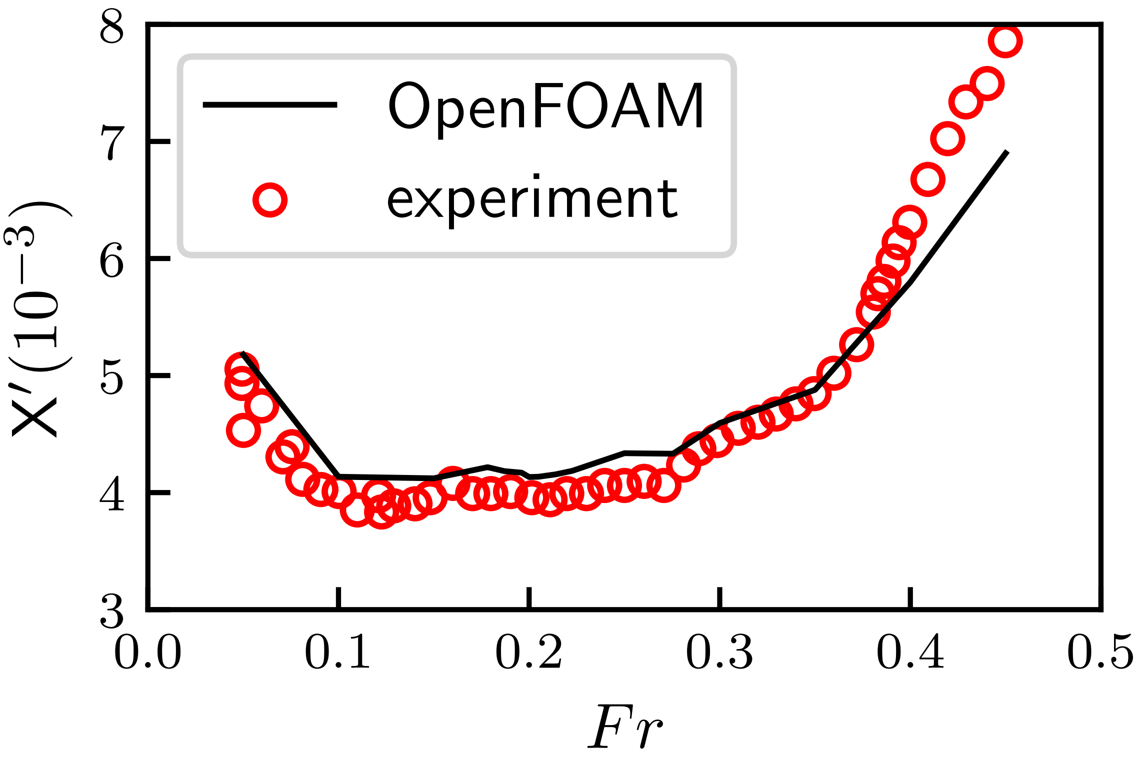


Figure 5.12: Resistance coefficients of DTMB 5415 at drift 0°. Experimental data from Olivieri [77].

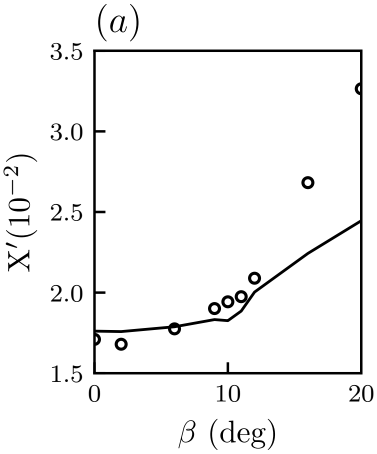


Figure 5.13: Forces and moment coefficient for static drift at different drift angles (DTMB 5415). ○ Experimental data [95]; – OpenFOAM result.

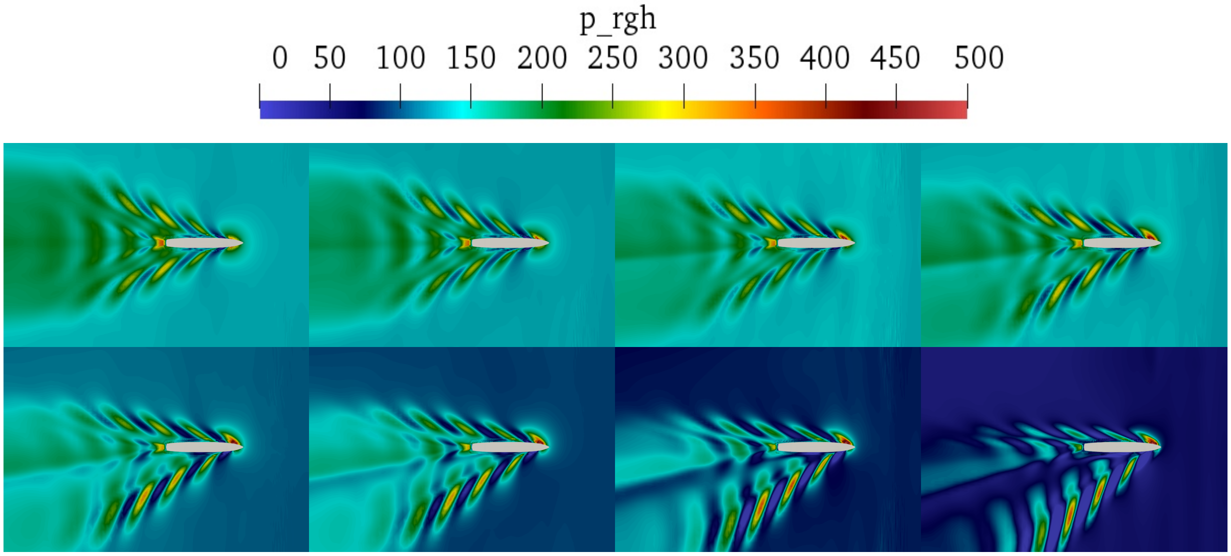


Figure 5.14: Hydrodynamic pressure distribution on the free surface at static drift 0°, 2°, 6°, 9°, 11°, 12°, 16°, 20° (from upper left to bottom right, DTMB 5415)

### 5.3.2 Pure Sway

The pure sway motion is defined as $y = y_{max} sin(\omega t)$ where $\omega = 0.733 rad/s$, and $y_{max} = 0.4158m$. The time evolution of the force and yaw moment coefficients from the current numerical simulations and the experimental data by Simonsen [95] are shown in Figure 5.15. In Figure 5.15, although there are some deviations in the resistance coefficient $X'$ and lateral force coefficient $Y'$ between the CFD and experimental data [95], the CFD results follow the trend of the experimental data. This is especially true for the yaw moment coefficient. A lack of refinement near the wake due to the sway motion may lead to the error. Figure 5.16 shows the pressure distribution at the waterline plane at different ship positions at sway path. The sway motion induced different pressure distributions on the hull for port and starboard side.

### 5.3.3 Pure Yaw

During the pure yaw motion, the ship axis is tangent to its path. Yaw motion is applied directly to the ship hull, while restricting all other motion. The yaw motion was performed by coupling of oscillatingDisplacement and angularOscillatingDisplacement motion solver, using fvMotionSolver in the dynamicMeshdict [73]. The motion is defined as:

$$\begin{aligned} y = y_{max} sin(\omega t) \qquad & \text{where } \omega = 0.733 \text{ rad/s}, y_{max} = 0.4158 \text{ m} \\ \theta = \theta_{max} sin(\omega t) \qquad & \text{where } \theta_{max} = 0.4179467 \text{ rad} \end{aligned} \tag{5.1}$$

In the pure yaw case, there is a phase shift between the yaw angle and lateral motion. The lateral motion is maximum when the yaw angle equals zero and vice versa. Lateral motion is set to 0 in the first quarter of period to ensure gradual positioning of the ship.

The time evolution of forces and yaw moment coefficient are shown in Figure 5.17. The simulation results were compared with experimental data by Simonsen [95]. The time history of forces and yaw moment coefficients in CFD capture the trend of experimental data; however the amplitude of lateral force in CFD is under predicted, since we chose a coarse mesh in the pure yaw simulation for numerical stability. Similar as the result in pure sway case, the deviation of yaw moment coefficient compared to experimental data are much smaller than the resistance and the lateral force coefficient, which could be due to the fact that yaw moment coefficient is integral parameters. Hydrodynamic pressure distribution on the interface is shown in Figure 5.18. The asymmetric hydrodynamic pressure distribution could result in the non-zero lateral force.

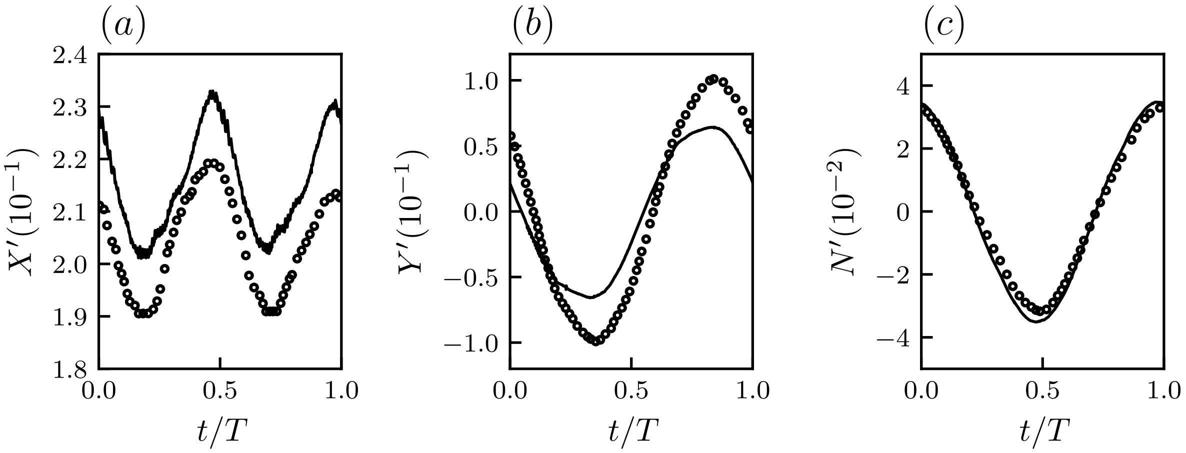
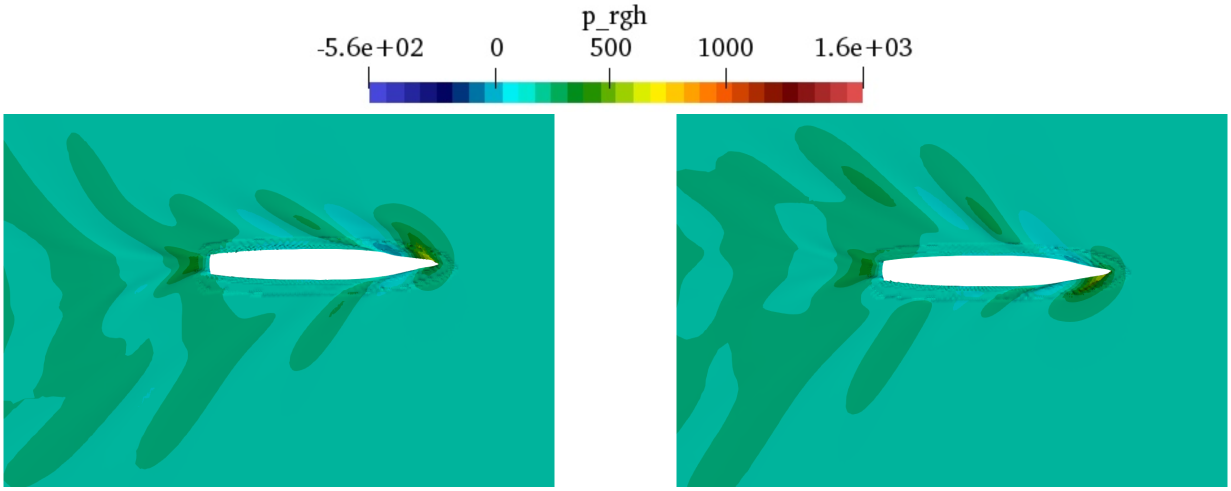


Figure 5.15: Forces and moment coefficients: pure sway. ○ Experimental data [95]; − OpenFOAM result.

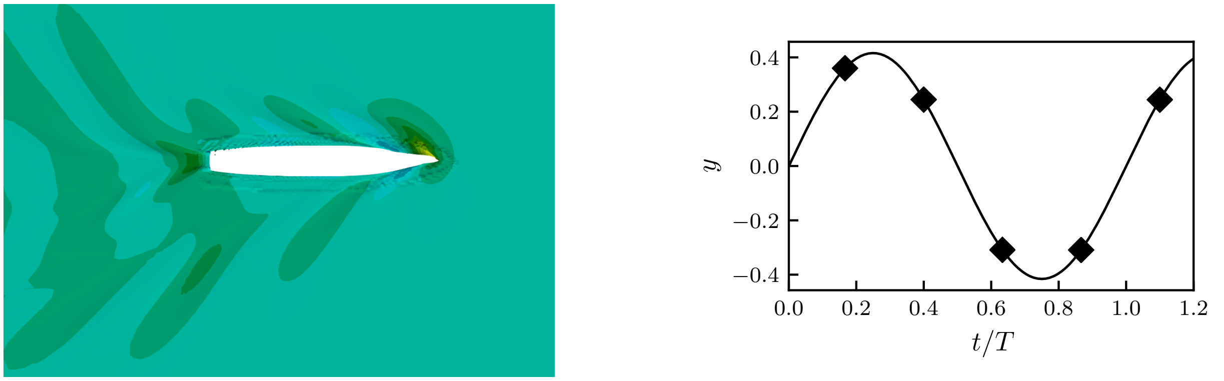
### 5.3.4 Yaw and Drift

In the case of a yaw rate of 0.3 and a drift of 10 degrees, the motion of the ship is set as the Equation 5.1, the same as the pure yaw motion in the previous section. The drift angle is prescribed by the direction of inlet flow, which is the same as the method used in the pure drift simulation, as shown in Figure 4.3b. The mesh in the yaw and drift case is the same as that in the pure yaw simulation. The time history of forces and moment coefficient, and the hydrodynamic pressure on the interface are presented in Figure 5.19, and Figure 5.20. A large deviation of forces and moment coefficient is observed between the CFD and experimental data, as presented in Figure 5.19. The deviation may result from the following: (1) the coarse mesh resolution close to the hull, near the wave surface, and/or in the region of the wake; (2) the $k-\omega$ SST model with the wall function, which may lead to a large error when the flow separation occurs at a large drift angle; (3) the non-orthogonality of the deformed grid due to the large magnitude motion of the ship; (4)the hull model without bilge keels in the simulation, compared with the model in the experiment [95].



