Verification and Validation Study of OpenFOAM on the Generic Prismatic Planing Hull Form

Jiahui Li

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Ocean Engineering

Stefano Brizzolara, Chair

Eric G Paterson

Alan J Brown

May 6, 2019

Blacksburg, Virginia

Keywords: Planing Hull, GPPH, OpenFOAM, URANS

Verification and Validation Study of OpenFOAM on the Generic Prismatic Planing Hull Form

Jiahui Li

(ABSTRACT)

In this research, hydrodynamic analysis, verification and validation will be performed on Generic Prismatic Planing Hull (GPPH) using OpenFOAM v1806 solver *interFoam*. The numerical simulation will be compared with the experimental result, which is a new set of high-quality experimental tests performed on a large model of a high deadrise prismatic planing hull with flat of chine, tested from pre-planing to fully planing regimes.

Firstly, the mesh convergence study and Verification and Validation (V&V) study are performed on the basis of fixed attitude simulations. Three grids are chosen and used to perform the free attitude simulations at the highest speed. Then, mesh convergence study is conducted for the results of highest speed free body simulations, which helps us to choose two grids for other speeds simulations. By performing free attitude simulations using two grids, resistance, heave, trim angle, wetted chine length, and wetted keel length are calculated and compared at seven different tested speeds. Computational Fluid Dynamics (CFD) results analysis regards pressure distribution on the bottom of the hull and in particular areas of interest (flat of chine, spray area, etc.), friction coefficient and volume fraction of fluid in areas where the free surface undergoes violent deformations (overturning wave at the chine and in the wake, spray jet development area). Different algorithms for dynamic mesh simulation and their effect on the quality of CFD predictions are also investigated.

Verification and Validation Study of OpenFOAM on the Generic Prismatic Planing Hull Form

Jiahui Li

(GENERAL AUDIENCE ABSTRACT)

The paper presents the first series of results obtained in an ongoing validation and verification study of inter-dynamic OpenFOAM solver framework on a new set of high quality experimental tests performed on a large (2.4m long) generic planing hull model (GPPH) with high deadrise (18deg), from the pre-planing (Fn_∇=2.6) to fully planing (Fn_∇=5.7) regimes. This test case is a good benchmark for the free surface capturing model implemented in OpenFOAM which is based on a rather simple transport equation for an additional scalar field that defines the fraction of water in each cell of the computational mesh.

This model, in spite of its simplicity, seems capable of reproducing complex violent free surface flows such as that observed in planing hulls, that includes jet spray forming on the bottom and detaching from the chine of the planing hull and overturning waves off the wet chine region, with some nuances.

The dependence of the flow solution on the mesh quality is presented and discussed. Practical indication of the level of uncertainty of CFD models for the prediction of the calm water hydrodynamics of the GPPH is given at the highest simulated speed using both fixed and free attitude simulation solutions. Predictions are then extended to the whole speed range, including resistance components, dynamic trim, heave, wetted chine length, and wetted keel length.The effect due to algorithms is also discussed by modifying the settings in wall functions and solvers for the improvements of future simulation.

Dedication

I dedicate my thesis work to my family and many friends. A special feeling of gratitude to my loving parents, Zhangyan and Baohong, whose words of encouragement and push for tenacity ring in my ears. I dedicate my thesis work and special thanks to my friend Rachit Salian for helping me with my simulations setup and debugging the codes. I also dedicate this dissertation to my many friends supported me throughout the process. I will always appreciate all they have done, especially Zhaokuan Lu, Zhongshu Ren, Yang Zeng and Nan Si for helping me develop my technology skills, Guangyao Wang for teaching me writing and format skills, Ben Zhao and Shanling Xu for helping me with course problems. I dedicate this work to my roommate Yuting Feng and Xinyu Yan for giving me support during my Master's program. My friends Zhao Li and Xu Wang for staying with me when I'm depressed. Both of you have been my best cheerleaders.

Acknowledgments

I wish to thank my advisor and committee members who were more than generous with their expertise and precious time. A special thanks to Dr. Stefano Brizzolara, my advisor and committee chairman for his countless hours of reflecting, reading, encouraging, and most of all patience throughout the entire process. Thank you Dr. Eric Paterson and Dr. Alan Brown for agreeing to serve on my committee. I would like to acknowledge and thank Luca Bonfiglio for helping me in this study. I want to thank my school division for allowing me to conduct my research and providing any assistance requested. Special thanks go to the staff members for their continued support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recent advances in modeling complex hydrodynamic phenomena using Computational Fluid Dynamic (CFD) are leading to a radical transformation in the naval hydrodynamic field. For conventional displacement hulls, the accuracy and reliability of numerical simulations have been extensively validated on the model and full-scale experiments performed on different hull forms [11] and are considered a mature tool for design and optimization purposes.

Planing crafts are different from conventional displacement hulls, the weight of planing craft when advancing in planing regime is predominantly supported by hydrodynamic lift, while conventional displacement hulls are supported mainly by hydrostatic lift, i.e., buoyancy force. In the case of planing crafts, the free surface flow regime changes completely with respect to displacement hulls. The free waves' patterns loose importance while the accurate estimation of the dynamic pressure distribution on the hull due to a water entry like flow becomes essential. Figure 1.1 shows planing hull bottom and waterline intersection.

In fact, the dynamic equilibrium of the vessel is primarily influenced by the dynamic forces acting on the hull (mostly dynamic lift and not buoyancy) which in turn depend on the (dynamic) wetted area of the planing craft. Dynamic pressure, forces and moments, wetted area are all highly variable with speed and center of gravity position. Hence the need to capture the specific flow patterns of planing surfaces: pressure peaks around the stagnation lines and relative flow pile up; spray formation and development, flow separation and from the transom and wet chines. The running attitude of planning craft strongly depends on the

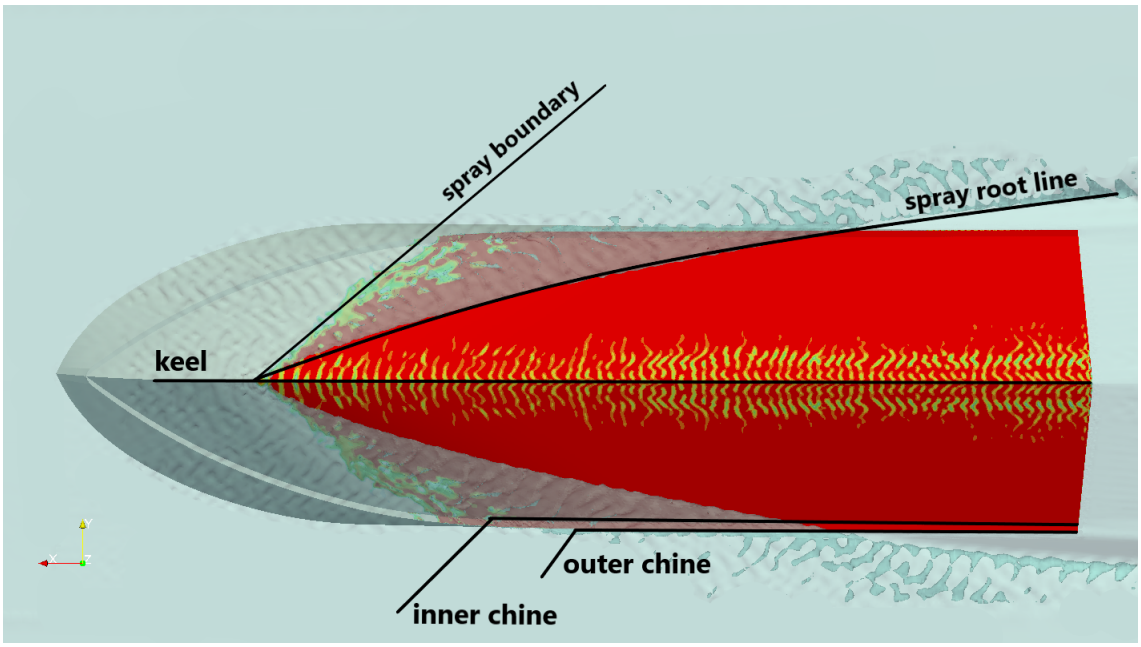


Figure 1.1: Planing Hull Bottom and Waterline Intersection

mesh resolution close to hull regions interested by jet spray formations such as the fore body (spray root line), chine and transom where flow separation phenomena occur.

Validation and Verification (V&V) studies for high speed planing crafts are very less in comparison with larger displacement hull forms and often lack of high-quality experimental reference data on modern hull forms. Previous validation studies have considered old systematic planing hulls series [21] [20] [4]. Only a few studies have considered modern planing hull shapes for practical seagoing application [2] [19].

The study presented in this thesis focuses on a V&V of numerical models based on the OpenFOAM CFD solver framework on a new set of high quality experimental tests performed on a large (2.414m long) model of a high deadrise (18 deg) prismatic planing hull with flat of chine model [12], tested from pre-planing (Fn_{∇}=2.6) to fully planing (Fn_{∇}=5.7) regimes. This hull has been the object of a recent workshop organized by the American Society of Naval Engineers, for which different participants presented predictions of resistance, trim

and dynamic sinkage, using different CFD methods. The goal was to assess the overall level of uncertainty of different hydrodynamic codes over a high-quality set of experimental results on a notional deep-V planing hull shape, namely the Generic Prismatic Planing Hull (GPPH).

One of the most relevant intricacies in the fluid dynamic predictions around planing hulls is the correct modeling of the non-linear free surface dynamics [2] even in presence of steps and ventilation [19]. The significant pressure field on the hull surface due to high Froude number and the sharp edges characterizing both the chine and the transom of planing craft induce jet sprays that need to be correctly modeled to achieve high accuracy in the prediction of the hydrodynamic forces exerted on the hull.

Under this perspective, this study intends to verify the performance of the free surface capturing model implemented in OpenFOAM which does not require any boundary condition at the free surface and is based on a rather simple transport equation for an additional scalar field that defines the fraction of water in each cell of the computational mesh. This model, in spite of its simplicity, seems capable of reproducing complex violent free surface flows, such as that observed in planing hulls, that includes jet spray forming on the bottom and detaching from the chine of the planing hull and overturning waves off the wet chine region. The Volume Of Fluid technique that requires the prediction of violent surface flows needs a good quality mesh. Typical problems noted by different researchers [20] [4] are numerical (non-physical) diffusion of air under the hull. Techniques to obtain high-quality unstructured meshes, specifically conceived to increase the mesh resolution close to the spray region is discussed in the paper, as well as tuning of the numerical parameters in the VOF transport equation to reduce this effect.

Savitsky Short Method is also used in this study for the assessment of numerical simulation results. Savitsky Short Method uses empirical planing equations which are functions of

speed, trim angle, deadrise angle and loading to describe the lift, drag, wetted area, center of pressure, and porpoising stability limits of planing surfaces [22]. In this study, Savitsky Method is applied in calculating total resistance, friction drag, pressure, trim angle, heave, wetted chine length and wetted keel length.

Practical indication of the level of uncertainty of CFD models for the prediction of the calm water hydrodynamics of the GPPH is given in the conclusion and overall accuracy well compares with the state of the art for displacement hull forms.

Chapter 2

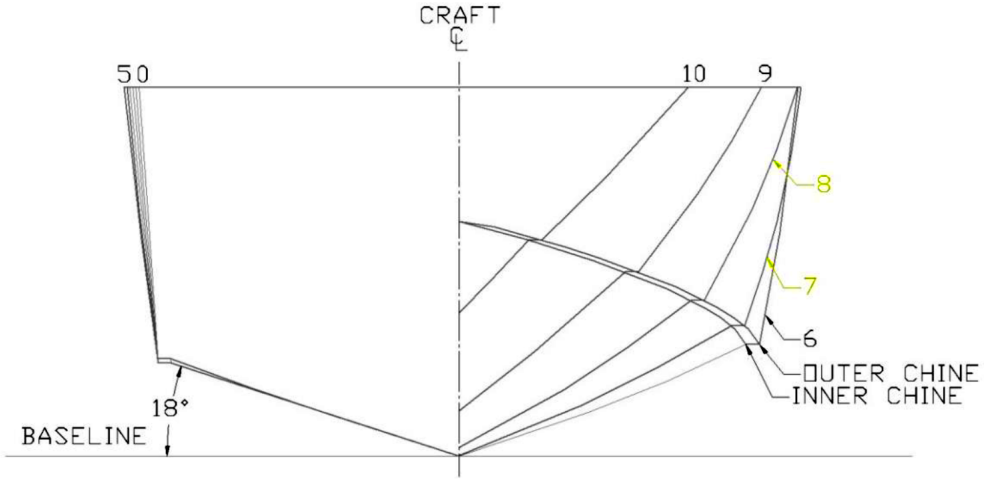
Reference Experimental Data

The test model of Generic Prismatic Planing Hull is designed by the Naval Surface Warfare Center as an assessment for CFD simulation tools. This model uses a modern planing hull bow shape and a prismatic hull aft of the bow, thus to have the physics and response to waves of a typical sea-going planing hull. The hull has a double chine (inner and outer in Figure 2.1a) with a small flat surface between the two acting as a spray deflector [12]. The test data used in this paper is from the November 2015 report [12] performed at Naval Surface Warfare Center Carderock Division (NSWCCD). Figure 2.1b shows profile view of the model. And a picture of the model is shown as Figure 2.1c. The tested GPPH model characteristics are summarized in Table 2.1.

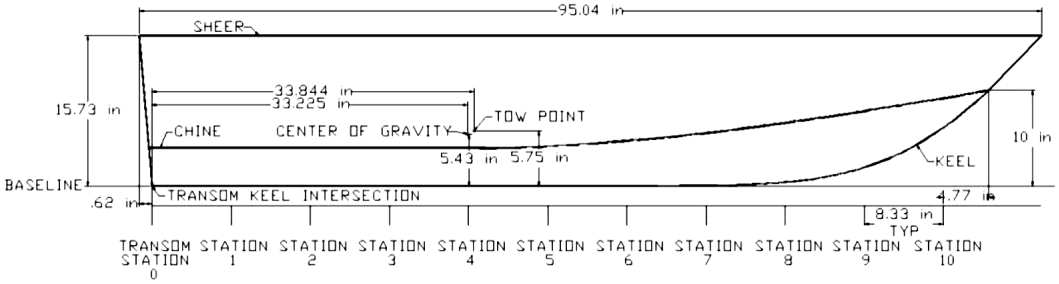
Table 2.1: Tested GPPH Characteristics

Parameter	Value
Length Between Particulars, L_{pp}	2.414 (m)
Maximum Projected Chine Beam, B_{PX}	0.6274 (m)
Deadrise Angle to Outer Chine, β	17.20 (degree)
Displacemet, ∇	101.5 (kg)
Longitudinal Center of Gravity, LCG	0.8439 (m)
Vertical Center of Gravity, VCG	0.1379 (m)
Longitudinal Center of Tow Point, LTP	0.8596 (m)
Vertical Center of Tow Point, VTP	0.1461 (m)
Hydrostatic Draft at Transom Above Baseline, $T_{Transom}$	0.1476 (m)
Hydrostatic Trim Angle from Baseline (+bow↑)	0.1270 (degree)

The model is only allowed to be free to heave and pitch while it was restrained in the surge, sway, roll, and yaw by the experimental setup (Figure 2.2). A pitch gimbal is used to attach



(a) GPPH Body Plan



(b) Model Profile

(c) Model Picture

Figure 2.1: Test Model Body Plan and Profile (adapted from [12])

the model to the tow post, and the pivot point of the pitch gimbal is designed to be the tow point. The tow point is placed at about 16 mm ahead of the center of gravity (CoG) and 8 mm higher than CoG as shown in Figure 2.1b. The heave and trim are all measured from the hydrostatic state with three string potentiometers at the bow, tow point and stern (Figure 2.3). The heave values are taken from the potentiometer at tow point while trim angles are calculated from the bow and stern potentiometers' measurements via trigonometry calculations [12].

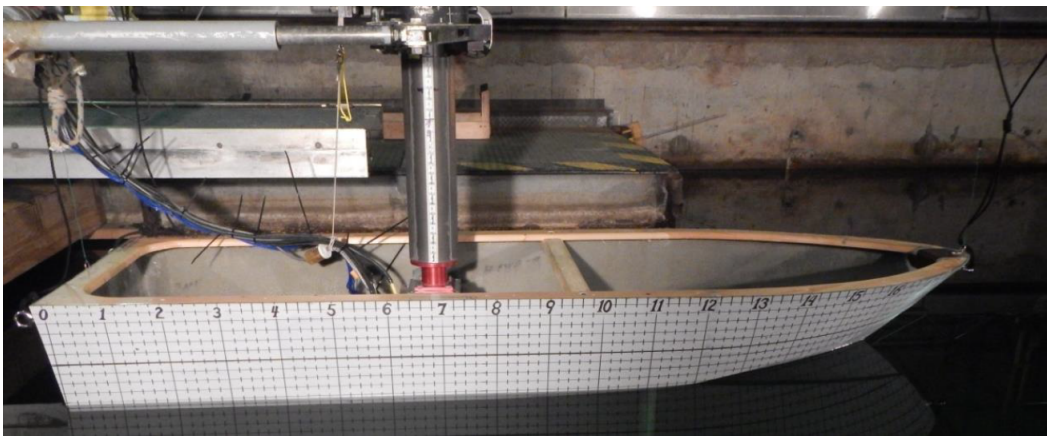


Figure 2.2: Experimental Setup (adapted from [12])

The model was tested under 7 different speeds from pre-planing to fully planing regimes. Table 2.2 shows the experimental results from NSWCCD at the 7 tested speeds, and the simulations in this study follow all these tested conditions. It is rare to find such large model size that adopted for the tests of the GPPH (length 2.414 meters). Additionally, the speed range arrives up to a very fast planing regime, corresponding to volumetric Froude number 5.72. Therefore, this set of experimental data is very good to be used as benchmark data for the assessment of numerical simulations.

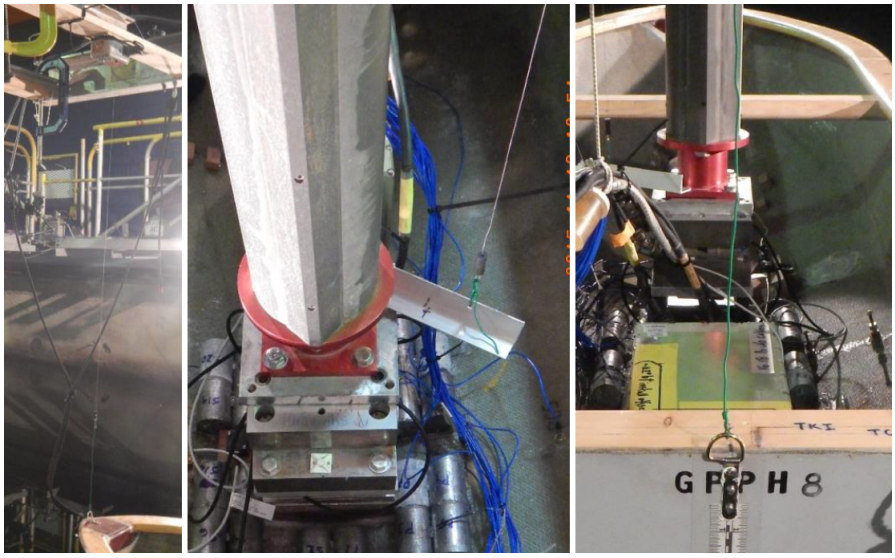


Figure 2.3: Potentiometers at Bow, Tow Point and Stern (adapted from [12])

Table 2.2: Experimental Results

V_m $[m/s]$	F_∇	R_{Tm} $[N]$	$Trim$ $[deg]$	CG rise $[cm]$	Wetted Chine Length $[cm]$	Wetted Keel Length $[cm]$
12.24	5.72	270	2.17	9.60	44.5	173
11.11	5.19	235	2.54	9.47	55.9	171
10.03	4.69	207	2.96	9.27	67.3	169
9.00	4.21	184	3.42	9.06	76.2	166
7.61	3.56	165	4.22	8.49	88.9	164
6.69	3.13	160	4.88	8.17	97.8	163
5.56	2.60	161	5.81	7.45	109	165

Chapter 3

RANS Solver and Numerical Model

This study uses OpenFOAM (Open source Field Operation And Manipulation) as the numerical simulation tool. OpenFOAM is a free, open source CFD software developed with C++. Users can not only use the solvers provided by OpenFOAM but also develop customized numerical solvers for different purposes. The standard solvers provided by OpenFOAM include the basic CFD codes like *potentialFoam* for solving potential flow, solvers for incompressible flow and compressible flow, solvers for multiphase flow like *interFoam* used in this study, the direct numerical simulation solver *dnsFoam*, and the other solvers used in many other areas. More information regarding OpenFOAM and its application can be found from *OpenFOAMWiki* web page. In our study, version OpenFOAM v1806 is used for the simulations, and it has been installed on Virginia Tech ARC clusters. *interFoam* solver is used for both fixed attitude and free attitude simulations.

The numerical model proposed in the present study is based on the solution of the Unsteady Reynolds Averaged Navier Stokes (URANS) equations. The system is a set of partial differential non-linear equations in the unknowns of pressure and velocity components. The numerical solution is here obtained using a finite volume technique with the collocated arrangement of variables. In particular, we have used OpenFOAM libraries to numerically solve the non-linear system of Partial Differential Equations (PDEs) describing the fluid dynamic problem on a computational domain discretized through the unstructured grid. The physical problem is described in terms of mass and momentum conservation equations which

are solved in a Cartesian reference frame for the pressure and velocity unknowns:

$$\begin{aligned} &\frac{\partial(\rho u_i)}{\partial x_i} = 0, \\ &\frac{\partial u_i}{\partial t} + \frac{\partial(u_i u_j)}{\partial x_j} = \frac{\partial}{\partial x_j}(\nu \frac{\partial u_i}{\partial x_j}) - \frac{1}{\rho}\frac{\partial p}{\partial x_i} + g_i. \end{aligned} \tag{3.1}$$

Even in model scale, the fluid dynamic problems considered in the present study is characterized by a high Reynolds number ($R_e \in [1e7, 3e7]$) inducing fully turbulent flow. The solution of the turbulent scales would require a very high resolution both in time and space which would significantly increase the computational complexity of the problem. Being the object of the present study the prediction of calm water resistance and dynamic attitude of a given planing hull, the quantities of interest to be predicted through hydrodynamic analysis do not require the solution of the turbulent fluctuations in time and space. For this reason, we approach the fluid dynamic problem by applying a Reynolds averaging technique to the solution of N-S equations. In particular, we have used the $k-\omega$ turbulence model (see [24]) reformulated in the Shear Stress Transport version (SST) by [16, 17]) with the goal to improve the model independence from freestream turbulent intensity through a formulation which blends the k-ϵ model (used in the free stream) and the k-ω model (used in the boundary layer). Freestream values of the turbulent quantities are calculated according to the turbulent intensity and length scale based on the flat plate boundary layer thickness formulation:

$$k = \frac{3}{2}(UI)^2 \qquad \omega = \frac{k}{\nu_T} \qquad \nu_T = \frac{1}{\rho}\sqrt{\frac{3}{2}} UIl \tag{3.2}$$

In equation (3.2) l indicates the turbulent length scale and has been selected according to the boundary layer thickness δ_{99} of a turbulent flow on a flat plate.

$$l = 0.4\delta_{99} \qquad \frac{\delta_{99}}{L} = \frac{0.374}{Re^{\frac{1}{5}}} \tag{3.3}$$

A Volume Of Fluid (VOF) surface capturing technique ([6]) is here used to predict the wave pattern generated by the hull advancing at the free surface. This method is based on the solution of an additional transport equation (3.6) for a scalar function α normalized to 1 and indicating the relative content of air with respect of water at each point of the computational domain. URANS equations are solved for a fluid mixture of water and air whose characteristics are determined on the basis of the scalar function α:

$$\rho = (1-\alpha)\,\rho_W + \alpha\rho_A \tag{3.4}$$

$$\mu = (1-\alpha)\,\mu_W + \alpha\mu_A \tag{3.5}$$

At each time-step the local value of α is determined by the solution of the following equation:

$$\frac{\partial\alpha}{\partial t} + \nabla\cdot(\alpha U) + \nabla\cdot[\alpha\,(1-\alpha)\,U_r] = 0 \tag{3.6}$$

In equation (3.6) the term U_r indicates the relative velocity between water and air and it is here used to steepen the gradient of the VOF scalar function α close to the interface. U_r is tuned through the parameter C_f as follow:

$$U_r = n_f \min\left[C_f\frac{\phi}{S_f}, \max\left(\frac{\phi}{S_f}\right)\right] \tag{3.7}$$

In particular $\nabla\cdot[\alpha\,(1-\alpha)\,U_r]$, also called artificial compression, is effective when $\alpha \neq 0$ and $\alpha \neq 1$ and it is opportunely tuned to increase the compression of the free surface hence reducing the interface smearing often experienced when surface capturing techniques are used to predict free surface flows.

A constant flow velocity is imposed at the inlet according to the specific operating condi-

tions. Neumann boundary condition is used for the dynamic pressure field at the inlet (zero gradient). Constant zero dynamic pressure is enforced at the outlet of the computational domain where a Neumann boundary condition is instead used for the velocity field (zero gradient). No-slip condition is imposed at the hull surface where a zero gradient dynamic pressure condition is used together with a zero velocity field. Top and bottom boundaries have been modeled as slip walls. We exploit the flow and geometric symmetry of the problem and we model only the port-side of the domain, imposing a symmetry boundary condition at the center plane of the planing hull.

The dynamic trim and sinkage of planing hulls is a direct consequence of the hydrodynamic and hydrostatic forces and moments acting on the body when advancing in calm water. These forces depend on the position of the hull with respect to the free surface which in turns depend on the balance of the aforementioned forces and moments. While the hydrodynamic forces of lift and drag converge to a steady state solution, they do not balance hydrostatic forces. For this reason, at each time-step the hull is allowed to trim and sink in order to instantaneously restore the balance between hydrodynamic and hydrostatic forces.

Ship motion is obtained using a dynamic mesh strategy supported by grid morphing algorithm that diffuse boundary motion through internal grid points. Mesh motion is computed from the pressure field on the hull. In particular, at each time-step the position and the velocity of the hull are known as well as the forces and moments acting on the rigid body (which result from pressure and velocity fields integration). We solve the forward dynamic problem using a 6 Degrees of Freedom solver in which surge, sway, roll and yam motions are constrained, while heave and pitch accelerations are computed using a Newmark solver (see [3]). A pseudo-solid Laplace smoothing equation distributes hull motion to the internal grid

points through a diffusivity coefficient Γ following equations (3.8) and (3.9) as in [9] and [8]:

$$\nabla \cdot (\Gamma \nabla V) = 0 \tag{3.8}$$

Once mesh deformation velocity (V) is known at each point, grid displacement is found at each new time step t^n solving a linear motion equation:

$$x_n = x_{n-1} + V\Delta t \tag{3.9}$$

Equation (3.9) describes the deformation of an initially valid mesh (t_{n-1}) into a deformed mesh at t_n. The quality of the grid is verified at each time step to ensure that no faces and no cells were inverted during motion. The solution of the pressure term (p) is here obtained using an algebraic multigrid method (*GAMG* - Geometric Algebraic Multi Grid) with a *DIC* (Diagonal Incomplete Cholesky) smoother. The tolerance has been set up to 5e-8. Velocity U and transported quantities (turbulent kinetic energy k and specific dissipation rate ω terms) have been solved using a smooth solver suitable for asymmetric matrices with a symmetric Gauss Seidel smoother with a tolerance of 1e-7.

Volume of fluid term α, has been solved with a smooth solver used also for other transported quantities, but in this case we have used a Multi-dimensional Limiter for Explicit Solution (MULES) to guarantee boundedness of α. This solver is particularly effective when used in combination with an adjustable time-step strategy that at each time iteration select the time-step in order to satisfy a given constraints on the Courant number, which in a 1-D case is defined as:

$$Co = u\frac{\Delta t}{\Delta x} \tag{3.10}$$

In particular, the unsteady solution of RANS equations is achieved differentiating in time

with an implicit Euler scheme and adjustable time step selected according to the maximum Courant number ($Co < 0.5$ at the free surface and $Co < 1$ everywhere else). Navier-Stokes equations have been solved including gravity effects but dropping the hydrostatic component from the pressure field. The pressure-velocity coupling is here solved using the PISO scheme of Issa [7] with 1 outer iteration and 2 correctors.

Chapter 4

Setup of Numerical Model

4.1 Mesh Generation

To generate the grids, we need the GPPH hull model, the calculation domain and the meshing strategy. The basic idea is starting from base model generated by widely used modeling software Rhino, we translate and rotate the hull to the hydrostatic state, then rotate the model to corresponding speed trim angle from experiment to make the simulation converge fast. In a word, the seven models we are using for the corresponding speeds' simulations are already set to the experiment steady state, respectively. Finally, a triangulated surface mesh represented by surface points saved in a *.stl (stereo-lithography) standard file is a prerequisite for OpenFOAM meshing and simulations. An example of model transformation at highest speed $V = 12.24$ m/s is shown in Figure 4.1.

The first step is to achieve a hydrostatic model. This is a compulsory step because all the heave and trim values in NSWCCD experiments are measured from the hydrostatic state. In the beginning, the keel rests on x axis and the transom is at the origin of the reference frame as shown in the upper left figure in Figure 4.1.

Next, the entire hull model is translated by moving the tow point used in the experiment from its original coordinates (0.8596, 0, 0.14605) in meters to (0, 0, 0). Thus, we have the origin of the reference frame at tow point for the convenience of calculating force and

momentum which were measured by a dynamometer fixed on the tow post in NSWCCD experiments. This is also the location at which the heave motion is measured on the model (potentiometer location).

Following the equilibrium test in the experiment, the hull model was rotated to the static trim angle 0.127 degrees, and the free surface Z_{wl} coordinate is calculated with the draft at the transom which is Z_{wl} = -0.0003752 m. Thus, the static model is achieved following these steps as shown in the upper right figure in Figure 4.1.

The second step is to achieve the seven steady state models corresponding to the seven different experimental speeds, respectively. Starting from the static model, the hull is rotated again to experimental trim angle and the free surface is recalculated according to experimental results. Finally, we get the steady-state models shown as the lower right figure in figure 4.1.

After the transformation, we need to calculate the waterline height coordinate Z_{wl} as well as the coordinate of Center of Gravity (CoG) as summarized in Table 4.1. The tow point is always at (0, 0, 0) and we don't need to change it or re-calculate it. A very important thing to notice is that, by transforming the model, the results for heave and trim from our simulations are not the true values of the heave and trim, they are actually the difference between the experiments. When processing the data, the experimental results are added to the difference to get the true values of simulation results.

Last but not least, the triangulated surface mesh needs to be generated with the steady-state model and exported from Rhino to a *.stl file. This surface mesh file defines all the coordinates of the surface points and it's telling OpenFOAM what kind of geometry it is going to mesh. The *.stl file is imported in OpenFOAM by saving in *constant/triSurface* under case directory. The polygon surface mesh example is also shown in Figure 4.1 as the

first graph on the second row. It's easy to observe that the surface mesh can well present the geometry of the hull.