(a) $t/T = 0.17$ (b) $t/T = 0.40$

(c) $t/T = 0.63$ (d) $t/T = 0.87$



(e) $t/T = 1.10$ (f) sway motion

Figure 5.16: Hydrodynamic pressure distribution on the free surface during pure sway motion (bottom figure presents different ship positions).

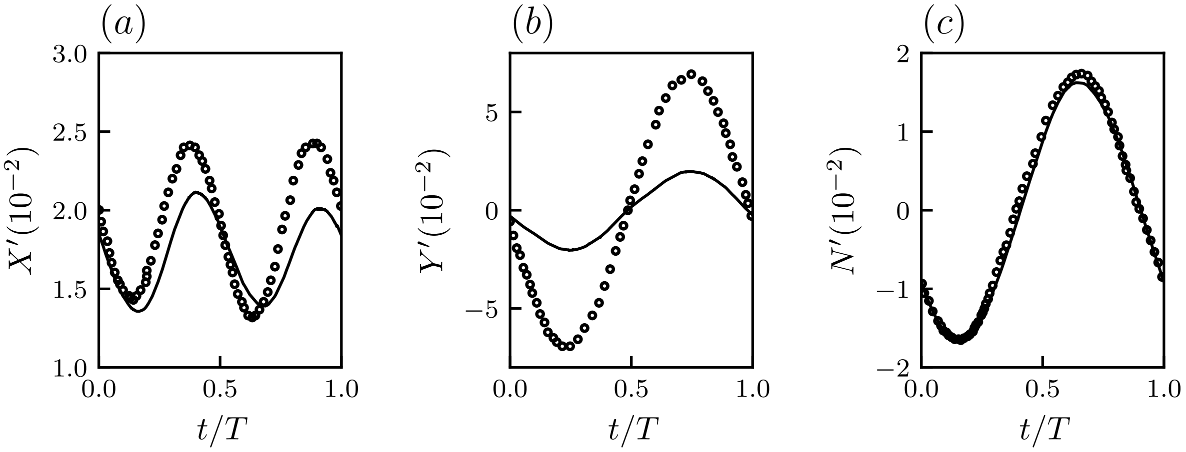
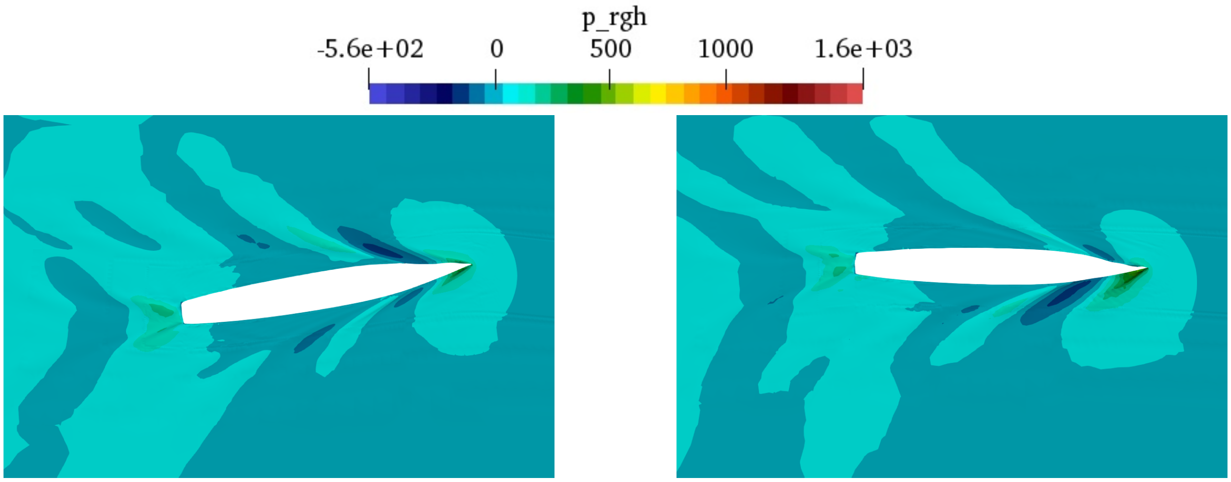
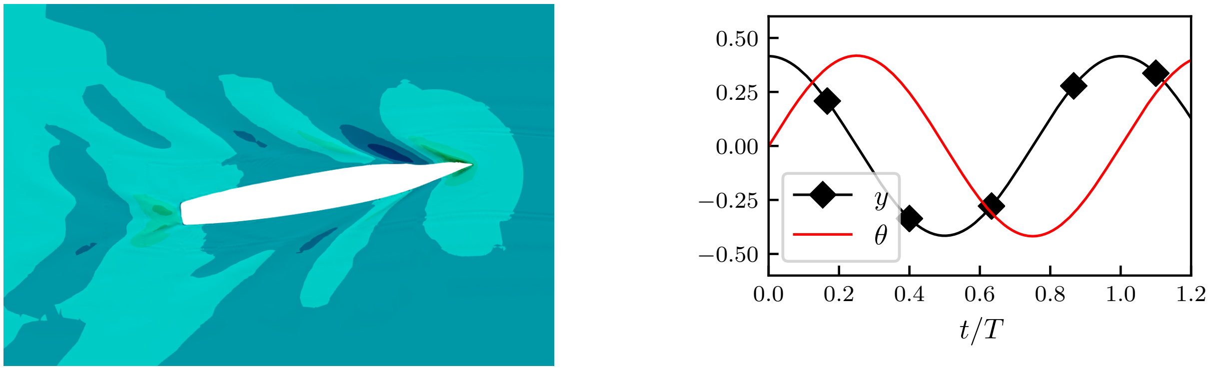


Figure 5.17: Forces and moment coefficient: pure yaw. ○ Experimental data [95]; – OpenFOAM result.



(a) $t/T = 0.17$ (b) $t/T = 0.40$

(c) $t/T = 0.63$ (d) $t/T = 0.87$



(e) $t/T = 1.10$ (f) yaw motion

Figure 5.18: Hydrodynamic pressure distribution on the free surface during pure yaw motion (bottom figure presents different ship positions).

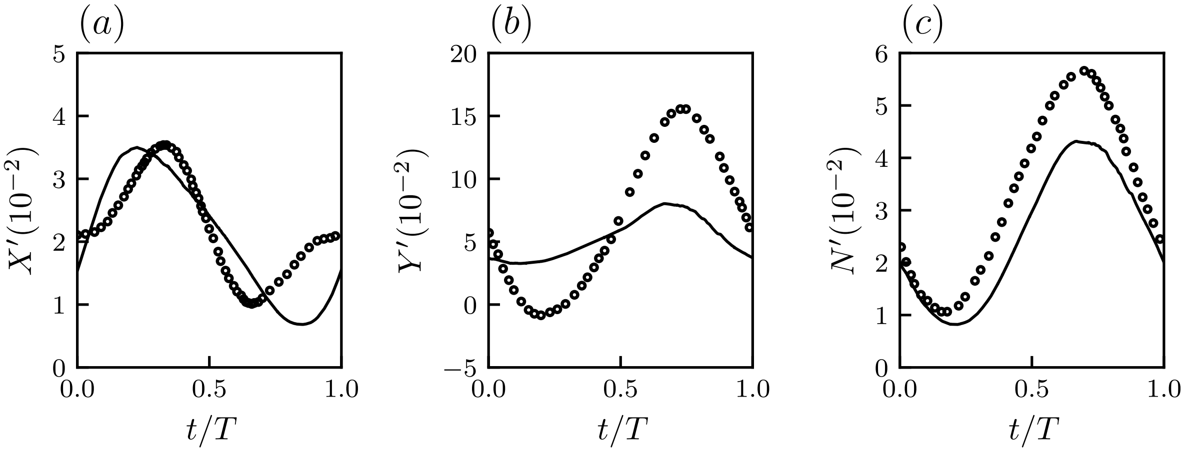
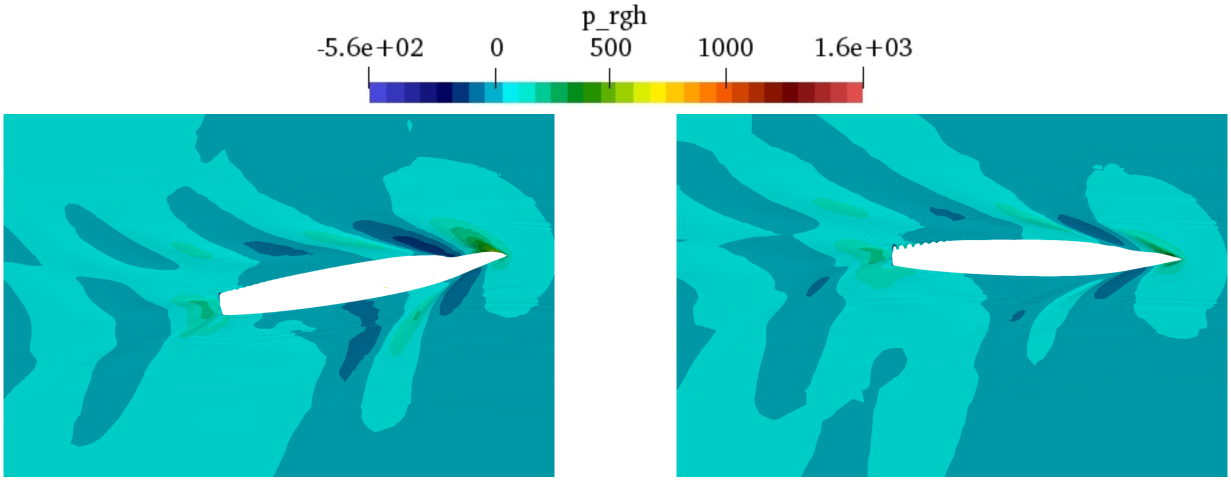
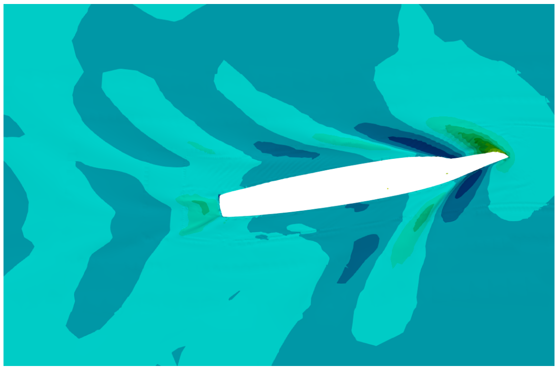


Figure 5.19: Forces and moment coefficient: yaw and drift 10°. ○ Experimental data [95]; − OpenFOAM result.



(a) $t/T = 0.17$ (b) $t/T = 0.40$

(c) $t/T = 0.63$ (d) $t/T = 0.87$



(e) $t/T = 1.10$

Figure 5.20: Hydrodynamic pressure distribution on the free surface (yaw and drift 10°).

# Chapter 6

# Conclusions and Future Work

## 6.1 Conclusions

This thesis investigates numerical error for the predictive static and dynamic PMM simulations of the force and moment coefficients during ship manoeuvring using the fully open-source software package, OpenFOAM. All the pre- and post-processing steps are conducted using open-source tools, except the body-conforming mesh generation tool Pointwise for the Wigley hull case. Three well-established ship geometries are used to benchmark the numerical simulations: the Wigley hull, the DTC hull, and the DTMB 5415. The error of forces and moments are investigated in the numerical simulations by comparison with experimental data. The results present good agreement with experimental data, except three specific cases:

1. at a high Froude number;
2. at a large drift angle;
3. for the specific combined yaw and drift case.

This thesis concludes that the current numerical methods are able to predict the forces and moment in ships manoeuvring problems with good accuracy.

The numerical simulations are done using the InterFoam and InterDyMFoam package within OpenFOAM, which is a turbulent RANS solver using a VOF method to account for the two-phases of the simulation in a continuous regime (without needing to track the discrete air-water interface). The InterFoam solver is used for static drift simulations and the InterDyMFoam solver is used for dynamic PMM (pure sway, pure yaw, combined yaw and drift) simulations. The body-conforming mesh is generated using Pointwise. The castellated mesh is generated via successive mesh refinement and adaptation. Dynamic mesh technique is used for dynamic PMM simulation, where the mesh deformed during the ship movement. Despite the lower accuracy of this grid generation method, the results show an overall good agreement with experimental data and published numerical results; all results are presented in a ship-fixed coordinate system.

For the steady drift simulation, a time-converged result is obtained to compare with experimental data. From the case of Wigley hull, the castellated mesh generated using snappyHexMesh is capable of predicting the drag and wave profile along the ship hull in multiphase turbulent flow. However, given the Cartesian nature and the local refinement criteria of the castellated mesh, a greater number of grid points are needed in order to obtain a sufficiently fine near wall resolution. The body-conforming mesh generated using Pointwise required fewer overall grid points for the same near wall resolution. Non-physical oscillations were observed with the castellated mesh (snappyHexMesh) due to the poorer near wall mesh alignment. The prediction of wave profile along the ship hull is better predicted at zero drift angle; errors increase with the drift angle. This is due to a poorer prediction of separated flows. The relative error of the resistance coefficient is smaller than 6% at various Froude numbers (from 0.25 to 0.408); larger error is observed at non-zero drift cases (from $2^\circ$ to $10^\circ$), especially for the yaw moment. At a drift angle $\beta = 0^\circ$, at various Froude numbers (from 0.174 to 0.218), the relative error of the resistance coefficient for DTC is smaller than 1.6%. For DTMB 5415, the relative error of drag coefficient increases at large Froude numbers and reaches 12.1% at $Fr = 0.45$, due to the increasing non-dimensional first layer distance to the hull $y^+$–resulting in a larger near wall resolution error. The insufficient mesh resolution induces larger error for the cases of static drift $16^\circ$ and $20^\circ$, especially for the lateral force coefficient.

The unsteady PMM simulations of DTMB 5415 are validated by comparing the converged time-periodic result in the simulation with experiment. Despite much effort, the prediction of pure sway and pure yaw compared less favorably than the steady simulation results. Further investigation is needed to improve the prediction for the yaw and drift cases. Different from the experiment by Simonsen [95], there is no bilge keel with the bare hull in the current simulation model, which may lead to the error. The combined effects of the low accuracy mesh generation method and the dynamic deformation mesh results in large deviations of the unsteady PMM simulations compared to the experimental data. The overset method is ideal for dynamic PMM simulation; however, currently it is only implemented in commercial software and in some in-house codes by some groups. To implement the overset method in the current OpenFOAM version 17.12 should be considered as the next step of work.