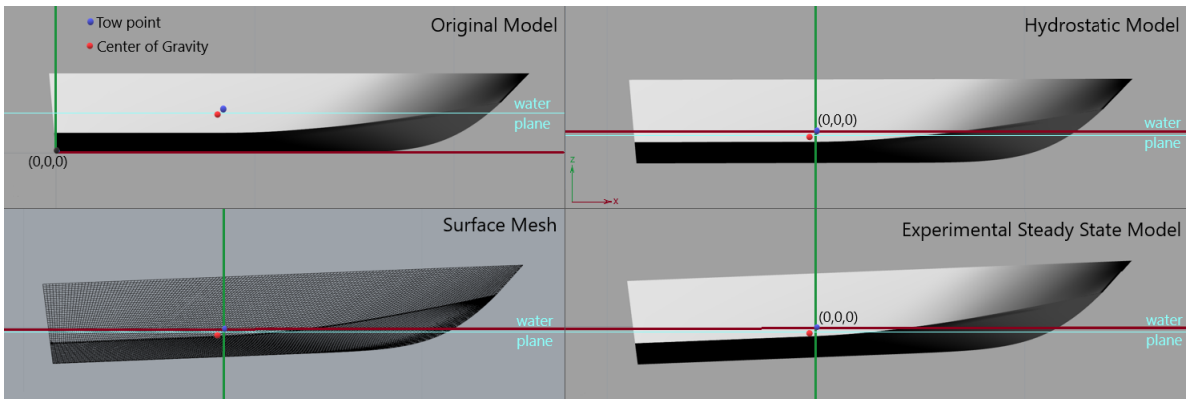


Figure 4.1: Translation and Rotation of the Model

Table 4.1: Coordinates of Center of Gravity (CoG) and Waterline Height at 7 Different Speeds [m]

Speeds [m/s]	Center of Gravity	Z_{wl}
12.24	(-0.01572, 0, -0.008128)	-0.09634
11.11	(-0.01533, 0, -0.008852)	-0.09504
10.03	(-0.01525, 0, -0.008976)	-0.09311
9.00	(-0.01519, 0, -0.009086)	-0.09095
7.61	(-0.01506, 0, -0.009300)	-0.08531
6.69	(-0.01495, 0, -0.009476)	-0.08209
5.56	(-0.01479, 0, -0.009724)	-0.07490

The computational domain covers a total length of 36.21 m, corresponding to 15 ship lengths, a total breadth of 7.5 ship lengths and a total height about 4 ship lengths. With reference to a Cartesian coordinates system (xyz), the tow point of the planing hull is positioned at (0, 0, 0). The longitudinal distance between the tow point and inlet is 12.07 m, corresponding to 5 ship lengths. The bottom of the domain is positioned at 7.242 m, corresponding to 3 ship lengths, while the top is located at 2 m from the free surface at rest (approximately 1 ship length) [15].

The computational domain is divided into 6 vertical regions, each representing a specific

depth-wise layer. With reference to the design waterline (free surface at rest), three layers are used to discretize the water region while the remaining three are employed in the air region. Each depth-wise region extends from inlet to outlet without any geometrical progression in the length-wise (x) or breadth-wise (y) direction. Geometrical progressions are used in the bottom and top layer to coarsen the grid resolution from the internal volume to the top and bottom boundaries of the computational domain.

We first construct a background mesh of structured elements discretizing the 6 depth-wise layers previously described. This structured grid is successively refined in the $x-y$ direction by iteratively selecting sub-domain regions containing the hull and refining them in the length-breadth direction. The refinement in the direction perpendicular to the free surface is performed only using the aforementioned depth-wise layers for the construction of the structured grid. Refinement in the $x-y$ directions is performed using 6 refinement boxes containing the hull. The plane-wise refinement process leads to an unstructured background grid that is then intersected with the hull surface. The hull is represented by means of a triangulated surface mesh serving as input for the generation the unstructured grid. The domain and the background mesh is shown in Figure 4.2.

The hull surface mesh is intersected with the unstructured background grid through an iterative procedure that first increases mesh resolution at the hull-surface/background-grid intersections and then discards cells with centroid inside the hull surface. Cell vertices of this jagged mesh are successively projected on the hull surface and repeatedly extruded in order to obtain layers of prismatic cells suitable for boundary layer flow resolution.

The background grid has been constructed using *blockMesh* and successively refined in the $x-y$ direction using six *topoSet-refineMesh* iterative loops, which is shown in Figure 4.3.The hull-background intersection has been performed using *snappyHexMesh* utility (see [18] for further details).

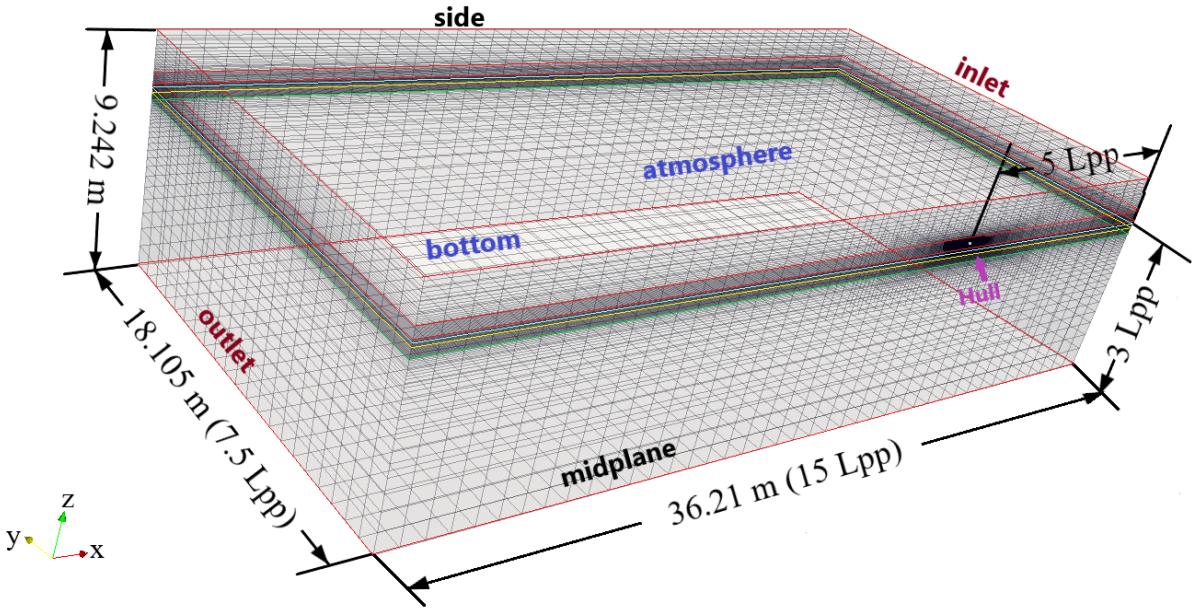


Figure 4.2: Domain and Background Mesh

The overall refinement is shown in Figure 4.4. In order to make the calculation more accurate around the chine and transom area, where the free surface undergoes violent deformations, it is notable that the fine mesh and the finest mesh have one refinement block around the free surface area covering the keel and the chine. This extra refinement block is generated in *snappyHexMesh* utility with the lowest refinement level defined in the *snappyHexMeshDict* file. This box largely refines the free surface area, the chine and the transom. We can observe from Figure 4.5 that fine mesh and finest mesh have much more grids generated around the transom area than coarse and medium mesh. The chine refinement is shown in Figure 4.6, the plots of the grids is taken from a clip 5 cm ahead of the transom where the chine is flat. The prism layers and near wall area are well refined in the fine and finest mesh.

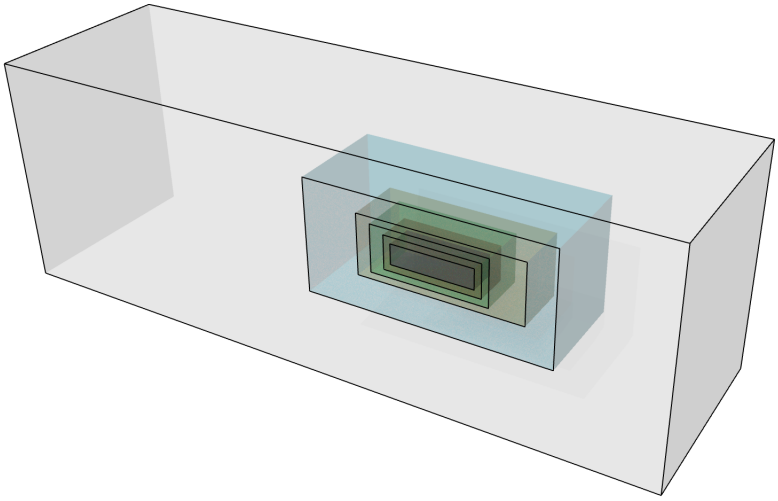


Figure 4.3: Background Grid Refinement with *topoSet-refineMesh*

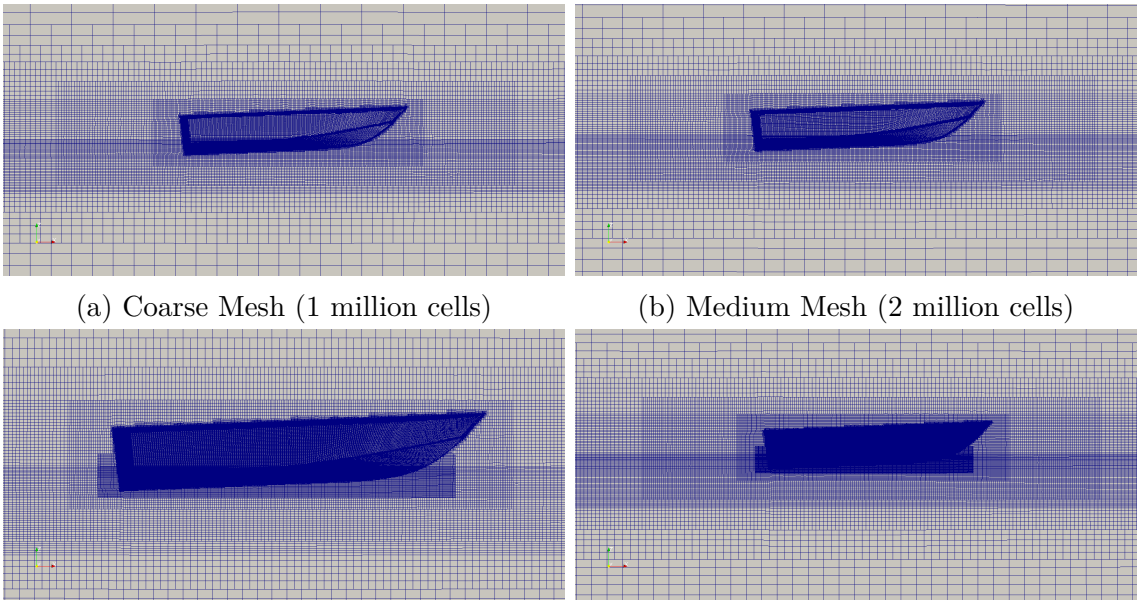
4.2 Simulation Setup and Boundary Conditions

In order to test the numerical model and four generated grids of different refinement level, fixed attitude simulations are performed. The basic simulation setup and explanations are described in Table 4.2. The total simulation time for static mesh cases is 1.5 seconds which is sufficient for convergence. However, the free attitude cases take longer time to converge and the total time is set to 3 seconds.

Table 4.2: Basic Simulation Setup

Settings	Value	Explanation
endTime	1.5	Total simulation time in seconds.
deltaT	0.00005	The first time step calculated by equation (3.10).
writeControl	adjustableRunTime	Time step is adjustable for the purpose of fast running.
maxCo	5	The maximum Courant number is 5 for fast computing.
numberOfSubdomains	192	Decompose the simulation to 192 sub-domains for parallel computation.
mass	50.757	Half hull mass.
inertia	(1.7979 0 0 10.46995 0 10.46995) (Ixx Ixy Ixz Iyy Iyz Izz)	Half hull inertia (from experiment).

The boundaries and the hull are plotted in Figure 4.2. Except for the hull, the boundaries are



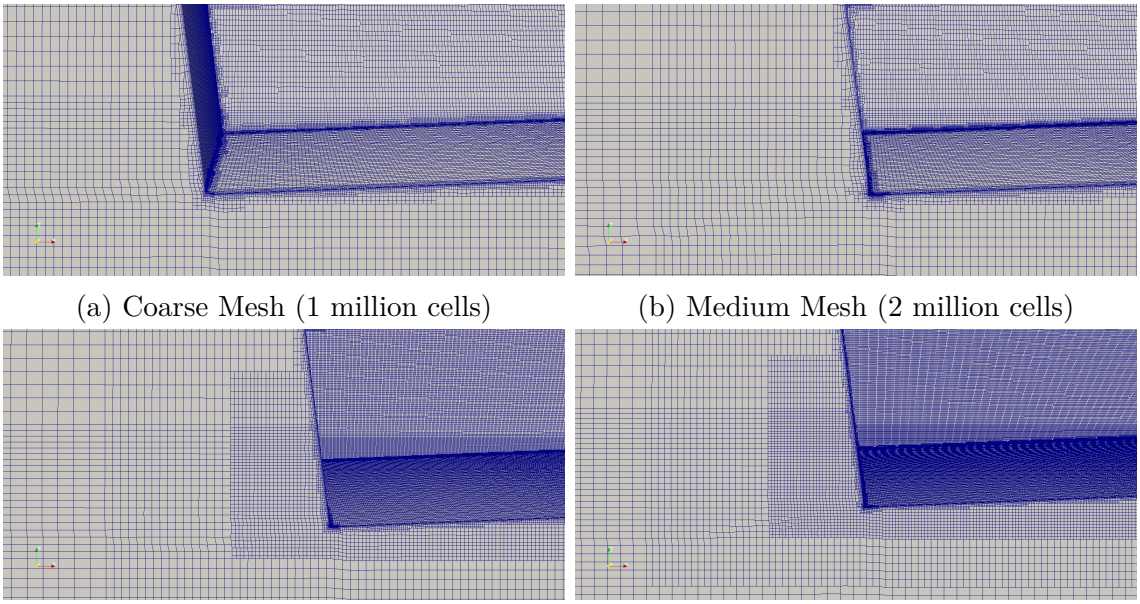
(a) Coarse Mesh (1 million cells) (b) Medium Mesh (2 million cells)

(c) Fine Mesh (4 million cells) (d) Finest Mesh (8 million cells)

Figure 4.4: Mesh Refinement Using Refinement Blocks

the inlet, outlet, atmosphere, side and midplane which are assigned as *patch*. The settings for the boundary conditions in OpenFOAM 0 directory are summarized in Table 4.3, 4.4 and 4.5 . Some explanations of the boundary conditions are provided following the corresponding table. These settings are applied to all the simulations for both static mesh and the dynamic mesh simulations unless specified or changed to improve the simulation results discussed in 7.

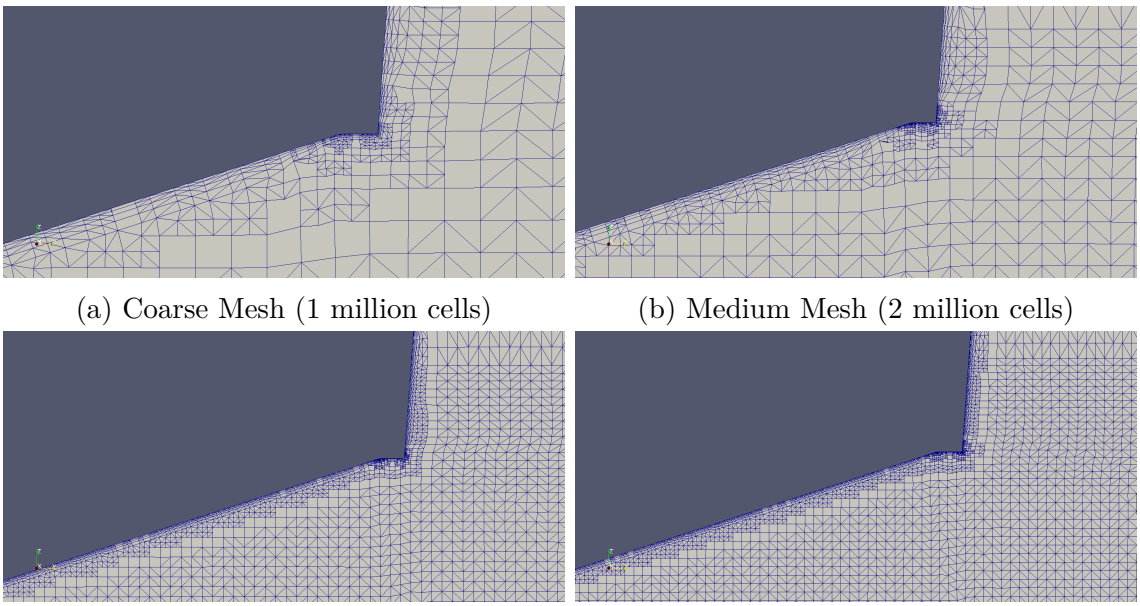
The basic constraint is called *fixedValue*, which appears several times in the settings and it provides a fixed value for the constraint, this value needs to be set by *value* keyword. For example, the *fixedValue* for *k*, *omega* and *nut* is calculated by equation (3.2) and the values are defined in the beginning of each file as *internalField*, respectively. Then, following the *fixedValue* constraint, *internalField* is specified after *value* keyword. Another commonly used constraint is *zeroGradient*. It applies a 0 gradient from the internal field to the boundary surface. For side, midplane and bottom, all the boundary constraints are set



(a) Coarse Mesh (1 million cells) (b) Medium Mesh (2 million cells)

(c) Fine Mesh (4 million cells) (d) Finest Mesh (8 million cells)

Figure 4.5: Transom Refinement



(a) Coarse Mesh (1 million cells) (b) Medium Mesh (2 million cells)

(c) Fine Mesh (4 million cells) (d) Finest Mesh (8 million cells)

Figure 4.6: Chine Refinement

to *symmetryPlane*.

kqRWallFunction is used in the k constraint for hull, this function provides a pure 0 gradient

Table 4.3: Boundary Conditions of *alpha.water* and k Settings in OpenFOAM

	alpha.water	k
internalField	uniform 0	uniform 0.0081
hull	type zeroGradient	type kqRWallFunction value internalField
inlet	type fixedValue value internalField	type fixedValue value internalField
outlet	type variableHeightFlowRate lowerBound 0 upperBound 1 value internalField	type zeroGradient
atmosphere	type inletOutlet inletValue internalField value internalField	type inletOutlet inletValue internalField value internalField
side	type symmetryPlane	type symmetryPlane
midPlane	type symmetryPlane	type symmetryPlane
bottom	type symmetryPlane	type symmetryPlane

boundary condition[13]. The *variableHeightFlowRate* is followed by the upper bound 1 and lower bound 0 are defined for α_{water}.

The *nutkWallFunction* offers a turbulence viscosity condition based on the turbulence kinetic energy. The modification of this boundary condition is discussed in 7.3. The *omegaWallFunction* is applied for hull since the turbulence model $k-\omega SST$ is based on ω. This wall function provides the constraint on turbulence specific dissipation and it's a combination of viscous and log equation. The viscous layer is given by (4.1) where $\beta_1 = 0.075$. The log-law layer equation is given by (4.2) where k indicates the value of the cell. The ω wall function then combines the viscous layer and the log-law layer as (4.3)[13].

$$\omega_{viscous} = \frac{6\nu}{\beta_1 y^2} \tag{4.1}$$

Table 4.4: Boundary Condition of *nut* and *omega* Settings in OpenFOAM

	nut		*omega*	
internalField	uniform	5e-07	uniform	1.8672
hull	type	nutkWallFunction	type	omegaWallFunction
	value	internalField	value	internalField
inlet	type	fixedValue	type	fixedValue
	value	internalField	value	internalField
outlet	type	zeroGradient	type	zeroGradient
atmosphere	type	zeroGradient	type	inletOutlet
			inletValue	internalField
			value	internalField
side	type	symmetryPlane	type	symmetryPlane
midPlane	type	symmetryPlane	type	symmetryPlane
bottom	type	symmetryPlane	type	symmetryPlane

$$\omega_{log} = \frac{k^{\frac{1}{2}}}{C_{\mu}^{\frac{1}{4}} \kappa y} \tag{4.2}$$

$$\omega = \sqrt{\omega_{viscous}^2 + \omega_{log}^2} \tag{4.3}$$

In *p_rgh*, *fixedFluxPressure* is used to set the pressure gradient to the defined *internalField* value. The velocity for the hull is always set to 0, the free body simulations use *movingWallVelocity* to provide a velocity condition for cases with moving hull.

Table 4.5: Boundary Condition of *p_rgh* and *U* Settings in OpenFOAM

	p_rgh	*U*
internalField	uniform 0	uniform (-12.24 0 0)
hull	type fixedFluxPressure value internalField	type fixedValue (static mesh) type movingWallVelocity (dynamic mesh) value uniform (0 0 0)
inlet	type fixedFluxPressure value internalField	type fixedValue value internalField
outlet	type zeroGradient	type zeroGradient
atmosphere	type totalPressure p0 uniform 0 U U phi phi rho rho psi none gamma 1 value uniform 0	type pressureInletOutletVelocity value uniform (0 0 0)
side	type symmetryPlane	type symmetryPlane
midPlane	type symmetryPlane	type symmetryPlane
bottom	type symmetryPlane	type symmetryPlane

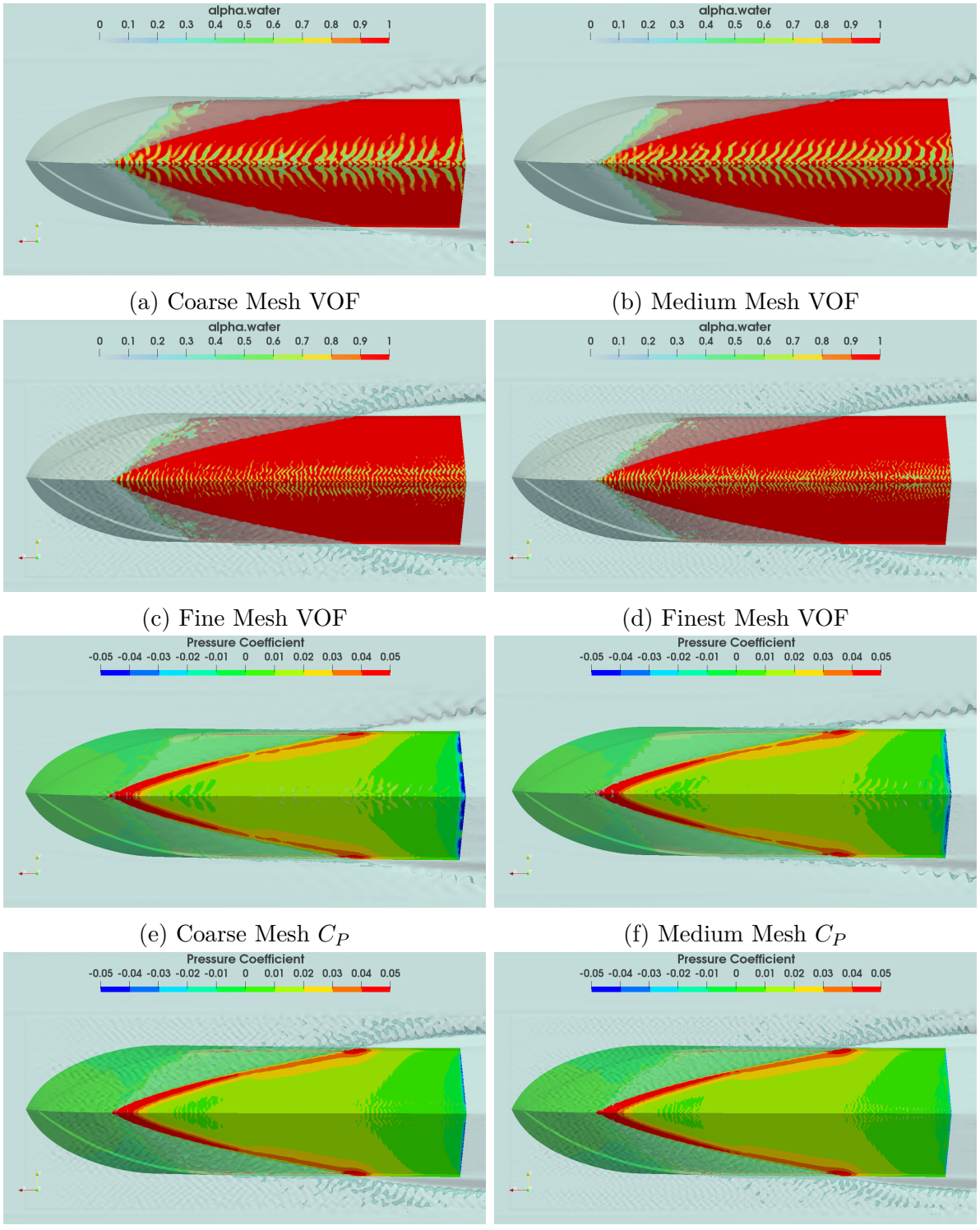
Chapter 5

Fixed Attitude Simulations: Verification & Validation

5.1 Fixed Attitude Simulation Results

By using the four generated grids described previously, the fixed attitude simulations are performed at the highest speed with *interFoam* solver. The hull is fixed in the experimental heave and trim attitude and the mesh won't change with time. Initial speed is given to air and water and the same speed is provided at the inlet for both fluids. The initial impulsive acceleration of the flow in air and water determines a sharp peak of pressure and forces during an initial transient that lasts about 0.1 seconds at this speed (see Figure 5.4). Then shear, pressure forces converge on their steady-state values that are reached after about 0.5 seconds, at this speed. The average steady-state values are taken to verify the convergence properties of the numerical model with respect to grid size. The results for resistance components (total resistance, friction drag, and pressure), wetted chine length L_C and wetted keel length L_K are shown in Table 5.1. For the cases that gain oscillatory convergence, the results summarized are the averaged results from the last 5 oscillations. The plot for VOF, pressure coefficient and $y+$ are taken from the steady state, i.e., approximately the time when the averaged results are taken.

From Table 5.1, the solutions for resistance are improved by increasing the resolution of the



(a) Coarse Mesh VOF

(b) Medium Mesh VOF

(c) Fine Mesh VOF

(d) Finest Mesh VOF

(e) Coarse Mesh C_P

(f) Medium Mesh C_P

(g) Fine Mesh C_P

(h) Finest Mesh C_P

Figure 5.1: Static Mesh VOF, Pressure Coefficient C_P and Non-dimensional Wall Distance $y+$

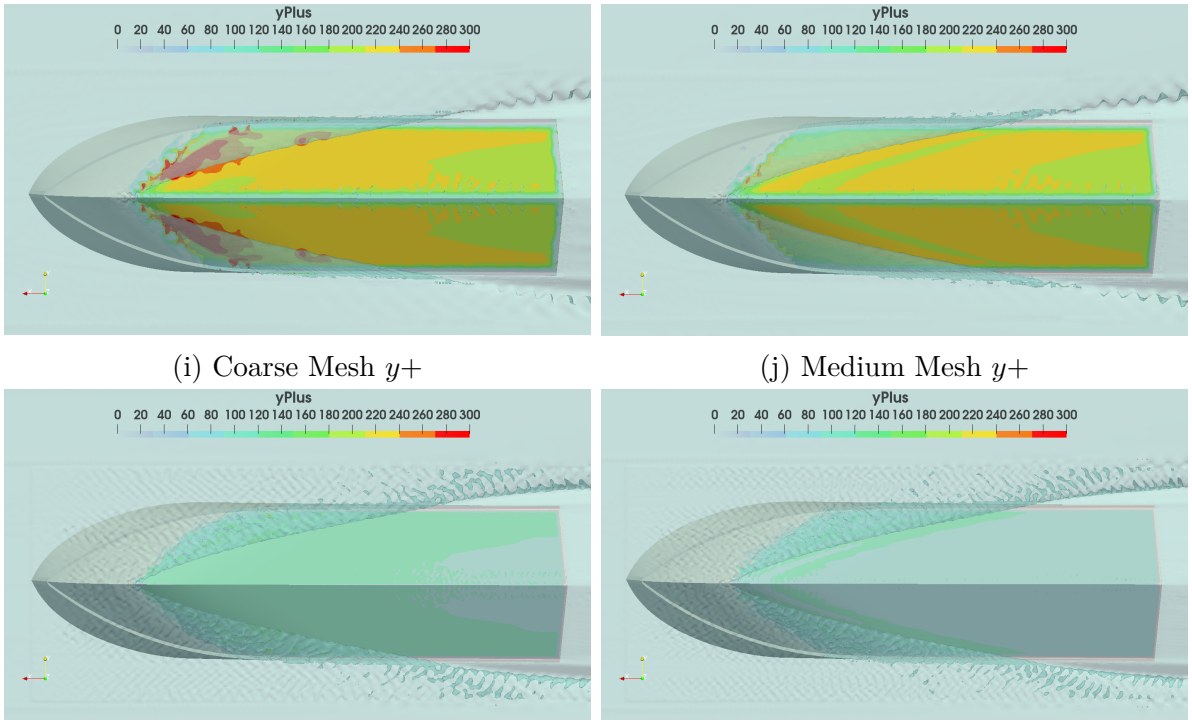
Table 5.1: Static Mesh Simulation Results at $Fn_{\nabla} = 5.72$

	Coarse Mesh	Relative Error (%D)	Medium Mesh	Relative Error (%D)	Fine Mesh	Relative Error (%D)	Finest Mesh	Relative Error (%D)	Experimental Result (D)	Savitsky Short Method
Mesh (Number of Cells)	1141900	—	2236192	—	4560229	—	8509768	—	—	—
Resistance $[N]$	243	-9.94	254	-5.87	262	-2.90	265	-1.79	270	231
Friction Drag $[N]$	178	—	184	—	193	—	194	—	—	182
Pressure $[N]$	65.0	—	70.0	—	69	—	71.0	—	—	49.3
L_C $[cm]$	52.5	18.1	52.2	17.4	52.1	17.3	50.8	14.3	44.5	50.5
L_K $[cm]$	175	1.56	175	1.45	174	0.930	175	1.54	173	175
Wetted Area $[10^3 cm^2]$	7.48	4.94	7.47	4.72	7.44	4.28	7.43	4.15	7.13	7.42

mesh. Savitsky Short Method is also used as a term of reference to assess the level of error that affects the CFD model with respect to the widely diffused semi-empirical prediction method, which still represents a kind of state of the art of planing hull drag prediction. As shown in Table 5.1, all the fixed-hull simulations give better results than Savitsky Short Method.

As shown in all VOF plots, the hull bottom has many wavy patterns along with the keel showing VOF varies from 0.5 to 0.8. This phenomenon does not correspond to any real physics observed during the experiments and it is attributed to artificial diffusion of the VOF scalar when its PDE is numerically solved with the technique described in chapter 3. This phenomenon is known to affect different CFD solvers using the volume of fluid method and it is called "numerical ventilation". The first reason is that VOF approach cannot take into account properly the wetting properties of the water. Another reason is due to the different size and orientation of prism layer cells along the hull surface. VOF is subjected to smearing between differently oriented cells, which enhance the advection of air on the bottom of the hull [5].