## 6.2 Future Work

To improve the accuracy of force and moment predictions for static and dynamic PMM cases, both high quality mesh generation and advanced turbulence modelling are required. High-quality body-conforming mesh using commercial software such as Pointwise should be considered for static cases. Unsteady DES solver is needed for large drift cases. For dynamic PMM cases, overset method is suggested.

# References

[1] ANSYS. https://www.ansys.com/. Accessed: 2019-10-23.

[2] A. Bakica, I. Gatin, V. Vukčević, H. Jasak, and N. Vladimir. Accurate assessment of ship-propulsion characteristics using CFD. *Ocean Engineering*, 175:149–162, 2019.

[3] Inlet outlet boundary condition in OpenFOAM. https://www.cfdsupport.com/OpenFOAM-Training-by-CFD-Support/node114.html. Accessed: 2019-3-30.

[4] R. F. Beck, J. N. Newman, and E. O. Tuck. Hydrodynamic forces on ships in dredged channels. *Journal of Ship Research*, 19(03):166–171, 1974.

[5] L. Benedetti, B. Bouscasse, R. Broglia, L. Fabbri, F. L. Gala, and C. Lugni. PMM model test with DDG51 including uncertainty assessment. *FORCE Technology, Department of Maritime Industry, Lyngby*, 2007.

[6] M. A. Benitz, D. W. Carlson, B. S. Aghazadeh, Y. M. Sadeghi, M. A. Lackner, and D. P. Schmidt. CFD simulations and experimental measurements of flow past free-surface piercing, finite length cylinders with varying aspect ratios. *Computers & Fluids*, 136:247–259, 2016.

[7] S. Bhushan, H. S. Yoon, F. Stern, E. Guilmineau, M. Visonneau, S. Toxopeus, C. Simonsen, S. Aram, S. E. Kim, and G. Grigoropoulos. CFD validation for surface combatant 5415 straight ahead and static drift 20 degree conditions. *IIHR—Hydroscience & Engineering, Iowa City, IA, Report*, 2015.

[8] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *Journal of Computational Physics*, 100(2):335–354, 1992.

[9] P. M. Carrica, F. Ismail, M. Hyman, S. Bhushan, and F. Stern. Turn and zigzag maneuvers of a surface combatant using a URANS approach with dynamic overset grids. *Journal of Marine Science and technology*, 18(2):166–181, 2013.

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

[11] IIHR CFD Code: CFDShip-Iowa. `https://www.iihr.uiowa.edu/shiphydro/cfd-code/`. Accessed: 2019-10-23.

[12] C. Y. Chen and F. Noblesse. Comparison between theoretical predictions of wave resistance and experimental data for the wigley hull. *Journal of Ship Research*, 27(04):215–226, 1983.

[13] C. Y. Chen and F. Noblesse. Preliminary numerical study of a new slender-ship theory of wave resistance. *Journal of Ship Research*, 27(03):172–186, 1983.

[14] C. Ciortan, J. Wanderley, and C. G. Soares. Turbulent free-surface flow around a wigley hull using the slightly compressible flow formulation. *Ocean Engineering*, 34(10):1383–1392, 2007.

[15] J. L. D. Dantas and E. A. De Barros. Numerical analysis of control surface effects on AUV manoeuvrability. *Applied Ocean Research*, 42:168–181, 2013.

[16] S.C.R. Dennis and G. Z. Chang. Numerical solutions for steady flow past a circular cylinder at reynolds numbers up to 100. *Journal of Fluid Mechanics*, 42(3):471–489, 1970.

[17] S. S. Deshpande, L. Anumolu, and M. F. Trujillo. Evaluating the performance of the two-phase flow solver interFoam. *Computational Science & Discovery*, 5(1):014016, 2012.

[18] US navy combatant, DTMB 5415. `http://www.simman2008.dk/5415/combatant.html`.

[19] O. el Moctar, V. Shigunov, and T. Zorn. Duisburg test case: Post-panamax container ship for benchmarking. *Ship Technology Research*, 59(3):50–64, 2012.

[20] O. el Moctar, F. Sprenger, T. Schellin, and A. Papanikolaou. Numerical and experimental investigations of ship maneuvers in waves. In *ASME 2016 35th International Conference on Ocean, Offshore and Arctic Engineering, Busan, South Korea*, 2016.

[21] N. E. Eryuzlu. Underkeel requirements for large vessels in shallow waterways. *28th International Navigation Congeress, PIANC, Paper*, 2:17–25, 1994.

[22] N. E. Eryuzlu and R. Hausser. Experimental investigation into some aspects of large vessel navigation in restricted waterways. *Proc. Symp. Aspects of Navigability of Constraint Waterways Including Harbour Entrances*, 2(9), 1978.

[23] B. Fabritius and G. Tabor. Improving the quality of finite volume meshes through genetic optimisation. *Engineering with Computers*, 32(3):425–440, 2016.

[24] P. Ghadimi, M. Ghandali, and M. R. A. Balootaki. Numerical simulation of regular waves rup-up over slopping beach by OpenFOAM. *Civil Engineering and Urban Planning: An International Journal (CiVEJ)*, 1(1), 2014.

[25] V. R. Gopala and B. G. M. Van Wachem. Volume of fluid methods for immiscible-fluid and free-surface flows. *Chemical Engineering Journal*, 141(1-3):204–221, 2008.

[26] J. Gorski, S. Kim, S. Aram, B. Rhee, and H. Shan. Development of a CFD framework for prognoses of resistance, powering, maneuvering, and seakeeping of surface ships. In *30th Symposium of Naval Hydrodynamics*, 2014.

[27] L. Gui, J. Longo, B. Metcalf, J. Shao, and F. Stern. Forces, moment, and wave pattern for surface combatant in regular head waves. Part I. Measurement systems and uncertainty assessment. *Experiments in Fluids*, 31(6):674–680, 2001.

[28] L. Gui, J. Longo, B. Metcalf, J. Shao, and F. Stern. Forces, moment and wave pattern for surface combatant in regular head waves. Part II. Measurement results and discussions. *Experiments in Fluids*, 32(1):27–36, 2002.

[29] User Guide. STAR-CCM+ Version (7.04), 2012.

[30] A. Hajivand and S. H. Mousavizadegan. Virtual simulation of maneuvering captive tests for a surface vessel. *International Journal of Naval Architecture and Ocean Engineering*, 7(5):848–872, 2015.

[31] R. He, Z. Z. Zhang, X. Z. Wang, and D. K. Feng. Numerical simulation of the ship bottom interaction of DTC container carrier for different keel clearance in pure sway motion. In *4th MASHCON-International Conference on Ship Manoeuvring in Shallow and Confined Water with Special Focus on Ship Bottom Interaction*, pages 65–72, 2016.

[32] J. A. Heyns and O. F. Oxtoby. Modelling surface tension dominated multiphase flows using the VOF approach. In *6th European Conference on Computational Fluid Dynamics*, pages 7082–7090, 2014.

[33] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981.

[34] IIHR: Ship Hydrodynamics- a revolution in naval ship design. `https://www.iihr.uiowa.edu/shiphydro/`. Accessed: 2019-10-23.

[35] M. Irvine, J. Longo, and F. Stern. Pitch and heave tests and uncertainty assessment for a surface combatant in regular head waves. *Journal of Ship Research*, 52(2):146–163, 2008.

[36] H. Islam and C.G. Soares. Estimation of hydrodynamic derivatives of a container ship using PMM simulation in OpenFOAM. *Ocean Engineering*, 164:414–425, 2018.

[37] International Towing Tank Conference. `https://ittc.info/`. Accessed: 2019-3-30.

[38] N. G. Jacobsen, D. R. Fuhrman, and J. Fredsøe. A wave generation toolbox for the open-source CFD library: OpenFoam®. *International Journal for Numerical Methods in Fluids*, 70(9):1073–1088, 2012.

[39] OpenCFD release: OpenFOAM v1712. `https://www.openfoam.com/releases/openfoam-v1712/`. Accessed: 2018-12-31.

[40] H. Jasak. *Error analysis and estimation for the finite volume method with applications to fluid flows.* PhD thesis, Imperial College London (University of London), 1996.

[41] H. Jasak. Dynamic mesh handling in OpenFOAM. In *47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 341, 2009.

[42] H. Jasak, A. Jemcov, and Z. Tukovic. OpenFOAM: A C++ library for complex physics simulations. In *International Workshop on Coupled Methods in Numerical Dynamics*, 2007.

[43] H. Jasak and Željko Tuković. Dynamic mesh handling in OpenFOAM applied to fluid-structure interaction simulations. In *Proceedings of the V European Conference on Computational Fluid Dynamics ECCOMAS CFD 2010*, 2010.

[44] D. A. Jones and D. B. Clarke. FLUENT code simulation of flow around a naval hull: the DTMB 5415. Technical report, Defence Science and Technology Organization Victoria (Austrilia) Maritime Platforms DIV, 2010.

[45] H. Kajitani, H. Miyata, M. Ikehata, H. Tanaka, H. Adachi, M. Namimatsu, and S. Ogiwara. The summary of the cooperative experiment on Wigley parabolic model in japan. Technical report, Tokyo University, 1983.

[46] M. Kashiwagi. *The study of fluid dynamical forces of maneuvering acting on a ship in broaching motion.* PhD thesis, University of Osaka, Japan, 1984.

[47] T. Kawamura, S. Mayer, A. Garapon, and L. Sørensen. Large eddy simulation of a flow past a free surface piercing circular cylinder. *Journal of Fluids Engineering*, 124(1):91–101, 2002.

[48] T. Kawamura and H. Miyata. Simulation of nonlinear ship flows by density-function method. *Journal of The Society of Naval Architects of Japan*, 1994(176):1–10, 1994.

[49] H. C. Kim, H. Akimoto, and H. Islam. Estimation of the hydrodynamic derivatives by RANS simulation of planar motion mechanism test. *Ocean Engineering*, 108:129–139, 2015.

[50] S. E. Kim, B. J. Rhee, and R. W. Miller. Anatomy of turbulent flow around DARPA SUBOFF body in a turning maneuver using high-fidelity RANS computations. *International Shipbuilding Progress*, 60(1-4):207–231, 2013.

[51] S. E. Kim and S. Schroeder. Numerical study of thrust-breakdown due to cavitation on a hydrofoil, a propeller, and a water jet. In *Proceedings of the 28th Symposium on Naval Hydrodynamics, Pasadena, CA*, 2010.

[52] O. K. Kinaci, A. Kukner, and S. Bal. On propeller performance of DTC post-panamax container ship. *International Journal of Ocean System Engineering*, 3(2):77–89, 2013.

[53] R. J. Lambert. Development of a numerical wave tank using OpenFOAM. Master's thesis, Universidade De Coimbra, 2012.

[54] B. E. Larsen, D. R. Fuhrman, and J. Roenby. Performance of interFoam on the simulation of progressive waves. *arXiv preprint arXiv:1804.01158*, 2018.

[55] L. Larsson, F. Stern, and V. Bertram. Benchmarking of computational fluid dynamics for ship flows: the Gothenburg 2000 workshop. *Journal of Ship Research*, 47(1):63–81, 2003.

[56] L. Larsson, F. Stern, and M. Visonneau. *Numerical ship hydrodynamics: an assessment of the Gothenburg 2010 workshop.* Springer, 2013.

[57] S. K. Lee, J. M. You, H. H. Lee, T. Lim, S. T. Park, J. H. Seo, S. H. Rhee, and K. P. Rhee. Experimental study on the six degree-of-freedom motions of a damaged ship floating in regular waves. *IEEE Journal of Oceanic Engineering*, 41(1):40–49, 2016.

[58] S. K. Lee, J. M. You, H. H Lee, T. Lim, S. H. Rhee, and K. P. Rhee. Preliminary tests of a damaged ship for CFD validation. *International Journal of Naval Architecture and Ocean Engineering*, 4(2):172–181, 2012.

[59] J. Ley, S. Sigmund, and O. el Moctar. Numerical prediction of the added resistance of ships in waves. In *ASME 2014 33rd International Conference on Ocean, Offshore and Arctic Engineering*. American Society of Mechanical Engineers, 2014.

[60] M. N. Linnick and H. F. Fasel. A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. *Journal of Computational Physics*, 204(1):157–192, 2005.

[61] H. Liu, N. Ma, and X. C. Gu. Numerical simulation of pmm tests for a ship in close proximity to sidewall and maneuvering stability analysis. *China Ocean Engineering*, 30(6):884–897, 2016.

[62] J. L. Liu, R. Hekkenberg, E. Rotteveel, and H. Hopman. Literature review on evaluation and prediction methods of inland vessel manoeuvrability. *Ocean Engineering*, 106:458–471, 2015.

[63] S. K. Liu, A. Papanikolaou, and G. Zaraphonitis. Time domain simulation of nonlinear ship motions using an impulse response function method. In *ICMT conference, July*, 2014.

[64] J. Longo and F. Stern. Resistance, sinkage and trim, wave profile, and nominal wake tests and uncertainty assessment for DTMB model 5512. In *Proc 25th American Towing Tank Conference, Iowa*, 1998.

[65] J. Longo and F. Stern. Uncertainty assessment for towing tank tests with example for surface combatant DTMB model 5415. *Journal of Ship Research*, 49(1):55–68, 2005.

[66] J. Ma and D. C. Wan. A numerical study of resistance and viscous flow around typical benchmark surface ship hull. *Scientia Sinica Physica, Mechanica & Astronomica*, 41(2):178, 2011.

[67] K. V. Mani, A. Cervone, and J. P. Hickey. Turbulence modeling of cavitating flows in liquid rocket turbopumps. *Journal of Fluids Engineering*, 139(1):011301, 2017.

[68] I. Martić, N. Degiuli, A. Farkas, and J. Bašić. Mesh sensitivity analysis for numerical simulation of a damaged ship model. In *The Twenty-Seventh (2017) International Ocean and Polar Engineering Conference*, 2017.

[69] F. Menter. Zonal two equation $k\omega$ turbulence models for aerodynamic flows. In *23rd Fluid Dynamics, Plasma Dynamics, and Lasers Conference*, page 2906, 1993.

[70] F. Menter, M. Kuntz, and R. Langtry. Ten years of industrial experience with the SST turbulence model. *Turbulence, Heat and Mass Transfer*, 4(1):625–632, 2003.

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

[72] J. H. Michell. Xi. The wave-resistance of a ship. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 45(272):106–123, 1898.

[73] twodoscillatingdisplacement: point displacement boundary condition for openfoam. `https://github.com/HIKassem/twoDOscillatingDisplacement/blob/master/README.md`. Accessed: 2019-10-23.