The plots in Figure 5.1 show corresponding solution to the results. As the mesh is more refined, the numerical ventilation is reduced, low-pressure area at the transom decreases and the $y+$ values also decrease a lot. Numerical ventilation is very similar in the coarse mesh and medium mesh shown as 5.1a and 5.1b, both meshes don't have the extra refinement block at the free surface area. Numerical ventilation also shows similarity in 5.1c and 5.1d, because



(i) Coarse Mesh $y+$ (j) Medium Mesh $y+$

(k) Fine Mesh $y+$ (l) Finest Mesh $y+$

Figure 5.1: Static Mesh VOF, Pressure Coefficient C_P and Non-dimensional Wall Distance $y+$ (cont.)

that both four-million-cell mesh and eight-million-cell mesh have the extra refinement block resulting in more refined hull bottom. In addition to that, it can be observed that the spray is solved better in the fine and finest mesh.

5.2 Effect of Prism Layer Cells

It is notable that GPPH hull has flat of chine, step, and very sharp edges, which makes the accuracy of solving the near wall area very important. Thus, changing the mesh to make the prism layers more refined is the first choice to improve the accuracy of numerical simulations. We can change the prism layers mesh by using different settings in *snappyHexMesh* as mentioned in 4.1.

When using OpenFoam *snappyHexMesh* utility, user can change the prism layer mesh with *addLayersControls* with 4 basic settings, i.e., defining the number of surface layers using *nSurfaceLayers*, the wanted thickness of final added cell layer with *finalLayerThickness*, the minimum thickness of cell layer by *minThickness* and the expansion factor for layer mesh using *expansionRatio*. There are also other advanced settings for extruding surface, snapping relaxation iterations, smoothing iterations, etc, which are all set to the default values in this study. Users can use absolute cell size for the settings or use the value that is relative to undistorted cell size outside layer by switching on *relativeSizes*, which is used in this study.

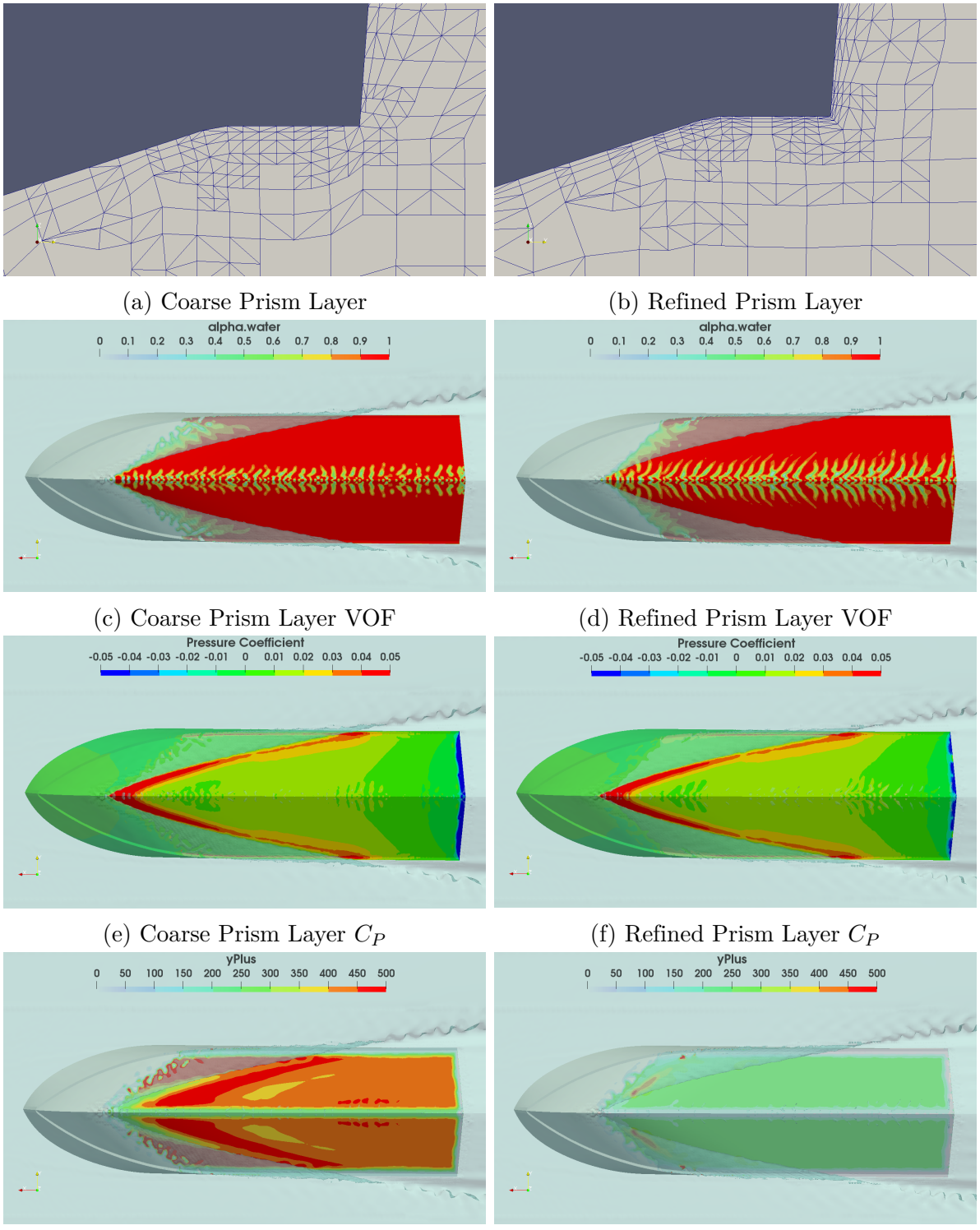
In order to compare the effect caused by different prism layers for both coarse and fine mesh, the one-million-cell mesh is chosen so as to save computational time. On the basis of the original coarse and fine mesh in the mesh convergence study, we use the same settings for background mesh and the *snappyHexMesh* except the surface layer settings. The changes are listed in Table 5.2.

Table 5.2: Surface Layer Settings

Settings	Original Mesh	Refined Prism Layers Mesh
nSurfaceLayers	3	4
expansionRatio	1.5	1.5
finalLayerThickness	1	0.7
minThickness	0.4	0.4

The modified prism layers simulation is also conducted using a fixed mesh strategy. The results are summarized in Table 5.3 and plotted in Figure 5.2 in terms of hull bottom VOF, C_P and $y+$. We can tell from Table 5.3, increasing the refinement of prim layer resolution really makes a difference to the calculation of total resistance and greatly increases the friction drag.

In Figure 5.2, the hull bottom numerical ventilation along the keel is actually increased a little by increasing the prism layer resolution. However, increasing the prism layer makes a



(a) Coarse Prism Layer

(b) Refined Prism Layer

(c) Coarse Prism Layer VOF

(d) Refined Prism Layer VOF

(e) Coarse Prism Layer C_P

(f) Refined Prism Layer C_P

(g) Coarse Prism Layer $y+$

(h) Refined Prism Layer $y+$

Figure 5.2: Results Comparison Between Coarse Prism Layer and Refined Prism Layer Simulations

Table 5.3: Static Mesh Simulation Results for Original Mesh and Refined Prism Layers Mesh

Results	Original Mesh	Relative Error (%D)	Refined Prism Layers Mesh	Relative Error Error (%D)	Experimental Results (D)
Resistance $[N]$	243	-9.94	260	-3.64	270
Friction Drag $[N]$	178	—	192	—	—
Pressure $[N]$	64.7	—	68.0	—	—
L_C $[cm]$	52.5	18.1	49.9	12.2	44.5
L_K $[cm]$	175	1.56	175	1.09	173
Wetted Area $[10^3 cm^2]$	7.48	4.94	7.37	3.36	7.13

big difference to $y+$ value, which reduces about 200 from the coarse prism layer simulation to the refined prism layer simulation. The $y+$ range in Figure 5.2 is enlarged from [0, 300] in Figure 5.1to [0,500] for the purpose of showing the value clearly.

Notice that the execution time for the coarse prism layer static mesh simulation is only 4.38 hours but the refined prism layer static mesh simulation took 10.22 hours to complete 1.5 seconds simulation. More than 2 times computation cost might be acceptable for fixed coarse mesh simulations, but for the dynamic mesh simulation with more grids, it would be a very large computational cost. In conclusion, refined prism layers can greatly improve the results but this method is not used for the further free body simulations for the large computational cost.

5.3 Mesh Convergence Study and Verification and Validation (V&V) for Fixed Attitude Simulations

Over the last 50 years, scientific computation has emerged as the third pillar of science to complement experiment and theory. To understand the strengths and limitations of scientific computing, we use Verification and Validation (V&V) which provide a means for quantitatively assessing credibility in scientific computing. Verification considers the math

problems, whether we are solving the equations correctly. While validation is checking the physics problem, whether we are solving the right equations.

In this study, the verification and validation process follows the work of Stern et al.[23] of verification and validation of CFD simulation results. Verification is used for assessing the numerical uncertainty and estimating the numerical error. The mesh convergence study is conducted using static mesh with four different resolutions, and the errors and uncertainties are estimated. Three different convergence conditions can be obtained: the first one is monotonic convergence, which will use Generalized Richardson Extrapolation in the estimation of errors and uncertainties. Detailed information of Generalized Richardson Extrapolation is described in paper [23]. The second condition is oscillatory convergence, for which the highest and the lowest solutions are used for the uncertainties. Finally, the study may also be diverging and errors and uncertainties can't be estimated [23]. In our study, the defined equations are used for error and uncertainty calculations.

For the validation study, the simulation modeling uncertainty is computed using the experimental results. The comparison error E is defined as the difference between experimental results D and simulation results S. Figure 5.3 clearly indicates the definition of E. U_D is the uncertainty of experimental results already calculated and opportunely specified in the NSWCCD test report.

The verification and validation process is performed for the mesh convergence study with 3 grids. Simulation results presented in chapter 4 obtained using grid 1 (finest mesh with about 8 million cells) are taken as the highest resolution data. Grid 2 is set to be the four-million-cell fine mesh, grid 3 to be two-million-cell medium mesh and grid 4 is the coarse mesh which has 1 million cells. Since we have 4 grids, two sets of studies are performed with these 4 different grids. Study 1 uses grid 1, grid 2 and grid 3, while Study 2 uses grid 2, grid 3 and grid 4. Since it's a 3D refinement, the uniform refinement ratio is calculated in (5.1)

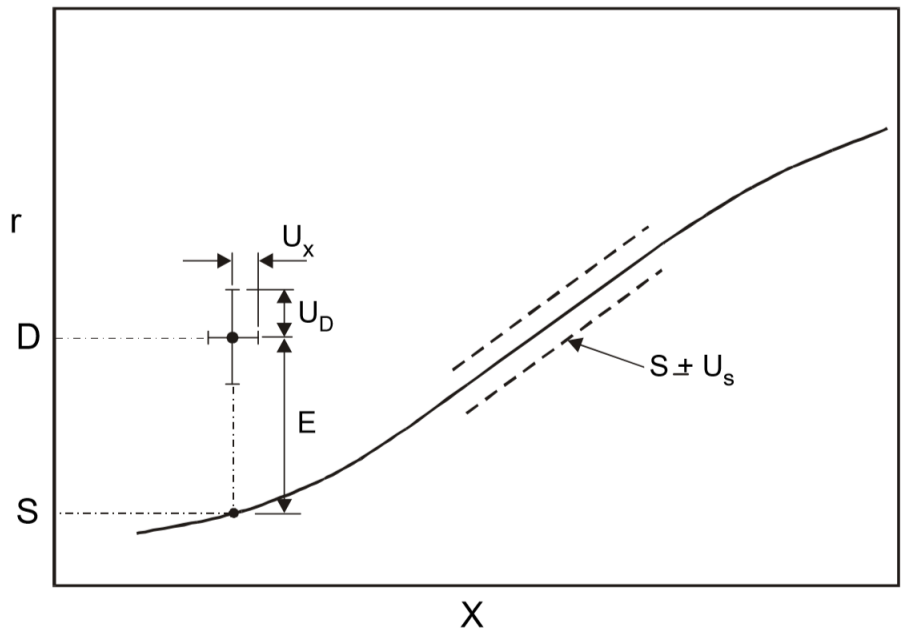


Figure 5.3: Comparison Error for Validation (adapted from [23])

where N_{G_2} indicates the number of cells of a fine mesh while N_{G_1} indicates the number of cells of a coarse mesh. Our mesh is 3 dimensional unstructured mesh and $\frac{N_{G_2}}{N_{G_1}}$ is set to 2, r_G = 1.26 is used in the entire verification and validation process.

$$r_G = \left(\frac{N_{G_2}}{N_{G_1}}\right)^{\frac{1}{3}} \tag{5.1}$$

5.3.1 Mesh Convergence Study

In the mesh convergence study, firstly the resistance coefficient is needed, which requires the calculation of wetted surface area. The wetted surface area is calculated using the wetted chine length and wetted keel length read from ParaView during the post-process of the simulation results. The value used for beam B is the maximum projected chine beam and all the hull characteristic used are taken from the experiment benchmark data. The wetted surface area S is calculated according to equation (5.2). Then, the resistance coefficients are

calculated, i.e. the total resistance coefficient C_T, pressure coefficient C_P and the friction drag coefficient C_F. The method is using the forces from simulation results, the averaged total resistance R_T, the averaged pressure R_P and the averaged friction drag R_F, and divide the forces by $\frac{1}{2}\rho V^2 S$. The force coefficients are calculated according to equation (5.3), (5.4) and (5.5).

$$S = Wetted\ \ Surface\ \ Area = \frac{(L_C + L_K) \times B}{2\cos(\beta)} \tag{5.2}$$

$$C_T = Total\ \ Resistance\ \ Coefficient = \frac{R_T}{\frac{1}{2}\rho V^2 S} \tag{5.3}$$

$$C_P = Pressure\ \ Coefficient = \frac{R_P}{\frac{1}{2}\rho V^2 S} \tag{5.4}$$

$$C_F = Friction\ \ Drag\ \ Coefficient = \frac{R_F}{\frac{1}{2}\rho V^2 S} \tag{5.5}$$

The solution change ε due to a grid refinement is calculated as follows: set the solutions for grid 1 as S_{k_1}, similarly, solutions for grid 2, 3 and 4 can be represented by S_{k_2}, S_{k_3} and S_{k_4}, respectively. The solution change is calculated by equation (5.6). The results for force coefficients and solution change are shown in Table 5.4.

$$\begin{aligned} \varepsilon_{21_k} &= \frac{S_{k_1} - S_{k_2}}{S_{k_2}} \times 100\% \\ \varepsilon_{32_k} &= \frac{S_{k_2} - S_{k_3}}{S_{k_3}} \times 100\% \\ \varepsilon_{43_k} &= \frac{S_{k_3} - S_{k_4}}{S_{k_4}} \times 100\% \end{aligned} \tag{5.6}$$

A global convergence ratio R_k is used to decided the convergence conditions. For study 1, R_k is defined as $R_k = \varepsilon_{21_k}/\varepsilon_{32_k}$, similarly, for study 2, R_k is defined as $R_k = \varepsilon_{32_k}/\varepsilon_{43_k}$. In our cases, the subscript k in R_k indicates different components of resistance, i.e. total resistance, pressure and the friction drag. According to the results of R_k, there are three convergence

Table 5.4: Mesh Convergence Study for Total Resistance C_T, Pressure C_P, Friction Drag C_F ($\times 10^{-3}$) at Fn_∇=5.72

Grid	Grid 4 (1 million cells)	Grid 3 (2 million cells)	Grid 2 (4 million cells)	Grid 1 (8 million cells)	Experimental Results
C_T	4.37	4.42	4.70	4.82	5.05
ε	—	1.14%	6.33%	2.55%	—
C_P	1.16	1.17	1.25	1.27	—
ε	—	0.862%	6.84%	1.60%	—
C_F	3.20	3.25	3.46	3.55	—
ε	—	1.56%	6.46%	2.60%	—

conditions that are possible [23]:

$$\begin{aligned} Monotonic \quad Convergence &: 0 < R_k < 1 \\ Oscillatory \quad Convergence &: R_k < 0 \\ Divergence &: R_k > 1 \end{aligned} \tag{5.7}$$

The values for R_k and convergence results are summarized in Table 5.5. From the table, all the R_k values for study 1 satisfy the convergence condition (i) described in (5.7), which indicates that study 1 attains monotonic convergence. However, the R_k values for study 2 are all lager than 1, which matches the condition (iii) in (5.7). Thus, mesh convergence study 2 is divergent.