[74] R. D. Murphy, D. J. Sabat, and R. J. Taylor. Least cost ship characteristics by computer techniques. *Marine Technology and SNAME News*, 2(02):174–202, 1965.

[75] John Nicholas Newman. *Marine Hydrodynamics*. MIT press, 2018.

[76] C. Oldfield, M. M. Larmaei, A. Kendrick, and K. McTaggart. Prediction of warship manoeuvring coefficients using CFD. In *World Maritime Technology Conference*, 2015.

[77] A. Olivieri, F. Pistani, A. Avanzini, F. Stern, and R. Penna. Towing tank experiments of resistance, sinkage and trim, boundary layer, wake, and free surface flow around a naval combatant INSEAN 2340 model. Technical report, Iowa Univ Iowa City Coll of Engineering, 2001.

[78] H. Orihara and H. Miyata. Evaluation of added resistance in regular incident waves by computational fluid dynamics motion simulation using an overlapping grid system. *Journal of Marine Science and Technology*, 8(2):47–60, 2003.

[79] A. P. Heredero, T. Xing, and F. Stern. URANS and DES analysis for a Wigley hull at extreme drift angles. *Journal of Marine Science and Technology*, 15(4):295–315, 2010.

[80] A. Papanikolaou. Holistic ship design optimization. *Computer-Aided Design*, 42(11):1028–1044, 2010.

[81] A. B. Phillips, S. R. Turnock, and M. Furlong. Evaluation of manoeuvring coefficients of a self-propelled ship using a blade element momentum propeller model coupled to a reynolds averaged navier stokes flow solver. *Ocean Engineering*, 36(15-16):1217–1225, 2009.

[82] H. P. Piehl. *Ship roll damping analysis*. PhD thesis, University of Duisburg Essen, 2016.

[83] J. E. Pilliod Jr and E. G. Puckett. Second-order accurate volume-of-fluid algorithms for tracking material interfaces. *Journal of Computational Physics*, 199(2):465–502, 2004.

[84] ITTC Procedures and Guidelines. Practical guidelines for ship CFD applications. *In Guidelines 2011; 26th ITTC Specialist Committee on CFD in Marine Hydrodynamics; Lyngby, Denmark*, pages 1–18, 2011.

[85] P. Queutey and M. Visonneau. Free-Surface Capturing RANSE Simulations for a Ship at Steady Drift. *Ship Technology Research*, 51(3):106–122, 2004.

[86] N. Sakamoto. *URANS and DES simulations of static and dynamic maneuvering for surface combatant*. PhD thesis, The University of Iowa, 2009.

[87] N. Sakamoto, P. M. Carrica, and F. Stern. URANS simulations of static and dynamic maneuvering for surface combatant: part 1. verification and validation for forces, moment, and hydrodynamic derivatives. *Journal of Marine Science and Technology*, 17(4):422–445, 2012.

[88] Z. R. Shen. *Development of overset grid technique for hull-propeller-rudder interactions*. PhD thesis, Shanghai Jiao Tong University, 2014.

[89] Z. R. Shen and D. C. Wan. RANS computations of added resistance and motions of a ship in head waves. *International Journal of Offshore and Polar Engineering*, 23(04):264–271, 2013.

[90] Z. R. Shen, D. C. Wan, and P. M. Carrica. RANS simulations of free maneuvers with moving rudders and propellers using overset grids in OpenFOAM. In *SIMMAN workshop on Verification and Validation of Ship Maneuvering Simulation Methods. Presented at the SIMMAN workshop on Verification and Validation of Ship Maneuvering Simulation Methods, Lyngby, Denmark*, 2014.

[91] Z. R. Shen, D. C. Wan, and P. M. Carrica. Dynamic overset grids in OpenFOAM with application to KCS self-propulsion and maneuvering. *Ocean Engineering*, 108:287–306, 2015.

[92] SIMMAN 2008 Workshop. `http://www.simman2008.dk/`. Accessed: 2019-3-30.

[93] SIMMAN 2014 Workshop. `https://simman2014.dk/`. Accessed: 2019-3-30.

[94] SIMMAN 2019 Workshop. `http://www.simman2019.kr/`. Accessed: 2019-3-30.

[95] C. D. Simonsen. PMM model test with DDG51 including uncertainty assessment. *FORCE Technology, Department of Maritime Industry, Lyngby*, 2004.

[96] C. D. Simonsen, J. F. Otzen, C. Klimt, N. L. Larsen, and F. Stern. Maneuvering predictions in the early design phase using CFD generated PMM data. In *29th Symposium on Naval Hydrodynamics*, pages 26–31, 2012.

[97] C. D. Simonsen, F. Stern, and F. Quadvlieg. SIMMAN 2014: Workshop on verification and validation of ship manoeuvering simulation methods. In *Workshop Proceedings, Copenhagen, Denmark, Dec*, pages 8–10, 2014.

[98] R. Skjetne. Ship maneuvering: the past, the present and the future. *Sea Technology*, 2003.

[99] F. Stern, K. Agdrup, S.Y. Kim, A.C. Hochbaum, K.P. Rhee, F.H.H.A. Quadvlieg, P. Perdon, T. Hino, R. Broglia, and J. Gorski. Experience from SIMMAN 2008 —- the first workshop on verification and validation of ship maneuvering simulation methods. *Journal of Ship Research*, 55(2):135–147, 2011.

[100] F. Stern, J. Longo, R. Penna, A. Olivieri, T. Ratcliffe, and H. Coleman. International collaboration on benchmark CFD validation data for surface combatant DTMB model 5415. In *Twenty-Third Symposium on Naval Hydrodynamics Office of Naval Research Bassin d'Essais des Carenes National Research Council*, 2001.

[101] F. Stern, J. M. Yang, Z. Y. Wang, H. S. Hosseini, M. Mousaviraad, S. Bhushan, and T. Xing. Computational ship hydrodynamics: nowadays and way forward. *International Shipbuilding Progress*, 60(1-4):3–105, 2013.

[102] J. S. Suh, J. M. Yang, and F. Stern. The effect of air–water interface on the vortex shedding from a vertical circular cylinder. *Journal of Fluids and Structures*, 27(1):1–22, 2011.

[103] M. Sussman and E. G. Puckett. A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows. *Journal of Computational Physics*, 162(2):301–337, 2000.

[104] I. A. Svendsen and I. G. Jonsson. *Hydrodynamics of coastal regions.* Den Private ingeniørfond, Technical University of Denmark, 1976.

[105] M. Terziev, T. Tezdogan, E. Oguz, T. Gourlay, Y. K. Demirel, and A. Incecik. Numerical investigation of the behaviour and performance of ships advancing through restricted shallow waters. *Journal of Fluids and Structures*, 76:185–215, 2018.

[106] Tokyo 2015: a workshop on CFD in ship hydrodynamics. `https://t2015.nmri.go.jp/announcement.html`. Accessed: 2019-3-30.

[107] S. L. Toxopeus. *Practical application of viscous-flow calculations for the simulation of manoeuvring ships.* PhD thesis, Technische Universiteit Delft, 2011.

[108] S. R. Turnock, A. B. Phillips, and M. Furlong. URANS simulations of static drift and dynamic manoeuvres of the KVLCC2 tanker. In *In: Proceedings of the SIMMAN 2008: Workshop on Verification and Validation of Ship Manoeuvring Simulation Methods, Lyngby, Denmark, 13–17 April.*, 2008.

[109] O. Ubbink. *Numerical prediction of two fluid systems with sharp interfaces.* PhD thesis, University of London, UK, 1997.

[110] O. Ubbink and R. I. Issa. A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics*, 153(1):26–50, 1999.

[111] G. Vaz, F. Jaouen, and M. Hoekstra. Free-surface viscous flow computations: validation of URANS code FRESCO. In *ASME 2009 28th International Conference on Ocean, Offshore and Arctic Engineering*, pages 425–437. American Society of Mechanical Engineers, 2009.

[112] V. Vukcevic, A. Ostman, and H. Jasak. Rapid simulations of pure sway motion using FVM in OpenFOAM. In *Proceedings of Workshop on Verification and Validation of Ship Manoeuvring Simulation Methods (SIMMAN 2014)*, 2014.

[113] J. H. Wang and D. C. Wan. Numerical simulation of pure yaw motion using dynamic overset grid technology. *Chinese Journal of Hydrodynamics*, 31(5):567–574, 2016.

[114] J. H. Wang, W. W. Zhao, and D. C. Wan. Development of naoe-FOAM-SJTU solver based on OpenFOAM for marine hydrodynamics. *Journal of Hydrodynamics*, 31(1):1–20, 2019.

[115] H. Weking, J. Schlottke, M. Boger, P. Rauschenberger, B. Weigand, and C.-D. Munz. DNS of rising bubbles using VOF and balanced force surface tension. In *High Performance Computing on Vector Systems 2010*, pages 171–184. Springer, 2010.

[116] M. P. Wood, L. M. González, J. Izquierdo, A. Sarasquete, and L .P. Rojas. RANSE with free surface computations around fixed DTMB 5415 model and other baliño's fishing vessels. In *9th International Conference on Numerical Ship Hydrodynamics, Michigan, USA*, pages 5–8, 2007.

[117] H. S. Yoon. *Phase-averaged stereo-PIV flow field and force/moment/motion measurements for surface combatant in PMM maneuvers.* PhD thesis, University of Iowa, 2009.

[118] H. S. Yoon, L. Gui, S. Bhushan, and F. Stern. Tomographic PIV Measurements for a Surface Combatant at Straight Ahead and Static Drift Conditions. In *30th Symposium on Naval Hydrodynamics, Hobart, Tasmania, Australia, Nov*, pages 2–7, 2014.

[119] H. S. Yoon, C. D. Simonsen, L. Benedetti, J. Longo, Y. Toda, and F. Stern. Benchmark CFD validation data for surface combatant 5415 in PMM maneuvers. Part I: Force/moment/motion measurements. *Ocean Engineering*, 109:705–734, 2015.

# APPENDICES

# Appendix A

# Multiphase Modelling and Meshing Requirements for Predicting Naval Ship Maneuvering Forces

## A.1 Abstract

The modelling complexity in predictive naval ship simulations stems from the multiscale and multiphase nature of the fluid dynamic problem. The accurate prediction of forces and moments are predicated on the correct modelling of these complex and coupled physical phenomena. Yet, often, researchers limit their numerical verification to matching well-established experimental results and inherently assume that if the forces and moments match experiments then the physics are adequately modelled. In the present work, we question this premise and specifically focus on the multiphase and multiscale modelling in this problem. In ship simulations, the wave along the interface, turbulent wake and flow separated regions, solid-liquid interface, and the near wall resolution can influence the forces and moments. We first conduct a detailed analysis on the canonical Wigley hull to assess the primary modelling errors in ship simulations. Based on these results and in order to isolate the multiphysics affects, we study four canonical subproblem which are representative of various flow features in naval simulations. These subproblems allow us to isolate and assess the modelling requirements for the: (1) surface wave propagation, (2) the mesh/flow field alignment, (3) run-up distance in multiphase flows, and (4) wake region/flow separation. The study of these detailed subproblems are informative for ship simulation engineers, as they provide guidelines for the accurate modelling of some of the important physical features of naval ship simulations.

## A.2 Introduction

Ship maneuvering performance is critical for safe operation and successful completion of missions [74, 80, 98]. The maneuverability characteristics of a vessel can be estimated through theoretical calculations, experimental/empirical models or, numerically, through the use of Computational Fluid Dynamics (CFD). The theoretical approach [4, 72] is limited to a slender body approximation and does not consider the interaction between the hull and the appendages, nor the more complex non-linear effects; this is best suited for order-of-magnitude maneuverability estimations. Physical modelling [21, 22] is the standard approach for the ship maneuverability estimates and relies on the measurement of forces and moments on a subscale ship model undergoing static and dynamic planar motion mechanism (PMM) tests. These experimental tests are expensive and time consuming as special experimental platforms are required. As a result, this approach is ill-suited for iterative ship optimization or design. Continually increasing computational power enables the use of CFD to numerically compute static and dynamic stability derivatives on arbitrarily complex ship geometries [87, 91, 119]. Although CFD is now considered a standard design tool, the multiphase and multiphysics challenges associated with predictive naval simulations render them prone to modelling and numerical errors. These insidious errors must be understood and quantified in order to be able to fully rely on predictive modelling results.

One of the challenges in naval simulations is to accurately account for the effects of turbulence due to the complexity and multiscale nature of the problem. Currently there are three main paradigms for CFD simulations of multiphase turbulent flows: Direct Numerical Simulation (DNS) [115], Large Eddy Simulation (LES) [47] and Reynolds-averaged Navier-Stokes equations (RANS) [91]. Owing to the high Reynolds number associated with ship maneuvering flows (generally from $10^6$ to $10^9$) , DNS and LES are impractical in most engineering/design contexts. RANS modelling relies on the time-averaged solution of the Navier-Stokes equations for which the effects of the turbulence are modelled through a number of characteristic turbulent parameters. Due to the more tractable computational cost, RANS modelling remains the standard tool for ship maneuvering simulations and has been the focus of many previous studies. Carrica [9] investigated steady turn and zigzag maneuver simulations with the model MARIN 7967 using the CFDShip-Iowa solver, which is an Unsteady-RANS, or URANS, solver using a blended $k-\epsilon/k-\omega$ model for the turbulence. The differences between simulations and experiments were within 10% for characteristic maneuvering parameters, such as tactical diameter. This was a clear improvement from the earlier predictive RANS-based simulations on these same test cases [92, 99]. Shen *et al.* [89] investigated added resistance, heave and pitch motions of a US Navy combatant, DTMB 5512 in head waves using OpenFOAM with a URANS turbulence closure; this six degrees-of-freedom (6-DoF) motion simulation was computed using a dynamic mesh. The simulation results were validated with experimental data [27, 28, 35]. More recently, Islam [36] computed PMM maneuvers of the KRISO Container ship (KCS) using OpenFOAM with URANS. Compared with experimental data, the relative errors are small in most cases, except in pure yaw motion. Although these promising results are encouraging, especially when considering integral parameters such as forces and moments, numerical prediction of naval ship maneuvering remains highly complex as one must consider the adequate temporal and spatial resolution, modelling of complex multiphysics interactions including the gas-liquid interface, complex turbulence effects, and the important wave-hull interactions.

As naval ship maneuvering is characterized by highly-coupled and complex physical phenomena, the CFD simulations of these flows are prone to numerical and modelling errors which can result in incorrect predictive estimates. The standard practice is to assess the validity of the CFD by comparing the numerical predictions with well-defined experimental test cases, such as those defined in the context of the ITTC Proceedings [37, 84] and the SIMMAN Workshop on Verification and Validation of Ship Maneuvering Simulation Method [92, 93, 94, 97, 99]. This essential validation step provides an assessment of the predictive capabilities of the numerical solvers based on the comparison of the measured integral quantities, such as forces and moments. The validation based on integral quantities, although critical, does not provide any indication on the adequacy of the simulation parameters to model the multiphysics interactions inherent to ship modelling simulations. Given the geometric complexity and multiscale nature of the problem, the incorrect modelling of the multiphysics interactions may limit the generalizability of these predictive simulations. In other words, a correct estimate of the integrated force on a ship, for example, does not necessarily mean that all the physics are correctly modelled.

Some works have proposed criteria for modelling multiphase turbulent flow in naval simulations. To evaluate the performance of the interface capture method, Suraj [17] did an in-depth investigation into the interFoam solver using a variety of validation cases: pure advection cases (kinematics), high Weber number limit cases (dynamics) and surface tension-dominated flows. The kinematic tests show that the performance of interFoam solver is similar with the algebraic VOF method [25], while it is worse than the geometric reconstruction schemes [83, 103]. The simulation results agree with analytical and experimental data in the inertial-dominated flows. For the simulation of surface tension-dominated flows, the interFoam solver provides a consistent formulation for pressure and surface tension. Although the pressure in the simulation is consistent with the analytical values, there is a 10% lower peak in the simulation compared to experiment, which could be due to the error of interface curvature prediction [17]. For the surface wave propagation, Larsen [54] evaluated the performance of the interFoam solver in OpenFOAM for the simulation of propagation of nonlinear regular waves. It was observed that over time, several undesirable effects arise: increase of surface elevation, oscillations in the free surface, and the over estimation of crest velocities; these errors could be mitigated by adjusting temporal and spatial resolution, Courant number, and the selection of discretization schemes. To provide fundamental understanding for engineers in floating offshore structures, Benitz [6] studied the drag coefficient of multiphase turbulent flow over surface-piercing cylinders with free lower ends at different aspect ratios using OpenFOAM and experiment. Based on the simulation results and the experimental data, the drag coefficient increases as the aspect ratio increases, which could be due to the suppressed of vortex shedding by the free surface at small aspect ratio. Even after grid-independence tests, the relative error of drag coefficient is more than 10% at the smaller aspect ratio. This could be due to insufficient modelling of the complex turbulence-interface interaction in the case of small aspect ratio. For ship resistance, Martic [68] performed a mesh sensitivity analysis for unstructured grids for numerical simulation of a damaged ship model using STAR-CCM+. The unsteady flow effect occurred due to the sharp geometry near the hole in the ship model and at the transverse bulkheads. Even though a steady resistance simulation was only obtained on the finest mesh, the time averaged results on the coarse, medium mesh and fine mesh are in good agreement with experimental data.

To the authors knowledge, no work has specifically addressed the comprehensive multiphysics modelling challenges and numerical errors associated with multiphase naval simulations. The present work seeks to provide a quantitative understanding of the modelling and meshing requirements needed to capture the multiphysics interactions which are characteristic in naval ship simulations. We first simulate the canonical Wigley hull as a benchmark validation case to assess the error with the well-established experimental validation case. Thereafter, the complexity of a ship simulation is decomposed into well-defined and canonical sub-problems, similar to the work done by Mani [67]. This allows us to isolate the various multiphysics interactions and assess the numerical and modelling error for each of these sub-problems against analytical or well-established experimental results. In the present work, we investigate the numerical simulation requirements for: (1) wave propagation at the water-air interface, (2) the mesh/flow field alignment, (3) run-up distance in multiphase flows, and (4) wake region/flow separation. The canonical nature of these flows permits an isolation of the physical complexity, thus enabling quantifiable guidelines for ship modelling.

## A.3 Description of Sub-Cases

Based on the systemic errors arising from the analytical ship geometry in the previous section, we decompose the naval ship problem into representative sub-problems. Although they may abstract from the true complexity of the naval ship simulations, these representative sub-problems admit either analytical solutions or well-defined experiments that serve as a basis to assess the modelling and numerical errors. This section describes these sub-problems and the modelling parameters that are used; the following section presents the results of these cases.

### A.3.1 Purpose of Sub-Cases

As shown in the previous section, predictive modelling of the analytical geometry of Wigley hull can lead to significant errors at large drift angle, even though a high-resolution, body-conforming mesh is used. The force and moments on the hull are surface integrated parameters and many factors may influence these results. Among the main parameters influencing these results are the surface wave propagation, the numerical modelling of the solid-liquid interface, the interaction between flow separation at the bow and the wake, and the numerical errors associated with angle between the main flow and the mesh. In this paper, the complexity of the multiphase ship simulations is further broken down into four tractable sub-cases that are studied via well-defined, canonical test cases. Figure A.1 illustrates a typical ship simulation with four separate test cases. The object of each sub-case is presented in the following paragraph.

- **Surface wave propagation**: The first case involves the study of numerical surface wave propagation. The waves propagating at the liquid-air interface dissipate energy away from the ship and represent a critical parameter for wave damage as well as the simulation of ship navigation in rough waters. To evaluate the simulation requirements for this case, we investigate the analytically defined case of a Stokes $1^{st}$ order wave propagation [104] using waves2Foam toolbox [38]. The wave parameters are calculated based on the Kelvin-ship wave theory [75] and the related parameters of the Wigley hull simulation shown in Section 5.1.
- **The mesh/flow field alignment**: The second case investigates the effect of castellated mesh and the mesh/flow field alignment on ship maneuvering simulations. For complex geometries, such as ships, body conforming meshes are sometimes difficult to generate, especially in the context of a dynamic simulation. As a result, castellated meshing (such as the snappyHexMesh package in OpenFOAM) is often used for grid generation. In addition, in ship simulation with a drift angle, it is often difficult to generate a high-quality mesh in cubic computational domain with the rotation of ship to the prescribed drift angle [36]. Usually, we simply change the direction of inlet velocity in the steady drift simulation. In this case, the main flow direction may not be aligned with the mesh, thus leading to greater numerical error. To quantitatively analyze the numerical error related with the castellated mesh and the angle between the main flow and the mesh, a cylinder in single phase flow with a castellated mesh using snappyHexMesh was investigated.
- **Solid-liquid interface**: The wet area of the hull represents a key parameter that needs to be accurately captured by naval simulations. As the wetted surface increases, so does the pressure acting on the surface–this affects the drag estimation from the simulation. To study the effects of solid-liquid interface modelling, we investigated the multiphase turbulent flow over surface-piercing cylinder [102]. In this case, we focus in the effect of mesh resolution on the prediction of wave run-up height on the surface of cylinder and the drag coefficient. An analytical expression for the non-dimensional wave run-up height based on the Froude number was used as a validation of the numerical simulation data.
- **Wake region/flow separation**: At large drift cases, the separation of the flow near the bow and the stern plays a significant effect on the drag and side force. The unsteady nature of flow separation makes it difficult to correctly predict the drag force using a steady solver

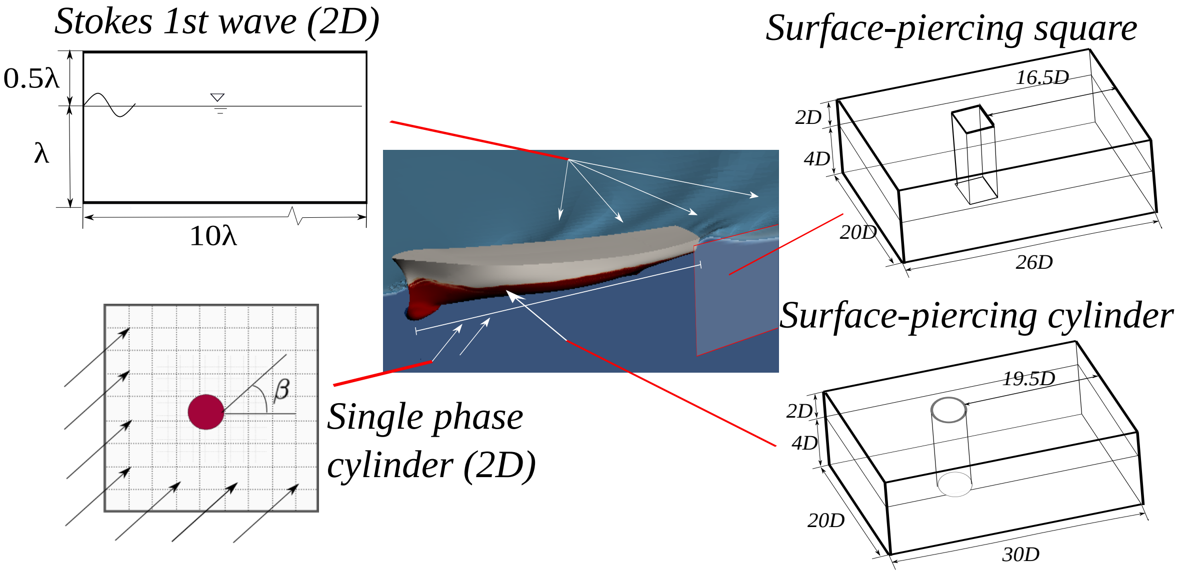


Figure A.1: A representative illustration of the canonical sub-problems for naval simulations. The important geometric details of the cases are defined. Cases are: (top left) surface wave propagation-Stokes $1^{st}$ order wave, (top right) ship wake modelling/flow separation-surface piercing square, (bottom left) flow/mesh alignment-single phase cylinder, (bottom right)solid-liquid interface at the bow-surface piercing cylinder.

or a RANS model with wall function. To investigate the effect of flow separation on drag prediction, numerical simulation of the flow over surface-piercing square was performed at $Re = 2100$, and $Fr = 0.1$; note, the Froude number is set to decrease the effects of wave.

### A.3.2 Case Details and Boundary Conditions

As shown in Table A.1, the baseline configurations for all sub-cases are defined. The computational domain, mesh and boundary condition in each case are described in the following paragraph. Compared with ship simulations, the current Reynolds number for sub-cases are smaller. However, these Reynolds number are selected for the representative benchmark cases. Since each sub-case captures the key phenomenon in ship simulations, the modelling and meshing requirements concluded from these sub-cases would also work in ship simulations.

The computational geometry for the numerical propagation of a Stokes $1^{st}$ order wave is shown in top left of Figure A.1. The length of the tank is ten times the wavelength of Stokes $1^{st}$ order wave ($10\lambda$) with $5\lambda$ long numerical sponger layer to avoid the reflection from the outlet boundary.

| | **Stokes $1^{st}$ order wave** | **Single-phase cylinder** | **Multiphase cylinder** | **Multiphase Square** |
|---|---|---|---|---|
| **Purpose of simulation** | Wave propagation | flow/mesh alignment | Solid-liquid interface | Wake region/flow separation |
| **Dimension** | 2D | 2D | 3D | 3D |
| **Steady** | No | Yes | Yes | No |
| **Reynolds number** | - | 40 | 27000 | 2100 |
| **Froude number** | - | - | 0.8 | 0.1 |
| **Turbulence model** | laminar | laminar | $k-\omega$ SST | $k-\omega$ SST |
| **Mesh type** | Castellated | Castellated | Body-conforming | Body-conforming |
| **Mesh resolution $y^+$** | - | - | 60 | 3.5 |
| **Total grid points** | 2.95E3 | 4.64E3 | 5.06E4 | 1.10E5 |

Table A.1: Baseline parameters for all sub-cases.

The height of the tank is $1.5\lambda$ relative to a water depth of $\lambda$, which is set for the generation of deep water wave. The wave profile is analytically defined by the equation: $h = H\cos(kx - \omega t)$. In the current simulation, $\lambda = 2\pi Fr^2 L_{pp}$, $k = 2\pi/\lambda$, $\omega = \sqrt{gk}$, $H = \lambda/50$ ($Fr = 0.25$, $L_{pp} = 2$, based on the Wigley hull case in Section 5.1). Time step of the simulation was set as $T/100$ to satisfy a $CFL \leq 0.25$. To capture the interface, the mesh was refined in the region of $y$ $[-2H, 2H]$ in $x$ and $y$ directions simultaneously using the snappyHexMesh utility. The number of points per wave length ($ppw = 2\pi/(k\Delta x)$), and the number of points per wave height ( $pph = H/(2\Delta y)$) are used to describe the mesh resolution in each direction. The baseline mesh was set as $ppw \times pph$=$10\times 1$, with 2.95E3 grid points.

In the case of single phase flow over a cylinder, a castellated mesh (in a square domain) is investigated. The size of the square domain is $45D \times 45D$ (D is the diameter of the cylinder). The cylinder is located in the center of the domain. To investigate the angle $\beta$ between flow and mesh in the square domain, two boundaries are set as inlet velocity boundary condition, with the remaining zero gradient outlet boundary condition, as shown in the bottom right of Figure A.1. For the castellated mesh, a uniform, two dimensional background mesh is generated and then a local mesh refinement and adaptation is performed (via the snappyHexMesh utility); we select 5 levels refinement from the initial coarse mesh. Finally, six body-fitted layers are applied near the wall, with stretching ratio 1.1; the final layer's ratio is 0.7 times of background mesh. A uniform background mesh with 25 points in each direction was set as baseline mesh, with 4640 grid points after the mesh refinement and adaptation.

In the case of the multiphase flow over cylinder, the computation domain size is set as $L_x \times L_y \times L_z = 30D \times 20D \times 6D$, where $D$ is the cylinder diameter. The distance between the center of cylinder and the outlet is set as 20D, in order to avoid nonphysical reflection from the outlet boundary. The water depth is set to 4D. Velocity is prescribed for the inlet boundary condition, symmetry boundary condition is set for the bottom of the tank, and zero gradient boundary condition is set at the remaining boundaries. Time integration was performed via a first order, pseudo transient local Euler scheme.

In multiphase turbulent flow over surface-piercing square, the computational domain is set as $L_x \times L_y \times L_z = 26D \times 20D \times 6D$, and the water depth is set as $4D$. The height of square is set as $H = 6$, and the length of square is set as $D = 1$. The center of square is located at the origin of the domain. Body-conforming mesh is generated with a local wall stretching ratio of 1.1. The Reynolds number was defined as $Re = UD/\nu$, where U is the free stream velocity, and $\nu$ is the kinematic viscosity of water. Time was non-dimensionalized as $t' = tU/D$. A first order Euler scheme is used for time marching and $k - \omega$ SST model without wall function is used for turbulence modelling in this case due to the unsteady flow separation in the wake. The time step was set to satisfy $CFL \leq 0.5$ and the simulation was run up to non-dimensional time $t/T = 80$ in order to attain a statistically steady result.