Table 5.5: Global Convergence Ratio R_k of Static Mesh Resistance

Resistance	Study 1 (Grid 1, 2, and 3)	Study 2 (Grid 2, 3, and 4)
C_T	0.403	5.55
Convergence Result	Monotonic Convergence	Divergence
C_P	0.234	7.94
Convergence Result	Monotonic Convergence	Divergence
C_F	0.403	4.14
Convergence Result	Monotonic Convergence	Divergence

5.3.2 Verification and Validation for Fixed Attitude Simulations

Verification study for total resistance is performed by taking both iterative and grid convergence study into account. The absolute value of simulation numerical error is expressed as δ_{SN}, and it is decomposed into iterative error contribution δ_I and grid size error contribution δ_G. The relationship is described as equation (5.8). The uncertainty of numerical simulation is U_{SN}, and it is also decomposed into the uncertainty from iteration U_I and uncertainty from grid size U_G. The relationship is described as equation (5.9).

Since we already know that study 2 which uses grid 2, grid 3 and grid 4 is diverging, the errors and uncertainties can't be estimated and it makes no sense to calculate the errors and uncertainties. It is also questionable to calculate these values according to equations for the monotonic convergence condition. But the errors and uncertainties are still computed for study 2 using the actual simulation results (as opposed to the converged values) just for reference and comparison to help us to choose the right mesh used in further simulations.

$$\delta_{SN} = \delta_I + \delta_G \tag{5.8}$$

$$U_{SN} = U_I + U_G \tag{5.9}$$

The plot 5.4 for the convergence of total resistance and resistance components for the 4 different grids are used for analyzing iterative components of both simulation numerical error and the uncertainty. The variation in C_T is calculated according to the maximum and minimum values over the last five oscillation periods as $max[S_{max} - S_G, S_{min} - S_G]$, the averaged simulation results are used as S_G for different resistance. The variations are summarized in Table 5.6. Since the variations for all components of all four grids are very small, the iterative components of errors and uncertainties are assumed to be negligible

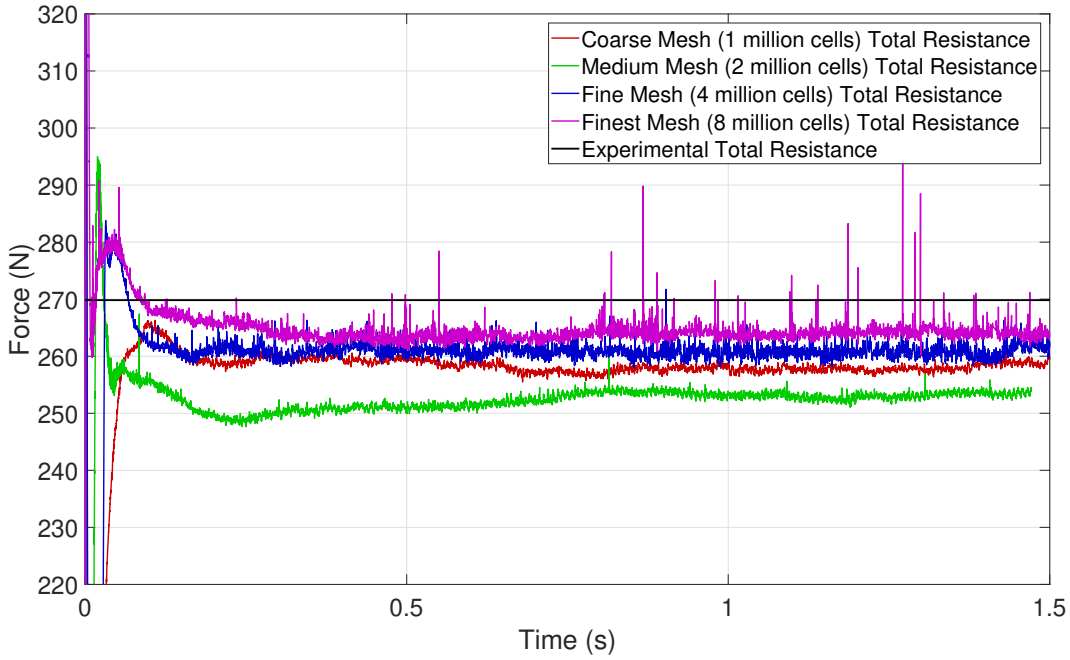


Figure 5.4: Total Resistance of Four Grids Versus Time

comparing to the grid size components of errors and uncertainties.

Table 5.6: Variation for Total Resistance R_T, Pressure R_P, Friction Drag R_F ($\times 10^{-3}$) at Fn_∇=5.72

Grid	Grid 4 (1 million cells)	Grid 3 (2 million cells)	Grid 2 (4 million cells)	Grid 1 (8 million cells)
S_G of R_T [N]	243	254	262	265
R_T Variation [N]	1.03	2.12	2.76	2.96
R_T Variation ($\%S_G$)	0.420	0.840	1.06	1.12

Next, we need to calculate the simulation numerical error and the uncertainty due to grid size. In this part, R_G is defined as the grid convergence ratio, the value of R_G is the same as the value of R_K summarized in Table 5.5. In order to calculate the observed order of accuracy p_G, a uniform parameter refinement ratio $r_G = 1.26$ is used according to the total number of cells. Also, the absolute value of the error of C_T is calculated by equation (5.10). Use the refinement ratio r_G and the errors, the observed order of accuracy is calculated by

equation (5.11).

$$\begin{aligned} \varepsilon_{43_G} &= |C_{T_{G4}} - C_{T_{G3}}| \\ \varepsilon_{32_G} &= |C_{T_{G3}} - C_{T_{G2}}| \\ \varepsilon_{21_G} &= |C_{T_{G2}} - C_{T_{G1}}| \end{aligned} \tag{5.10}$$

$$\begin{aligned} Study1 : p_{G1} &= \frac{\ln\left(\varepsilon_{32_G}/\varepsilon_{21_G}\right)}{\ln\left(r_G\right)} \\ Study2 : p_{G2} &= \frac{\ln\left(\varepsilon_{43_G}/\varepsilon_{32_G}\right)}{\ln\left(r_G\right)} \end{aligned} \tag{5.11}$$

A correction factor is defined as C_G by equation (5.12), where $p_{G_{est}} = 2$. Since we use Generalized Richardson Extrapolation method, the first-order Richardson Extrapolation estimate $\delta_{RE_{G1}}$ and $\delta_{RE_{G2}}$ are calculated by equation (5.13) for study 1 and study 2, respectively. The numerical error due to grid size δ_G is estimated using C_G and δ_{RE_G} as described in equation (5.14). The uncertainty due to grid size U_G is estimated using δ_{RE_G} instead of δ_G as described in equation (5.15), also the corrected U_G defined as U_{G_C} is calculated by equation (5.16).

$$C_G = \frac{r_G^{p_G} - 1}{r_G^{p_{G_{est}}} - 1} \tag{5.12}$$

$$\begin{aligned} Study1 : \delta_{RE_{G1}} &= \frac{\varepsilon_{21_G}}{r_G^{p_{G1}} - 1} \\ Study2 : \delta_{RE_{G2}} &= \frac{\varepsilon_{32_G}}{r_G^{p_{G2}} - 1} \end{aligned} \tag{5.13}$$

$$\begin{aligned} Study1 : \delta_{G_1} &= C_G \times \delta_{RE_{G1}} \\ Study2 : \delta_{G_2} &= C_G \times \delta_{RE_{G2}} \end{aligned} \tag{5.14}$$

$$U_G = |C_G \delta_{RE_G}| + |\left(1 - C_G\right) \delta_{RE_G}| \tag{5.15}$$

$$U_{G_C} = |\,(1 - C_G)\,\delta_{RE_G}|$$
$$\delta_G = C_G \delta_{RE_G} \tag{5.16}$$

Use the solution for grid 1 as S_{G_1}, the solution for grid 2 as S_{G_2} and the corrected solution S_C is calculated by equation (5.17). All the calculation results are summarized in Table 5.7. We can see from the table that the verification for study 1 attains 6.60% S_{G_1} uncertainty, and a smaller corrected uncertainty 2.37% S_{G_1}. The error for study 1 verification is 4.23% S_{G_1}, which is relatively small. However, the results for the verification of study 2 is very large. The uncertainty is 27.6% S_{G_2} and the corrected uncertainty is 17.4% S_{G_2}, which is smaller than the original uncertainty. It also proves that there is no point in calculating these values because of the divergence result of mesh convergence study of study 2.

$$Study1 : S_{C_1} = S_{G_1} - \delta_{G_1}$$
$$Study2 : S_{C_2} = S_{G_2} - \delta_{G_2} \tag{5.17}$$

Table 5.7: Verification of Total Resistance C_T ($\times 10^{-3}$) of Fixed Attitude Simulations, Uncertainties and Errors Expressed in % S_{G_1} for Study 1, and % S_{G_2} for Study 2

Study	R_G	p_G	C_G	U_G	δ_G	U_{G_C}	S_C
1 (Grids 1-3)	0.403	3.67	2.27	6.60%	4.23%	2.37%	4.62
2 (Grids 2-4)	5.55	-7.45	-1.40	27.6%	10.2%	17.4%	4.22

The validation is also performed for both studies. Study 1 use $S = S_{G_1}$ while study 2 use $S = S_{G_2}$, both of them use their own corrected solution described in Table 5.7, respectively. The comparison error is calculated by equation (5.18). The data uncertainty U_D is given in experimental result and $U_D = 0.32\%D$ where data D is the experimental result. Considering

$U_{SN} = U_G$, the validation uncertainty is given by equation (5.19).

$$E = D - S \tag{5.18}$$

$$U_V = \sqrt{U_{SN}^2 + U_D^2} \tag{5.19}$$

When using the corrected solution, the process is almost the same except using S_C summarized in table 5.7 instead of S. Similarly, the corrected grid uncertainty $U_{G_C} = U_{S_cN}$ is used instead of U_G, meanwhile, U_D doesn't change according to the experiment. The equation for the corrected comparison error is (5.20), the corrected validation uncertainty is computed by equation (5.21).

$$E = D - S_C \tag{5.20}$$

$$U_{V_C} = \sqrt{U_{S_CN}^2 + U_D^2} \tag{5.21}$$

The validation results for study 1 are summarized in Table 5.8, the results for study 2 are summarized in Table 5.9. The validation uncertainty and the corrected validation uncertainty are both smaller than 1% while study 2 has both uncertainties over 10%, which is similar to the results of verification.

Table 5.8: Validation of Total Resistance C_T for Study 1

	E (%D)	U_V (%D)	U_D (%D)	U_{SN} (%D)
$E = D - S$	4.55	6.31	0.32	6.30
$E_C = D - S_C$	8.51	2.28	0.32	2.26

Table 5.9: Validation of Total Resistance C_T for Study 2

	E (%D)	U_V (%D)	U_D (%D)	U_{SN} (%D)
$E = D - S$	6.93	25.7	0.32	25.7
$E_C = D - S_C$	16.4	16.2	0.32	16.2

By comparing the verification and validation results for study 1 and study 2 using 4 different resolutions, it's very obvious that grid 1, 2 and 3 from study 1 should be chosen for the free attitude simulations. The results estimations are listed in Table 5.10. Due to divergence of study 2, only grid 1 estimation is plotted in Figure 5.5 using equation (5.22), where $S_{G1_{est}}$ is the estimated solution for grid 1 using S_{G_2} and S_{G_3}. $S_{G2_{est}}$ represents the estimation for grid 2 using S_{G_3} and S_{G_4}.

$$\begin{aligned} S_{G1_{est}} &= S_{G_2} + \frac{S_{G_2} - S_{G_3}}{r_G^{P_G 1} - 1} \\ S_{G2_{est}} &= S_{G_3} + \frac{S_{G_3} - S_{G_4}}{r_G^{P_G 2} - 1} \end{aligned} \tag{5.22}$$

Table 5.10: Estimated Solution Using Richardson Extrapolation ($\times 10^{-3}$)

	Grid 4	Grid 3	Grid 2	Grid 1	Experimental Results
Simulation C_T	4.37	4.42	4.70	4.82	5.05
Estimated C_T	—	—	4.36	4.91	—

By comparing the estimations and the simulation results in Figure 5.5, $S_{G1_{est}}$ is very close to the grid 1 simulation result, which means that using Richardson Extrapolation can make a good estimation for the finest mesh. However, the estimated result for grid 2 shown in Table 5.10 has a big discrepancy with the simulation result. This also explains the divergence of study 2. Again, this result might be also due to the non-systematically refined mesh.

Study 2 suggests not using grid 4, but it is noted that the grid 4 simulation results are close to grid 3 and it's computation time is much shorter. Considering that we need to find the mesh that gives the best results as well as taking acceptable computation time, thus, the computational cost is taken into consideration to choose the right meshes for further simulations. All the simulations were run on Virginia Tech ARC Cascades cluster. The total execution time and processors used for each grid are shown in Table 5.11. Also, a bar plot

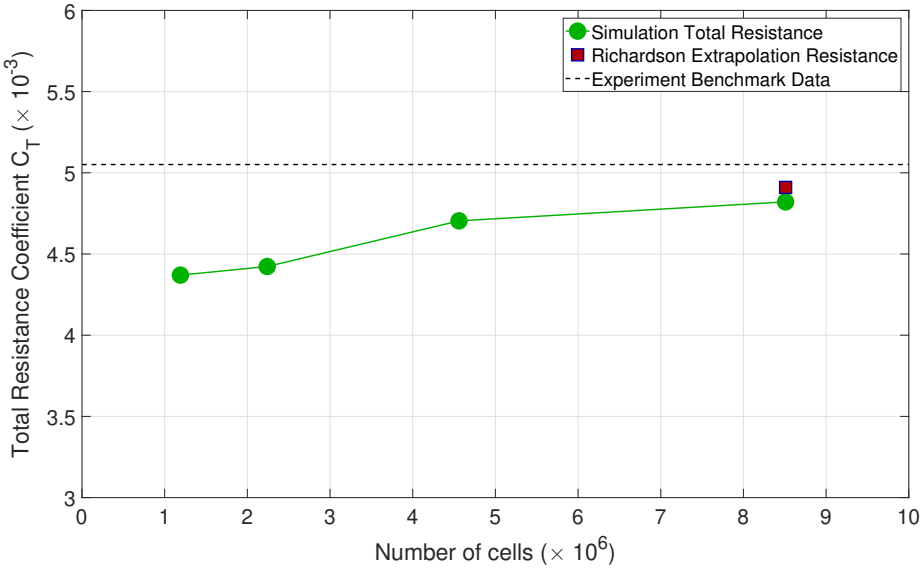


Figure 5.5: Estimated Results Using Richardson Extrapolation

of S_G/S_{G_4} is presented as Figure 5.6 for the comparison of computation cost and resistance relative error.