## A.4 Results

In this section, the numerical results of all sub-cases, defined in the previous section, are presented. All the results shown in the following paragraphs are fully converged in steady cases or statistical steady in unsteady cases.

### A.4.1 Resolution Requirements for Wave Propagation

The wave propagation plays an important role in the wave induced resistance, which is the major source of drag in ship navigation. The Stokes $1^{st}$ order wave represents an analytically defined linear wave and it is valid for deep-water waves when the wavelength is much larger than the wave height. As the analytical solution is known, the numerical error is defined as $|h_{numerical} - h_{analytical}|/H$. To investigate the mesh resolution requirements for $ppw$ and $pph$, numerical simulations were performed in four mesh resolutions: $ppw \times pph$=$10 \times 1$, $20 \times 3$, $40 \times 6$, $80 \times 12$. The number of grid points for each mesh resolution is 2.95E3, 1.48E4, 5.89E4, and 2.33E5. The non-dimensional height of wave $h' = h/H$ at the time $t/T = 9$ 10 is presented in Figure A.2 with different mesh resolutions. The simulation results converge to the analytical solution monotonically as mesh resolution increases and we conclude that 40 points per wave length and 6 points per wave height are needed to reach an average error small than 2% for the Stokes $1^{st}$ order wave simulation, which is in agreement with ITTC guidelines [84]. In multiphase ship simulations, a relation between Kelvin wave length and Froude number is defined as $\lambda = 2\pi Fr^2 L_{pp}$ for deep water. The current criteria of $ppw = 40$ implies that for a given ship simulation, a decrease in the Froude number leads to a decrease of Kelvin wave length and, consequently, more grid points in the horizontal plane are needed to resolve the surface wave. On the other hand, a decrease of the Froude number results in Reynolds number decrease, resulting in a coarser mesh requirement near the wall; this means that fewer grid points are required in the direction normal to the ship surface.

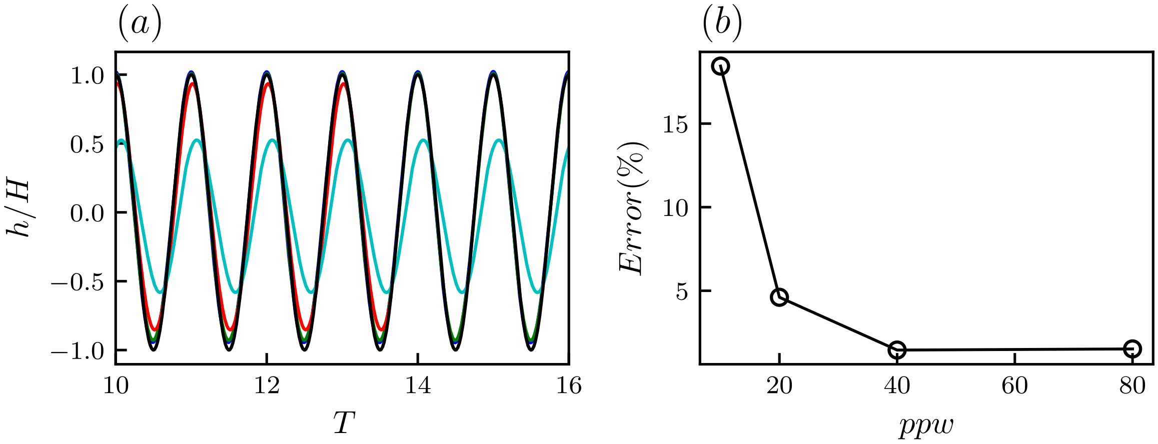


Figure A.2: Non-dimensional wave profile at different mesh resolutions ($ppw \times pph$). (a)– analytical solution; – $10 \times 1$; – $20 \times 3$; – $40 \times 6$; – $80 \times 12$; (b) numerical error compared with the analytical solution: $\overline{|h_{numerical} - h_{analytical}|}/H$.

### A.4.2 Near-Wall Resolution Effect

To compare the effect of castellated mesh and the body-conforming mesh on the drag prediction, a single phase flow over circular cylinder at $Re = 40$ is investigated. The drag coefficients with different castellated mesh resolution are shown in Figure A.3. As the increase resolution of the castellated mesh, the drag coefficient converges to 1.554, close to the result 1.522 from the high resolution body-conforming mesh result by Dennis [16], and the result 1.54 from the immersed boundary method by Linnick [60].

The above simulations were performed in the idealized case when the grid and mean flow field are in perfect alignment. In many ship simulations, this may not be the case. For example, in

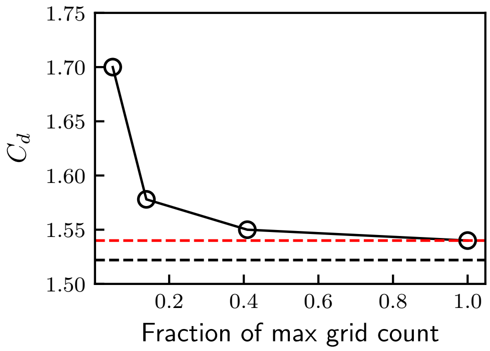


Figure A.3: Drag coefficient of cylinder at Re=40 in different mesh resolutions (maximum number of points: $5.2 \times 10^4$): ○ result in the current simulation (castellated mesh); −− result from Dennis [16]; −− result from Linnick [60].

| $\beta$ [°] | 0 | 10 | 20 | 30 | 40 | 45 | Linnick [60] | Dennis [16] |
|---|---|---|---|---|---|---|---|---|
| $C_d$ | 1.535 | 1.536 | 1.541 | 1.545 | 1.546 | 1.546 | 1.54 | 1.522 |

Table A.2: Drag coefficient ($C_d$) of cylinder (Re=40) at different angle $\theta$ between the main flow and the mesh

pure drift ship simulations, we often modify the direction of the incoming velocity vector. As such the underlying mesh remains unaffected but it is no longer aligned with the principal flow direction. To investigate the effect of angle between the mesh and the main flow, a flow over cylinder in square domain with different angle of attack is studied. For this canonical cylinder flow, the magnitude of the drag coefficient should remain the same no matter what the angle of incoming flow; thus, the changes in the drag coefficient are a direct result of the increased in numerical error due to the mesh/flow misalignment. The streamlines at different angle of attack (0°, and 45°) are shown in Figure A.4, where slight differences are observed. The drag coefficient at different angle of attack is listed in Table A.2, and ranges from 1.535 to 1.546 as the angle varies from 0° to 45°. The deviation is smaller than 0.7%, which means that the numerical error due to misalignment of the mesh and flow plays a minor role in the drag prediction. That said, this error may be more significant if the simulation is marginally resolved.

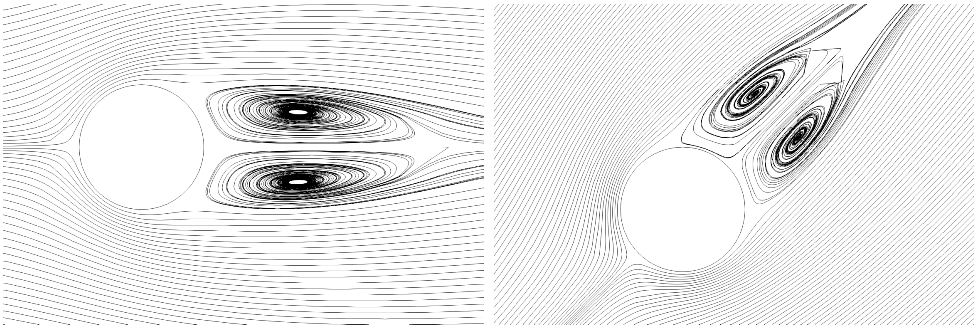


Figure A.4: Streamline for different angle of attack in the case of 2D cylinder at Re=40: left $\beta = 0°$; right $\beta = 45°$

### A.4.3 Air-Water Interface of Cylinder in Crossflow

In order to quantify the resolution requirement to capture the run-up height, which is important for ship drag computation of naval flows, we study the flow over surface piercing cylinder at $Re = 27,000$, $Fr = 0.8$ with a body-conforming mesh. The objective of this case is to investigate mesh resolution needed to capture run-up height and drag coefficient, which are validated against a highly-resolved Large Eddy Simulation (LES) [102] and supplemented with analytical comparisons using Bernoulli's equation: $h/D = Fr^2/2$. Table A.3 shows the drag coefficient and non-dimensional run-up height at different mesh resolutions; the relative error in the drag coefficient decreased from 7.2% to 5.7% with a significant increase in mesh resolution (first layer distance $y^+$=10). The relative wave run-up height between numerical simulation data and theoretical result decreases significantly from 32.8% to 7.8% with increasing mesh resolution. These results support the idea that a correct prediction of drag coefficient does not guarantee good accuracy of run-up height prediction. The predicted accuracy of drag coefficient does not improve as significantly as the run-up height, which could be due to the relatively small wetted surface area change induced by the change of run-up height. However, in the ship simulation, the wave induced drag plays a significant role in the total drag, which means the solid-liquid interface must be captured accurately.

To further investigate the effect of Froude number on wave run-up height, different Froude number $[0.2, 1.6]$ cases are simulated at the same Reynolds number $Re = 27,000$, using the fine mesh as shown in Table A.3. The representative side view of the run-up height can be seen in Figure A.5. Figure A.5 presents the wave profile near the cylinder: strong run-up in front of the cylinder and depression at the rear of the cylinder are observed. The wave run-up height at different Froude numbers in the simulation gives good agreement with the theoretical result from the Bernoulli equation, as shown in Figure A.6. The run-up height increases monotonically with the increase of Froude number. The current relation between wave run-up height and Froude number $h = Fr^2D/2$, and the criteria of $pph = 6$ from the Stokes $1^{st}$ order wave in Section A.4.1

| **Cases** | $\mathbf{C_d}$ | **Error** $C_d$ [%] | $\mathbf{H_{max}/D}$ | **Error** $H_{max}/D$[%] | $y^+$ | **Grids points** |
|---|---|---|---|---|---|---|
| **Coarse** | 0.91 | 7.2 | 0.215 | 32.8 | 60 | 5.06E4 |
| **Medium** | 0.93 | 5.4 | 0.37 | 15.6 | 42 | 1.35E5 |
| **Fine** | 1.04 | 5.7 | 0.345 | 7.8 | 10 | 3.86E5 |
| **LES** [102] | 0.984 | - | - | - | 0.485 | 8.39E6 |

Table A.3: Grid parameters, drag coefficients $C_d$, and run-up height $H_{max}/D$ in the case of surface-piercing cylinder. The comparative LES results are from Yang et al. (2011) [102].

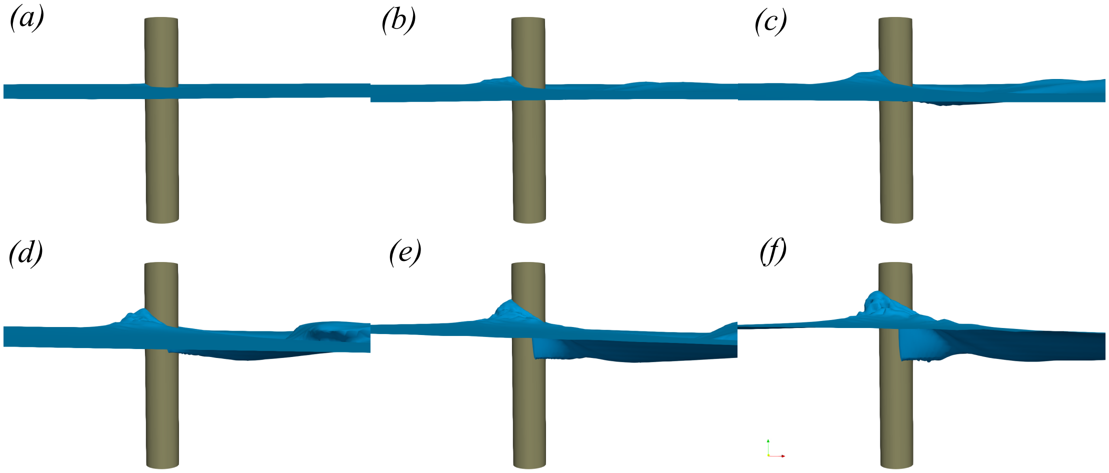


Figure A.5: Interface around the cylinder at (a) $Fr = 0.4$, (b) $Fr = 0.8$, (c) $Fr = 1.0$, (d) $Fr = 1.2$, (e) $Fr = 1.4$, (f) $Fr = 1.6$

imply that a higher mesh resolution in the direction of wave height is needed in high Froude ship simulations (for a same Reynolds number).

### A.4.4 Wake Region/Flow Separation Modelling: Flow Over Surface Piercing Square

For large drift angle ship maneuvering simulations, the flow separation appears in the front of the ship, which influences the drag coefficient significantly. To investigate the modelling requirement of flow separation at large drift angles for ship maneuvering simulations, an unsteady surface piercing square in a multiphase turbulent flow is simulated at $Re = 2,100$, and $Fr = 0.1$. The

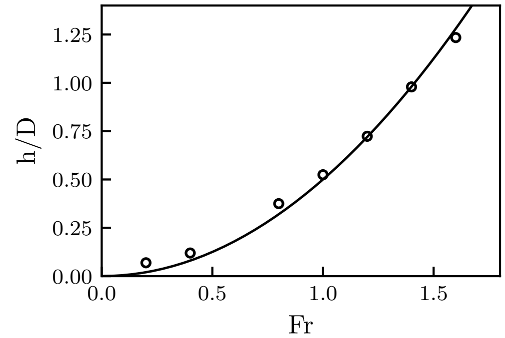


Figure A.6: Wave run-up height in the case of surface-piercing cylinder, − analytical solution, ∘ OpenFOAM result.

small Froude number is set to minimize the effect of the wave on the interface. The numerical simulation was performed with three different mesh resolutions: $y^+ = 3.5$ (coarse), 2.5 (medium), 1.75 (fine). The mesh resolutions were set to maintain a refinement ratio of $\sqrt{2}$. The number of grids in different meshes are 0.11 million (coarse), 0.23 million (medium), and 0.39 million (fine). The history of drag coefficients over time and time averaged results from different meshes are plotted in Figure A.7. All results reach statistically steady state. The result from the mid mesh converges well to the result from the fine mesh. Time-averaged drag coefficients were obtained as $\overline{C}_{d1} = 1.449$, $\overline{C}_{d2} = 1.505$, $\overline{C}_{d3} = 1.510$ in the case of $y^+ = 3.5$, 2.5, and 1.75, respectively. The deviation of time-averaged drag coefficient between the case with coarse mesh and mid mesh is $|\overline{C}_{d1} - \overline{C}_{d2}|/\ \overline{C}_{d2} = 3.7\%$, and it decreases to $|\overline{C}_{d3} - \overline{C}_{d2}|/\ \overline{C}_{d3} = 0.3\%$ from the result with mid mesh and refine mesh. The deviation between results from coarse mesh and mid mesh indicates that the drag coefficient is very sensitive to the mesh resolution in the separation flow, even though the first layer distance satisfies $y^+ \leq 3.5$ in all of the current simulations. Figure A.8 presents four snapshots of streamline derived from velocities on the horizontal plane $z/D = -2$ in the case of mid mesh. Flow separation occurs in the front edge and the rear edge. An asymmetrical vortex pair is attached near the rear of the square. The relative size of vortex changes periodically with the time, which leads to the periodic oscillation of the drag coefficient. The streamlines near the front edge of the square are further zoomed in, as presented in Figure A.9. Similar to the flow separation in the rear region, strong asymmetric streamlines near the top edge and the bottom edge appear and evolve periodically, which also influences the drag coefficient. The surface piercing square case illustrates the challenge of the drag coefficient prediction with the flow separation, even with high resolution mesh.