Table 5.11: Computational Cost for the Four Different Grids

Grid	Grid 4 (1 million cells)	Grid 3 (2 million cells)	Grid 2 (4 million cells)	Grid 1 (8 million cells)
Total Resistance Relative Error %D	-9.94	-5.87	-2.90	-1.79
S_G/S_{G_4}	1.00	0.590	0.290	0.180
Execution Time [hours]	4.38	25.8	70.8	133
S_G/S_{G_4}	1.00	5.89	16.2	30.4
Processors	192	192	192	192

In Table 5.11, grid 1 takes a long time to finish the 1.5 seconds' simulation. grid 1 only improves the results by 1.11% of experimental resistance than grid 3 while it takes almost 2 times the computational cost of grid 3. It's also very clearly shown in the bar plot that the errors decrease a little while the execution time is much more longer. In conclusion, grid 4 is not suitable to be chosen for further dynamic mesh simulations. It is noted that the verification process is performed with a uniform parameter refinement ratio $r_G = 1.26$ which requires the uniform refinement of structured mesh. For example, the coarse mesh

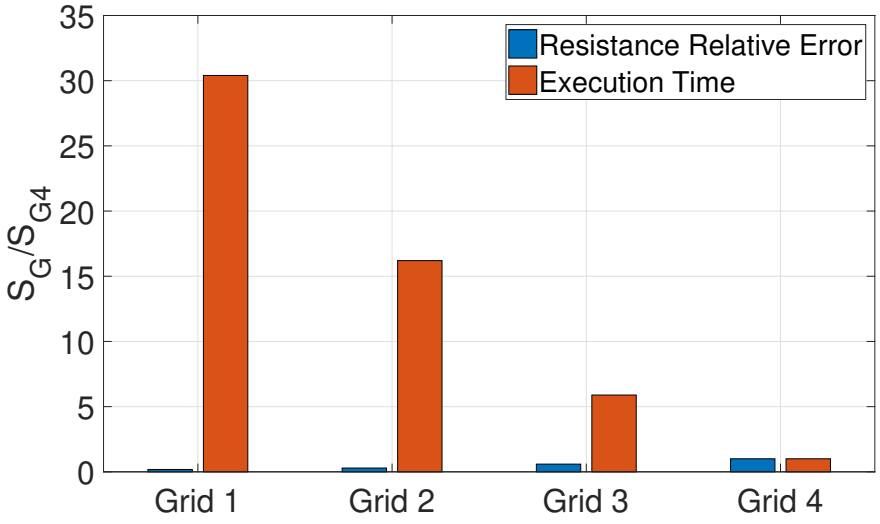


Figure 5.6: Computation Cost and Resistance Relative Error Comparison Plot for Mesh Convergence Study

has $4 \times 4 \times 4$ grids and the medium mesh has $8 \times 4 \times 4$ grids. However, our mesh strategy is using unstructured mesh for the refinement blocks and *topoSet* strategy, the refinement is not uniform in one direction. The refinement ratio is just based on the total number of cells. Thus, this questions the applicability of the verification and validation method for the simulations. This study is just used for estimation and reference to help to choose the right mesh for further dynamic simulations.

Chapter 6

Free Attitude Simulation Analysis

The dynamic mesh strategy is used in performing the free attitude simulations at all speeds from pre-planing (Fn_∇=2.6) to fully planing (Fn_∇=5.7) regimes. In previous OpenFOAM versions, for example, OpenFOAM v5.0 also used by the author previously, the dynamic mesh is applied by using the solver *interDymFoam*. However, in the new releases like OpenFOAM v1806 and v1812, the moving mesh functionality is incorporated directly into *interFoam*, which only had static mesh functionality in early releases. If running free attitude simulation, *dynamicMeshDict* and *dynamicMeshDict.sixDoF* files need to be added into *constant* directory. During the entire simulation, the mesh is deforming to respond to the prescribed or solution-dependent boundary motion by solving a motion equation formulated as a Laplacian with variable diffusivity[10]. The simulations took longer time than the static mesh due to this continue re-adaptation of the mesh nodes (also called morphing) at each time step.

Following the mesh sensitivity analysis and verification and validation study performed in Chapter 5, we choose the coarse mesh (1 million cells) and medium mesh (2 million cells) to perform the free attitude simulations for all 7 speeds. The fine mesh is firstly used at the highest speed and the results obtained at this speed are used for mesh convergence study and V&V study. However, the four-million-cell mesh at highest speed takes more than one month to complete only 3 seconds simulation with 192 cores, and the simulation has to be manually restarted every 72 hours. Therefore, the fine mesh is not used in the other 6 speeds. The simulation results for different speeds are summarized in the following sections.

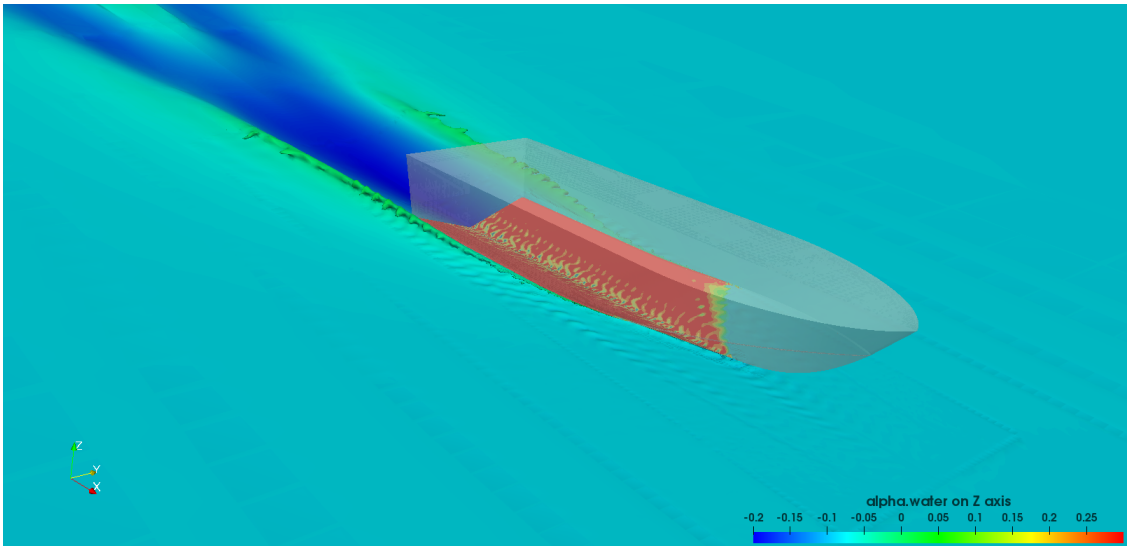


Figure 6.1: Free Body Simulation at $Fn_{\nabla} = 4.21$

The results are presented in terms of total resistance, friction drag, pressure, trim angle, heave, wetted chine length and wetted keel length. Simulation results obtained with the coarse mesh (1 million cells) and medium mesh (2 million cells) are also compared in terms of plots of VOF, pressure coefficient and non-dimensional wall distance $y+$. Figure 6.1 gives a perspective view of the hull running at its dynamic attitude with predicted free surface and VOF distribution on the bottom at 9 m/s.

6.1 Free Attitude Simulation Results at $Fn_{\nabla} = 5.72$

This section presents the results for the fully planing regime, i.e., the highest speed at 12.24 m/s, corresponding to Fn_{∇}=5.72. The fine mesh (4 million cells) is also used at this speed for dynamic mesh convergence study. Results are summarized in Table 6.1 and comparison of the solutions obtained with 3 different mesh simulations are plotted in Figure 6.2, 6.3, and 6.4.

Figure 6.2 clearly shows the numerical ventilation at the hull bottom and it is greatly reduced

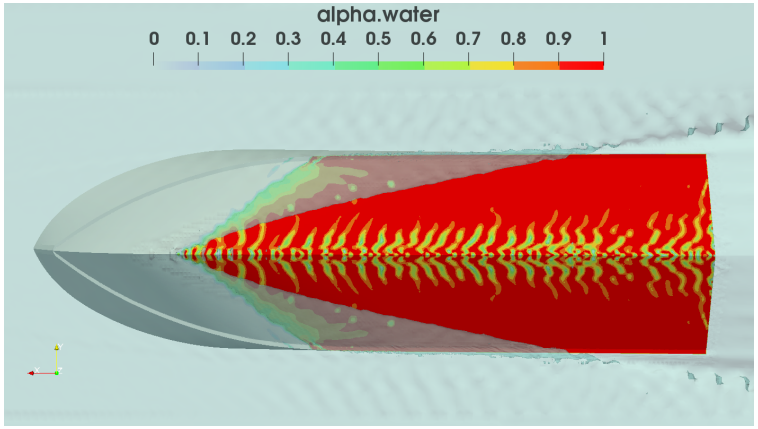
Table 6.1: Numerical Results for Free Attitude Simulations Obtained with the Three Meshes at $Fn_\nabla = 5.72$

	Coarsest Mesh	Relative Error (%)	Medium Mesh	Relative Error (%)	Fine Mesh	Relative Error (%)	Experimental Result	Savitsky Short Method
Mesh (Number of Cells)	1141900	—	2424676	—	4560229	—	—	—
Resistance [N]	220	18.3	244	9.57	252	6.61	270	231
Friction Drag [N]	158	—	182	—	190	—	—	182
Pressure [N]	62.0	—	62.0	—	62.0	—	—	49.3
Trim Angle [degree]	1.87	34.2	2.14	1.46	2.19	1.08	2.17	2.84
Heave [cm]	10.3	7.54	9.99	4.08	9.89	3.14	9.60	10.3
L_C [cm]	43.5	2.25	43.1	3.01	43.0	3.55	44.5	50.5
L_K [cm]	169	2.19	173	0.280	174	0.740	173	175

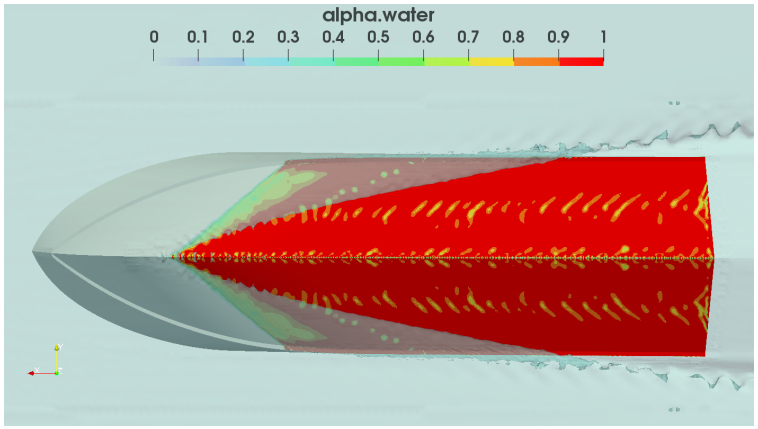
as the mesh becomes more refined. The low pressure area at the transom also reduces in the pressure coefficient plots 6.3a, 6.3b and 6.3c. The third column of plots shows non-dimensional wall distance and it decreases greatly by refinement at the bottom. Note that the range for $y+$ in 6.4a is [0,300], [0,270] in 6.4b and [0,130] in 6.4c. This is because using the same range would lead to a vague coloring in fine mesh $y+$ plot. Also, it can be observed that the spray is solved better by the fine and finest mesh. In a word, the solutions are improved by reducing the numerical ventilation and non-dimensional wall distance through increasing the mesh resolution.

6.2 Verification and Validation for Free Attitude Simulations at Highest Speed $Fn_\nabla = 5.72$

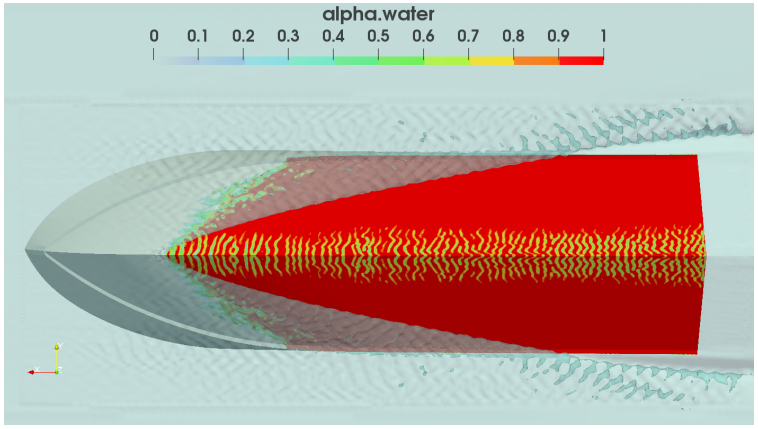
Since free attitude simulation is different from the fixed attitude simulation due to the deformation of the mesh, a mesh convergence study, as well as a V&V study, are required to analyze the errors and uncertainties. The mesh convergence study requires at least 3 simulations, therefore, we use the free body simulation results obtained at highest speed V = 12.24 m/s. The same process is performed as described in previous chapter for the fixed attitude simulations, the mesh convergence results are summarized in Table 6.2.



(a) Coarse Mesh VOF

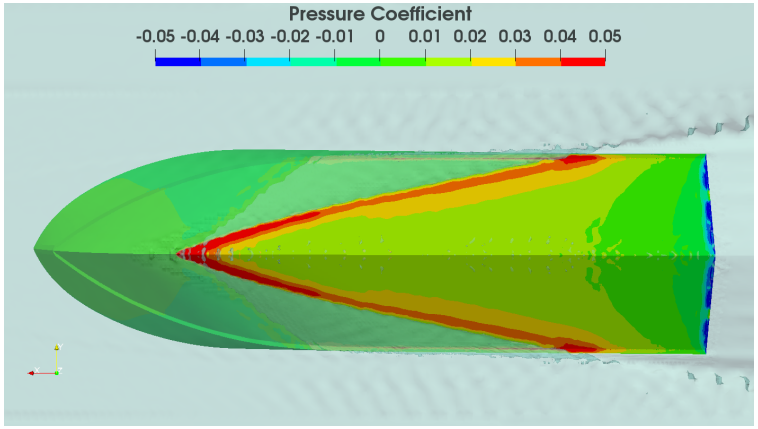


(b) Medium Mesh VOF

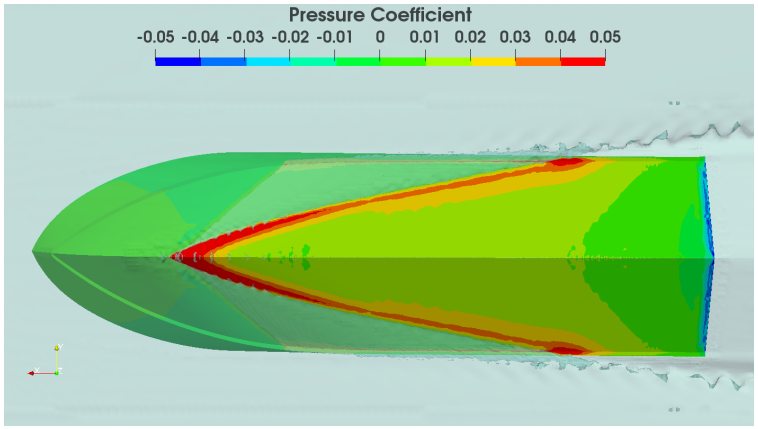


(c) Fine Mesh VOF

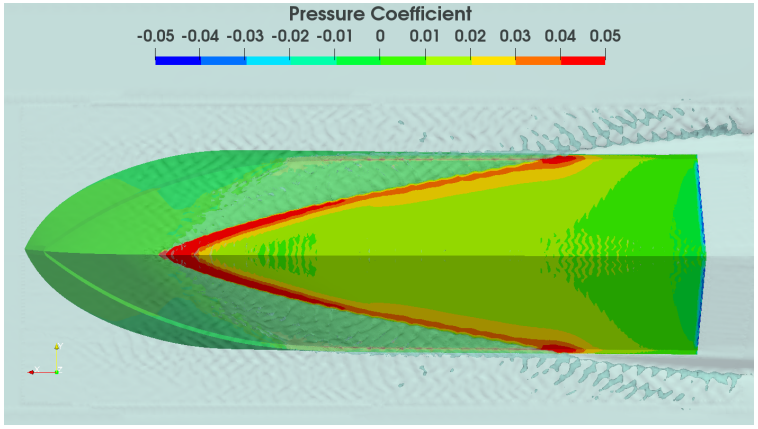
Figure 6.2: Numerical Solution Obtained Using the Three Different Grids Described in Table 6.1. Results Presented in Terms of Volume of Fluid on the Hull Bottom, Steady-state Solution at $\mathrm{Fn}_\nabla = 5.72$.