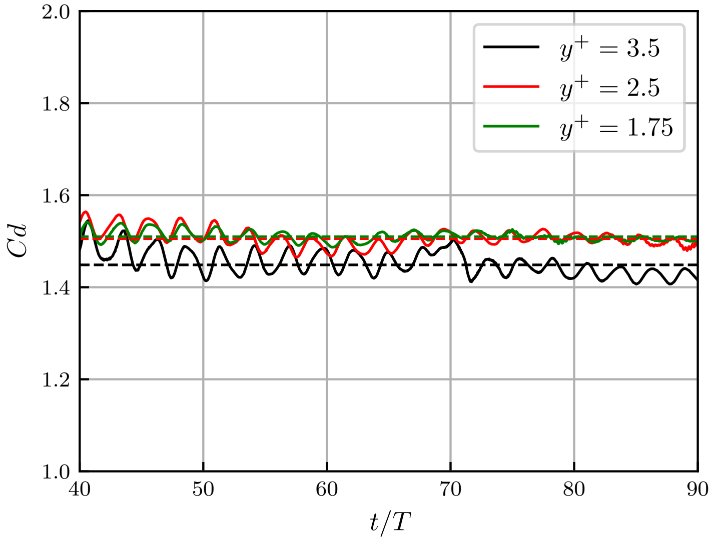


Figure A.7: History of drag coefficients ($C_d$) in the case of surface-piercing square with different mesh resolutions

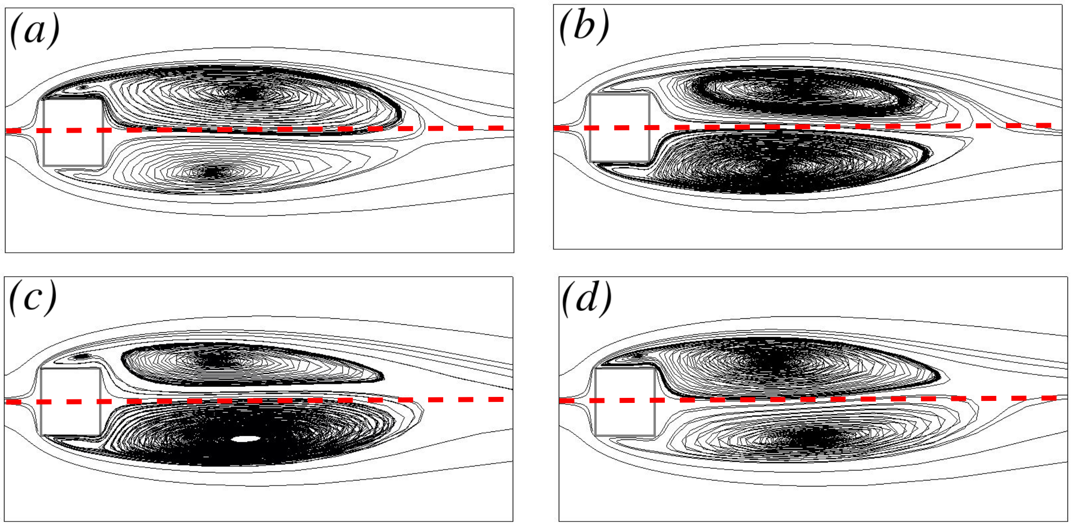


Figure A.8: Instantaneous streamlines near the rear of the square in the plane $z/d = -2$. (a)$t/T = 82$, (b)$t/T = 84$, (c)$t/T = 86$, (d)$t/T = 88$.

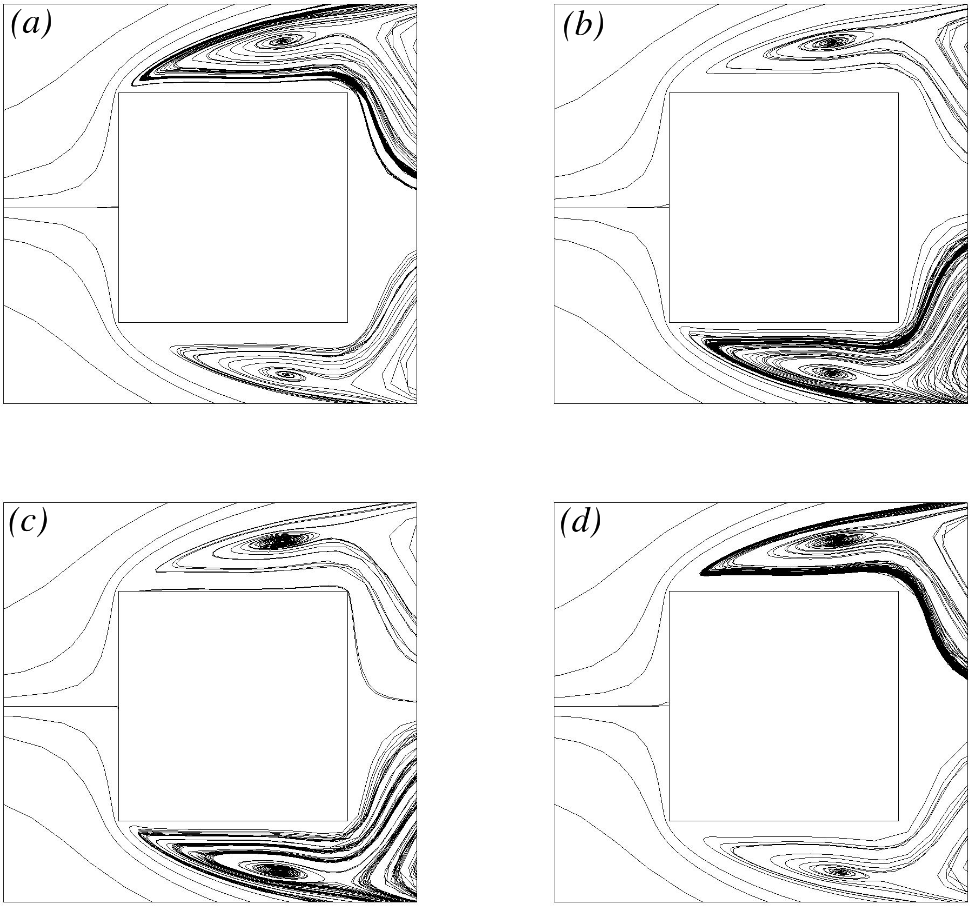


Figure A.9: Instantaneous streamlines (near the front edge of the square) in the plane $z/d = -2$. (a)$t/T = 82$, (b)$t/T = 84$, (c)$t/T = 86$, (d)$t/T = 88$.

## A.5 Conclusion and Discussion

Numerical modelling of naval ship maneuvering is a necessary complement to towing tank tests and it is playing an increasingly important role in the early stages of ship design. Reynolds-Averaged Navier-Stokes (RANS) based simulations are computationally affordable for engineers and are able to capture the integral flow fields with reasonable accuracy. Yet, the physical complexity of multiphase and multiscale naval ship simulations represents a significant modelling challenge that is the focus of the present work. More specifically, we assess the modelling and numerical errors on canonical multiphysics sub-problems which characterize naval simulations. The modelling and numerical error assessment on these canonical sub-problem provides a guideline for the full scale ship simulations. The best way to fully isolate the modelling requirements for these multiphase problems is to decompose the complex problem into its simplest building blocks. Although these canonical problems represent an abstraction of the complex engineering flow, their simplicity is what makes them attractive to assess the numerical modelling errors. These simple flows either admit an analytical solution or have been studied experimentally. More complex flow/geometries, which may have the benefit of being more relevant, often do not provide the same level of scientific merit. The cases we selected represent the most fundamental flows in which we are able to isolate: surface wave propagation, liquid run-up height on a solid body, flow separation near geometric discontinuities, and wake dynamics.

Although many works provide grid converged results, to the authors' knowledge, no work has sought to comprehensively understand the modelling and numerical requirements on the naval flow prediction. Many physical phenomena influence the prediction of naval ship maneuvering forces such as the surface wave dispersion, flow separation, solid-liquid interface etc. Due to the complex geometry of a commercial or military ship it is often very difficult to investigate the numerical challenges in modelling these complex physical problems. In this work, we started with an analytical ship geometry (Wigley hull) at different drift cases and identify the main modelling and numerical errors. Then we decomposed the complex multiphase ship simulation into several canonical sub-problems to investigate the influence of mesh resolution and modelling assumptions on resistance prediction. The main outcomes of this work can be summarized as follows:

From the sub-cases test, we observed that:

- Poor mesh resolution/setup leads to errors in:
  - surface wave propagation (important)
  - drag coefficient in single phase flow (minor)
  - air-water run-up height on hull (modest)
  - drag coefficient with flow separation (important)

- For accurate energy dissipation through surface wave propagation (Stokes $1^{st}$ order wave), 40 points per wavelength and 6 points per wave height are needed. For simple geometries (single phase cylinder), the effect of grid alignment is minimal on the drag coefficient. For flow separation/wake modelling (surface-piercing square), large deviations are observed despite high-resolution mesh is used in the simulation.

The consideration of mesh resolutions and modelling requirements to account for the multiphase turbulent flow in ship maneuvering simulation can lead to the improvement in capabilities of prediction in numerical simulation, which is desired in engineering to save development costs. These extensions should be considered for ship engineers.

# Appendix B

# Configuration file geometry STL modification

## B.1 Wigley Hull Geometry STL: wigley-stern.py

```
# %load http://matplotlib.org/mpl_examples/mplot3d/trisurf3d_demo2.py
#Lpp=1, B=0.1, T=0.0625, ref:
#URANS and DES analysis for a Wigley hull at extreme drift angles
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.tri as mtri

# u, v are parameterisation variables
u = (np.linspace(-0.5,0.5, endpoint=True, num=200) * np.ones((200, 1))).flatten()
v = np.repeat(np.linspace(-0.0625, 0.04, endpoint=True, num=200), repeats=200).flatten()

print(type(u),type(v))
# This is the Mobius mapping, taking a (u, v) pair and returning an x, y, z
# triple

L=1
B=0.1
T=0.0625

x = u
```

```
z = v
y = 0.5*B*(1-np.power(2*u/L,2))*(1-np.power(v/T,2))*(v<0)+0.5*B*(1-np.power(2*u/L,2))*(v>=0)

print(type(x),type(y))


# Triangulate parameter space to determine the triangles
tri = mtri.Triangulation(u, v)


from stl import mesh

data = np.zeros(len(tri.triangles), dtype=mesh.Mesh.dtype)
mobius_mesh = mesh.Mesh(data, remove_empty_areas=False)
mobius_mesh.x[:] = x[tri.triangles]
mobius_mesh.y[:] = y[tri.triangles]
mobius_mesh.z[:] = z[tri.triangles]
mobius_mesh.save('wigley.stl')
```

## B.2 Rotate Geometry STL: rotateSTL.py

```
#!/usr/bin/python
# translateRotateSTL
# Gery Casiez - 2015
# https://github.com/casiez/translateRotateSTL/blob/master/translateRotateSTL.py
import sys, getopt
import numpy
from stl import mesh
import math
import pdb

def usage():
print 'translateRotateSTL.py -i <inputfile> -o <outputfile> -t <x>,<y>,<z> -r <x>,<y>,<z>'
print 'Example: translateRotateSTL.py -i model.stl -t 0,50,0 -r 0,0,45'
print 'Rotations are defined in degrees'

def main(argv):
inputfile = ''
outputfile = ''
```

```
tx = 0
ty = 0
tz = 0
rx = 0
ry = 0
rz = 0

try:
opts, args = getopt.getopt(argv,"hi:o:t:r:")
except getopt.GetoptError:
usage()
sys.exit(2)
for opt, arg in opts:
if opt == '-h':
usage()
sys.exit()
elif opt == '-i':
inputfile = arg
elif opt == '-o':
outputfile = arg
elif opt == '-t':
xyz = arg.split(',')
tx = float(xyz[0])
ty = float(xyz[1])
tz = float(xyz[2])
elif opt == '-r':
xyz = arg.split(',')
rx = float(xyz[0])
ry = float(xyz[1])
rz = float(xyz[2])

your_mesh = mesh.Mesh.from_file(inputfile)
#        pdb.set_trace()
if outputfile == '':
outputfile = inputfile[:-4] + '-modified.stl'
#translation
for i in range(0, len(your_mesh.vectors)):
for j in range(0, len(your_mesh.vectors[i])):
your_mesh.vectors[i][j] = your_mesh.vectors[i][j] + numpy.array([tx, ty, tz])
#rotation
```

```
if (rx != 0.0):
your_mesh.rotate([1.0, 0.0, 0.0], math.radians(rx))

if (ry != 0.0):
your_mesh.rotate([0.0, 1.0, 0.0], math.radians(ry))

if (rz != 0.0):
your_mesh.rotate([0.0, 0.0, 1.0], math.radians(rz))

your_mesh.save(outputfile)

if __name__ == "__main__":
if (len(sys.argv) == 1):
usage()
exit(1)
main(sys.argv[1:])
```

## B.3 Scale Geometry STL: scaleSTL.c

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<math.h>

char partName[80];//the file name
struct Normal
{
float i;
float j;
float k;
};

struct Vertrex
{
float x;
float y;
float z;
};
```

```
struct Face
{
struct Normal normal;
struct Vertrex vertex1;
struct Vertrex vertex2;
struct Vertrex vertex3;
};

//the node of linked list to receive the triangular geometry
struct Faces
{
struct Face data;
struct Faces *pointer;
};

int main(int argc,char **argv)
{
FILE *fp;
char fileName[64];
struct Faces *faces = (struct Faces*)malloc(sizeof(struct Faces));//List header of the linked
struct Faces *p = faces;
struct Faces *q;
float ratio;
ratio=0.001;
// printf("please type the input file name\n");
// gets_s(fileName);  //fileName=scanf from the keyboard
// fileName="tmodel.stl";
fp = fopen("DTM.stl", "r");


if (fp)
{
char str_a[24];
char str_b[24];
char str[24];
// move the file pointer, to jump over the string "solid", move the pointer to the position
fscanf(fp, "%s", str_a);
fscanf(fp, "%s", str_a);
fscanf(fp, "%s", str_a);
fscanf(fp, "%s", str_a);
```

```
// printf("%s\n", str_a);
//fscanf(fp, "%s", partName);//read the name of file,first line in the file
// printf("%s\n", partName);
// read the perpendicular direction and the vertex data
while (!feof(fp))
{
p->pointer= (struct Faces*)malloc(sizeof(struct Faces));//create new node to receive data
q = p;
p = p->pointer;
fscanf(fp, "%s%s\n", str_a,str_b);//jump the pointer over \facetnormal"point to the perpendic

// printf("%s %s\n", str_a,str_b);
//read the perpendicular direction data
fscanf(fp, "%f%f%f", &p->data.normal.i, &p->data.normal.j, &p->data.normal.k);
// printf("%f,%f,%f\n", p->data.normal.i, p->data.normal.j, p->data.normal.k);
//return(0);

fscanf(fp, "%s%s\n", str_a,str_b);//jump the pointer over \outer loop"
fscanf(fp, "%s", str);//jump the pointer over\vertex"point to the data of vertex 1
fscanf(fp, "%f%f%f", &p->data.vertex1.x, &p->data.vertex1.y, &p->data.vertex1.z);

fscanf(fp, "%s", str);////jump the pointer over\vertex"point to the data of vertex 2
fscanf(fp, "%f%f%f", &p->data.vertex2.x, &p->data.vertex2.y, &p->data.vertex2.z);
fscanf(fp, "%s", str);////jump the pointer over\vertex"point to the data of vertex 3
fscanf(fp, "%f%f%f", &p->data.vertex3.x, &p->data.vertex3.y, &p->data.vertex3.z);
fscanf(fp, "%s", str);////jump the pointer over \endloop"
fscanf(fp, "%s", str);//jump the pointer over \endfacet\
}
free(q->pointer);//since there is a string "endsolid .....\ at the end of filethe additional
q->pointer = NULL;//deallocate the last node data, then the pointer point to NULL, in order t
}
else
{
printf("failed to open the file");
}
fclose(fp);
// output file
p = faces->pointer;  //get the pointer for the second node in the linked list
fp=fopen("DTM-scaled.stl","w");
fprintf(fp,"%s\n","solid Created by Gmsh");
```

```
while (p!=NULL)
{
//output the linked list
// printf("perpendicular direction\n");
fprintf(fp,"%s %f  %f  %f\n","facet normal", p->data.normal.i, p->data.normal.j, p->data.norm
  fprintf(fp,"%s\n","outer loop");
fprintf(fp,"%s %f  %f  %f\n"," vertex", p->data.vertex1.x*ratio, p->data.vertex1.y*ratio, p->
fprintf(fp,"%s %f  %f  %f\n"," vertex", p->data.vertex2.x*ratio, p->data.vertex2.y*ratio, p->
fprintf(fp,"%s %f  %f  %f\n"," vertex", p->data.vertex3.x*ratio, p->data.vertex3.y*ratio, p->
  fprintf(fp,"%s\n","endloop");
fprintf(fp,"%s\n","endfacet");

p = p->pointer;
}
fprintf(fp,"%s\n","endsolid Created by Gmsh");
fclose(fp);
return 0;
//--------------------------------------------------------
p = faces->pointer;//get the pointer for the second node in the linked list
while (p!=NULL)
{
//output the linked list
printf("perpendicular direction\n");
printf("%f  %f  %f\n", p->data.normal.i, p->data.normal.j, p->data.normal.k);
printf("vertex\n");
printf("%f  %f  %f\n", p->data.vertex1.x*ratio, p->data.vertex1.y*ratio, p->data.vertex1.z*ra
printf("%f  %f  %f\n", p->data.vertex2.x*ratio, p->data.vertex2.y*ratio, p->data.vertex2.z*ra
printf("%f  %f  %f\n", p->data.vertex3.x*ratio, p->data.vertex3.y*ratio, p->data.vertex3.z*ra
p = p->pointer;
}
return 0;
}
```

## B.4 Create blockMeshDict File: blockMeshDict.py

```
#change from a tutorial of NACAFoil
#https://github.com/petebachant/NACAFoil-OpenFOAM/blob/master/scripts/blockmeshdict.py
```

```
from __future__ import division, print_function
import argparse
import numpy as np
from numpy import linspace, zeros, ones, sin, cos, arctan, pi
import os
#input a1,q,Sn
#output a_n,n
def refine(a_1,q,Sn):
    a_n=(a_1-(1.0-q)*Sn)/q
    n=int(np.log(a_n/a_1)/np.log(q)+1)
    return(a_n,n)
# half model of wigley by wan de cheng, drift0
#x streamwise,
#y perpendicular direction
#z gravity direction
#input parameters
Re=4.67e6
Fr=0.408
rho=998.8
g=9.8

Lpp=2.5
waterline=0.0


beta_deg=10
U=Fr*np.sqrt(g*Lpp)
nu=U*Lpp/Re

beta = np.deg2rad(beta_deg)
Ux=-U*cos(beta)
Uy=-U*sin(beta)
print("U=%f, Ux=%f, Uy=%f" %(U,Ux,Uy))
#ship  region
#x(-1.25,1.25)
#y(-0.125,0.125)
#z(-0.156,0.1)
ship_x_max=1.25
ship_y_max=0.125
ship_z_max=0.1
```

```
ship_x_min=-1.25
ship_y_min=-0.125
ship_z_min=-0.156


xmax=1.5*Lpp
xmin=-3.5*Lpp
ymax=2.0*Lpp
ymin=-0*Lpp
zmax=0.0399*Lpp
zmin=-Lpp

Lx=xmax-xmin
Ly=ymax-ymin
Lz=zmax-zmin
#-----------------------#
#multigrading
#-----------------------#
stretch_ratio=1.2
dz_min=0.0528
dx_min=dz_min
dy_min=dz_min
#z direction
refine_z_min=ship_z_min
refine_z_max=zmax
z_uniform=zmax-refine_z_min
nz_refine=int(z_uniform/dz_min)

#z_up=zmax-refine_z_max
z_down=refine_z_min-zmin
dz_down,nz_down=refine(dz_min,stretch_ratio,z_down)
ratio_z_down=dz_min/dz_down

#x direction
refine_x_min=-0.5*Lpp
refine_x_max= 0.5*Lpp
x_uniform=refine_x_max-refine_x_min
nx_refine=int(x_uniform/dx_min)
```

```
x_up=xmax-refine_x_max
x_down=-(xmin-refine_x_min)
dx_up,nx_up=refine(dx_min,stretch_ratio,x_up)
dx_down,nx_down=refine(dx_min,stretch_ratio,x_down)
ratio_x_up=dx_up/dx_min
ratio_x_down=dx_min/dx_down

#y direction
refine_y_max=0.1*Lpp
y_uniform=refine_y_max-ymin
ny_refine=int(refine_y_max/dy_min)

y_up=ymax-refine_y_max
dy_up,ny_up=refine(dy_min,stretch_ratio,y_up)
ratio_y_up=dy_up/dy_min



#-----------------------#
#total mesh nx,ny,nz
#-----------------------#
nx=nx_refine+nx_up+nx_down
ny=ny_refine+ny_up
nz=nz_refine+nz_down
print("nx=%i,ny=%i,nz=%i,nxnynz=%i" %(nx,ny,nz,nx*ny*nz))
print("zmax=",zmax,"zmin=",zmin)


scale = 1            # Scaling factor



vertices = zeros((4, 3))

vertices[0, :] = [xmin,ymin,zmin]
vertices[1, :] = [xmax,ymin,zmin]
vertices[2, :] = [xmax,ymax,zmin]
vertices[3, :] = [xmin,ymax,zmin]

# Create vertices for other side (negative y-axis)
```

```
vertices2 = vertices.copy()
vertices2[:, 2] = zmax
vertices = np.vstack((vertices, vertices2))

f = open("SnappyHexMeshDict", "w")
f.write("refinementBox     \n")
f.write("{ \n")
f.write("   type    searchableBox; \n")
f.write("   min     (%f %f %f); \n"%(xmin,ymin,refine_z_min))
f.write("   max     (%f %f %f); \n"%(xmax,ymax,refine_z_max))
f.write("} \n\n\n")

f.write("refinementWaterline     \n")
f.write("{ \n")
f.write("   type    searchableBox; \n")
f.write("   min     (%f %f %f); \n"%(xmin,ymin,-0.010*Lpp))
f.write("   max     (%f %f %f); \n"%(xmax,ymax, 0.015*Lpp))
f.write("} \n\n\n")

f.write("refinementRegions \n")
f.write("{ \n")

#f.write("  hull \n")
#f.write("  { \n")
#f.write("   mode distance;\n")
#f.write("   levels ((%f 4) (%f 3) (%f 2) (%f 1) );\n"%(dx_min/4,dx_min*0.75,dx_min*1.75,dx_m
#f.write("  } \n")

f.write("  refinementBox \n")
f.write("  { \n")
f.write("   mode inside;\n")
f.write("   levels ((1e15 2));\n")
f.write("  } \n")

f.write("  refinementWaterline \n")
f.write("  { \n")
f.write("   mode inside;\n")
f.write("   levels ((1e15 3));\n")
f.write("  } \n")
```

```
f.write("} \n\n\n")

f.write("    locationInMesh (%f %f %f);"%(xmax-0.01,ymax-0.01,zmin+0.01))
f.close()

# Open file
f = open("blockMeshDict", "w")
# Write file
f.write("/*--------------------------------*- C++ -*----------------------------------*\\ \n"
f.write("| =========                 |                                                 | \n")
f.write("| \\\\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           | \n
f.write("|  \\\\    /   O peration     | Version:  1712                                 | \n"
f.write("|   \\\\  /    A nd           | Web:      www.OpenFOAM.com                      | \n
f.write("|    \\\\/     M anipulation  |                                                 | \n
f.write("\\*---------------------------------------------------------------------------*/ \n"
f.write("FoamFile                                                                        \n")
f.write("{                                                                               \n")
f.write("    version     2.0;                                                            \n")
f.write("    format      ascii;                                                          \n")
f.write("    class       dictionary;                                                     \n")
f.write("    object      blockMeshDict;                                                  \n")
f.write("}                                                                               \n")
f.write("// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * // \n")
f.write("\n")
f.write("convertToMeters %f; \n" % scale)
f.write("\n")
f.write("//stretch ratio=%f,dx_min=%f\n"%(stretch_ratio,dx_min))
f.write("// nx1= %i, ny=%i, nz=%i\n" %(nx,ny,nz))

f.write("vertices \n")
f.write("( \n")
for vertex in vertices:
    f.write("    (%f %f %f)\n" % tuple(vertex))
f.write("); \n")
f.write("\n")

f.write("blocks \n")
f.write("( \n")
#x ridius direction
#y tangential direction
```

```
#inner cylinder block number: 0,1, 5,6, 10,11, 15,16
#nx2: number of points r ring, total stretch ratio=2
#4*ny1: number of points theta ring, uniform mesh
f.write("hex (0 1 2 3 4 5 6 7)   (%i %i %i) \n" %(nx,ny,nz))
f.write("simpleGrading \n")
f.write("( \n")

f.write("       (     \n")
f.write("             (%f %f %f)\n" %(x_down,   nx_down,  ratio_x_down))
f.write("             (%f %f %f)\n" %(x_uniform,nx_refine,1))
f.write("             (%f %f %f)\n" %(x_up,     nx_up,    ratio_x_up))
f.write("       )     \n")

f.write("       (     \n")
#f.write("             (%f %f %f)\n" %(x_down,   nx_down,  ratio_x_down))
f.write("             (%f %f %f)\n" %(y_uniform,ny_refine,1))
f.write("             (%f %f %f)\n" %(y_up,     ny_up,    ratio_y_up))
f.write("       )     \n")

f.write("       (     \n")
f.write("             (%f %f %f)\n" %(z_down,   nz_down,  ratio_z_down))
f.write("             (%f %f %f)\n" %(z_uniform,nz_refine,1))
#f.write("             (%f %f %f)\n" %(z_up,     nz_up,    ratio_z_up))
f.write("       )     \n")

f.write(") \n")

f.write("); \n")
f.write("\n")

f.write("edges \n")
f.write("( \n")
f.write("); \n")

f.write("\n")
f.write("boundary \n")
f.write("( \n")

f.write("atmosphere \n")
f.write("        { \n")
```

```
f.write("            type patch;\n")
f.write("            faces \n")
f.write("            ( \n")
f.write("              (4 5 6 7) \n")
f.write("            );\n")
f.write("        } \n")
f.write("\n")

f.write("inlet \n")
f.write("        { \n")
f.write("            type patch;\n")
f.write("            faces \n")
f.write("            ( \n")
f.write("              (1 2 6 5) \n")
f.write("            );\n")
f.write("        } \n")
f.write("\n")

f.write("outlet \n")
f.write("        { \n")
f.write("            type patch;\n")
f.write("            faces \n")
f.write("            ( \n")
f.write("              (0 4 7 3) \n")
f.write("            );\n")
f.write("        } \n")
f.write("\n")

f.write("bottom \n")
f.write("        { \n")
f.write("            type symmetryPlane;\n")
f.write("            faces \n")
f.write("            ( \n")
f.write("              (0 3 2 1) \n")
f.write("            );\n")
f.write("        } \n")
f.write("\n")

f.write("midPlane \n")
f.write("        { \n")
```

```
f.write("            type symmetryPlane;\n")
f.write("            faces \n")
f.write("            ( \n")
f.write("              (0 1 5 4) \n")
f.write("            );\n")
f.write("        } \n")
f.write("\n")

f.write("side \n")
f.write("        { \n")
f.write("            type patch;\n")
f.write("            faces \n")
f.write("            ( \n")
f.write("              (3 7 6 2) \n")
f.write("            );\n")
f.write("        } \n")
f.write("\n")

f.write(" );\n")
f.write(" \n")
f.write("mergePatchPairs \n")
f.write("( \n")
f.write("); \n")
f.write(" \n")
f.write("// ************************************************************************* // \n")
f.close()
```