(a) Coarse Mesh C_P

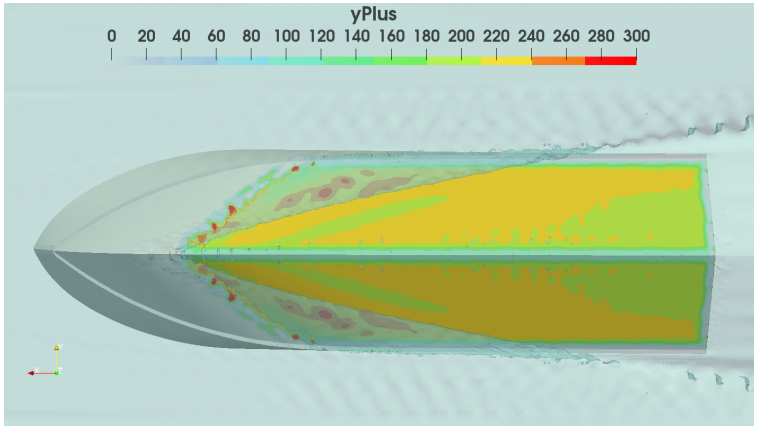


(b) Medium Mesh C_P

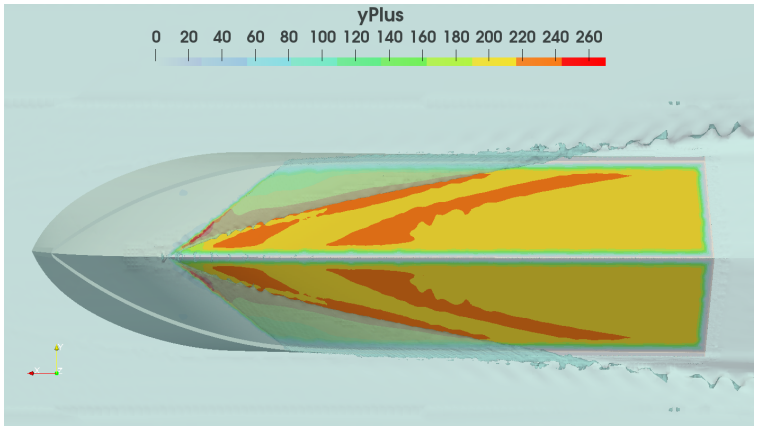


(c) Fine Mesh C_P

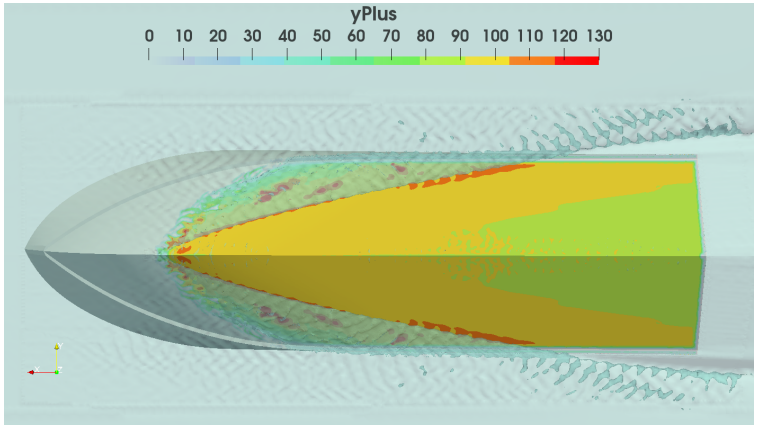
Figure 6.3: Numerical Solution Obtained Using the Three Different Grids Described in Table 6.1. Results Presented in Terms of Pressure Coefficient on the Hull Bottom, Steady-state Solution at $Fn_\nabla = 5.72$.



(a) Coarse Mesh $y+$



(b) Medium Mesh $y+$



(c) Fine Mesh $y+$

Figure 6.4: Numerical Solution Obtained Using the Three Different Grids Described in Table 6.1. Results Presented in Terms of Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at $\mathrm{Fn}_\nabla = 5.72$.

Table 6.2: Mesh Convergence Study for Total Resistance C_T, Pressure C_P, Friction Drag C_F ($\times 10^{-3}$) Obtained with Free Attitude Simulations at $Fn_\nabla = 5.72$

Grid	Grid 3 (1 million cells)	Grid 2 (2 million cells)	Grid 1 (4 million cells)	Experimental Results
C_T	4.22	4.59	4.72	5.05
ε	—	8.77%	2.83%	—
C_P	1.19	1.16	1.16	—
ε	—	-2.75%	0%	—
C_F	3.03	3.42	3.56	—
ε	—	11.4%	3.93%	—

The value of global convergence ratio and the convergence results according to the convergence criteria are shown in Table 6.3. We can observe from the table that mesh convergence study only gained oscillatory convergence of C_P while the total resistance and the frictional drag solutions attain monotonic convergence. Although we get divergence result for fixed attitude study 2 using the same coarse, medium and fine mesh, monotonic convergence is attained for the free attitude simulations due to dynamic mesh strategy.

Table 6.3: Global Convergence Ratio R_k And Convergence Results Obtained with Free Attitude Simulations at $Fn_\nabla = 5.72$

	C_T	C_P	C_F
R_k	0.323	0.00	0.345
Result	Monotonic Convergence	Oscillatory Convergence	Monotonic Convergence

For verification, firstly, the variation of the results are computed from the last five oscillations and shown in Table 6.4. Figure 6.5 shows the resistance variation versus time. Due to the variations for all components of the four grids are very small, the iterative components of error and uncertainty are assumed to be negligible comparing to the grid size components of error and uncertainty.

From Table 6.5 and 6.6, the verification study yields simulation numerical uncertainty $7.9\%S_{G_1}$ and a corrected uncertainty $4.5\%S_{G_1}$. After performing the validation study, a

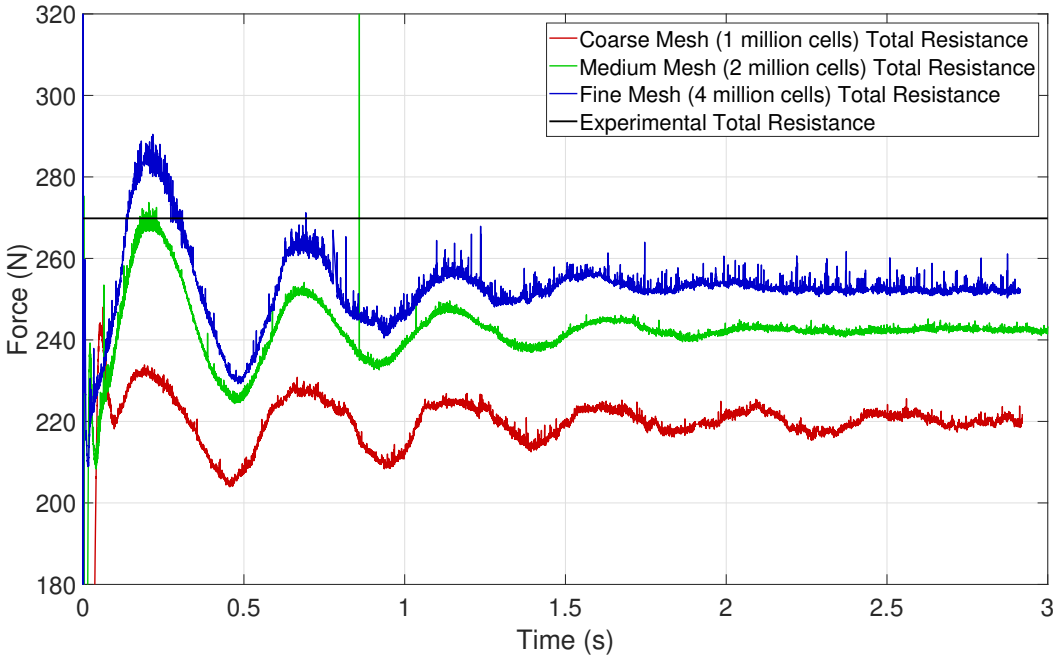


Figure 6.5: Total Resistance of Four Grids Versus Time

Table 6.4: Variation for Total Resistance R_T, Pressure R_P, Friction Drag R_F ($\times 10^{-3}$) Obtained with Free Attitude Simulations at $\mathrm{Fn}_\nabla = 5.72$

Grid	Grid 3 (1 million cells)	Grid 2 (2 million cells)	Grid 1 (4 million cells)
S_G of R_T	220	244	252
R_T Variation	1.40	1.12	1.61
R_T Variation ($\%S_G$)	0.636	0.459	0.639

validation uncertainty of 7.39%D is obtained. The Richardson Extrapolation estimated solution for grid 1 is shown in Table 6.7 and plotted in Figure 6.6. Richardson Extrapolation provides very close estimation to grid 1 simulation solution.

Finally, in order to select the meshes for free attitude simulations, the computational cost needs to be considered. The computational cost is compared in Table 6.8 and plotted in 6.7.

As shown in Table 6.8 and Figure 6.7, the four-million-cell mesh is not an ideal choice for

Table 6.5: Verification of Total Resistance C_T ($\times 10^{-3}$) Obtained with Free Attitude Simulations at $Fn_\nabla = 5.72$, Uncertainties and Errors Expressed in % S_{G_1}

Study	R_G	p_G	C_G	U_G	δ_G	U_{G_C}	S_C
Grids 1-3	0.323	4.53	3.15	7.90%	4.69%	3.20%	4.50

Table 6.6: Validation of Total Resistance C_T Obtained with Free Attitude Simulations at $Fn_\nabla = 5.72$

	E (%D)	U_V (%D)	U_D (%D)	U_{SN} (%D)
$E = D - S$	6.53	7.39	0.32	7.38
$E_C = D - S_C$	10.9	3.01	0.32	2.99

dynamic mesh because of the large computational cost and small improvement of the results. In the process of running the simulations, the 4 million mesh simulation took far more than 337.86 hours to compute due to the nodes and resource distribution of Cascades cluster. The simulations need to be manually restarted due to the maximum wall time is three days of Cascades cluster. The simulations gained very low priority each time it is submitted to the cluster for using so many processors, therefore it stays in the queue for a long time before execution. The four-million-cell mesh simulation actually takes more than 1 month to finish due to these uncontrollable reasons. Thus, grid 3 is not chosen for further computation because of the computational cost reasons.

6.3 Free Attitude Simulation Results at $Fn_\nabla = 5.19$

The free body simulations at V = 11.11 m/s, corresponding to $Fn_\nabla = 5.19$ use the dynamic coarse and medium mesh, and solutions obtained are summarized in Table 6.9 and compared in Figure 6.8.

It's unusual for the medium mesh simulation to have more numerical ventilation at hull bottom than the coarse mesh. The numerical ventilation for medium mesh presented in

Table 6.7: Estimated Solution for Free Attitude Simulation Using Richardson Extrapolation ($\times 10^{-3}$)

	Grid 3	Grid 2	Grid 1	Experimental Results
Simulation C_T	4.22	4.59	4.72	5.05
Estimated C_T	—	—	4.79	—

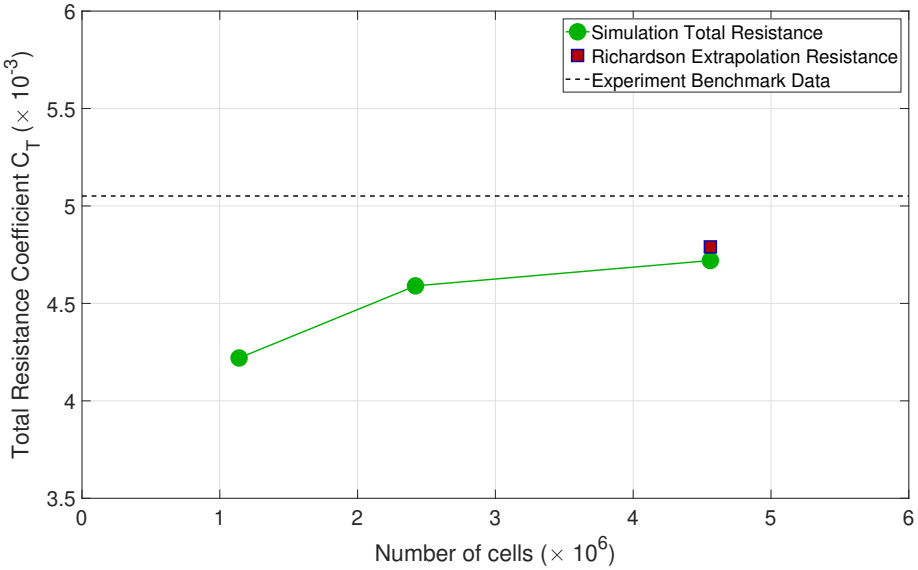


Figure 6.6: Estimated Results for Free Attitude Simulation Using Richardson Extrapolation

Figure 6.8b is small and wide-spread on the entire wetted area. It might result from the model, which has been already rotated to the experiment steady state before imported into OpenFOAM. And this angle might make the hull bottom mesh completely different from the highest speed, which contributes to the inaccuracy of the calculation for VOF. The $y+$ plots use the same range [0,240] this time and $y+$ value drops about 100 as increasing the refinement.

Table 6.8: Computational Cost for the Three Free Attitude Simulations at $Fn_\nabla = 5.72$

Grid	Grid 3 (1 million cells)	Grid 2 (2 million cells)	Grid 1 (4 million cells)
Total Resistance Relative Error %D	-18.3	-9.57	-6.61
S_G/S_{G_3}	1.00	0.520	0.360
Execution Time [hours]	12.5	210	338
S_G/S_{G_3}	1	16.7	27.0
Processors	192	192	192

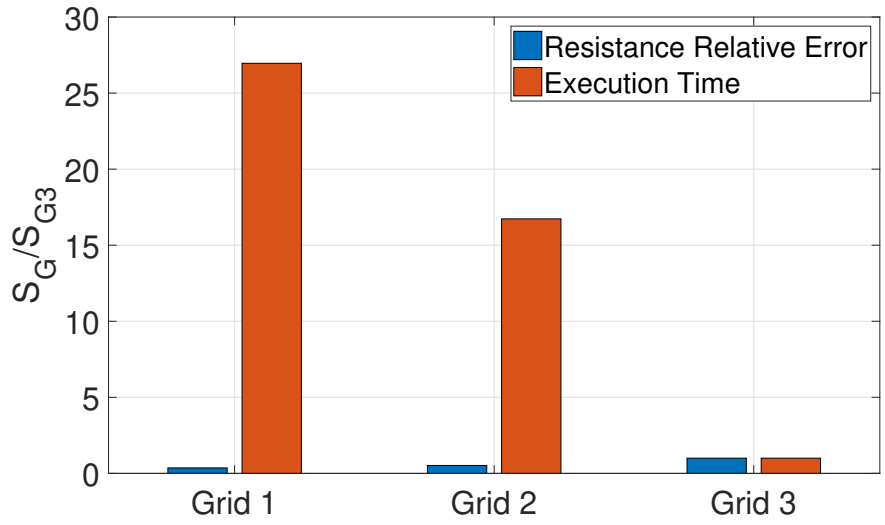
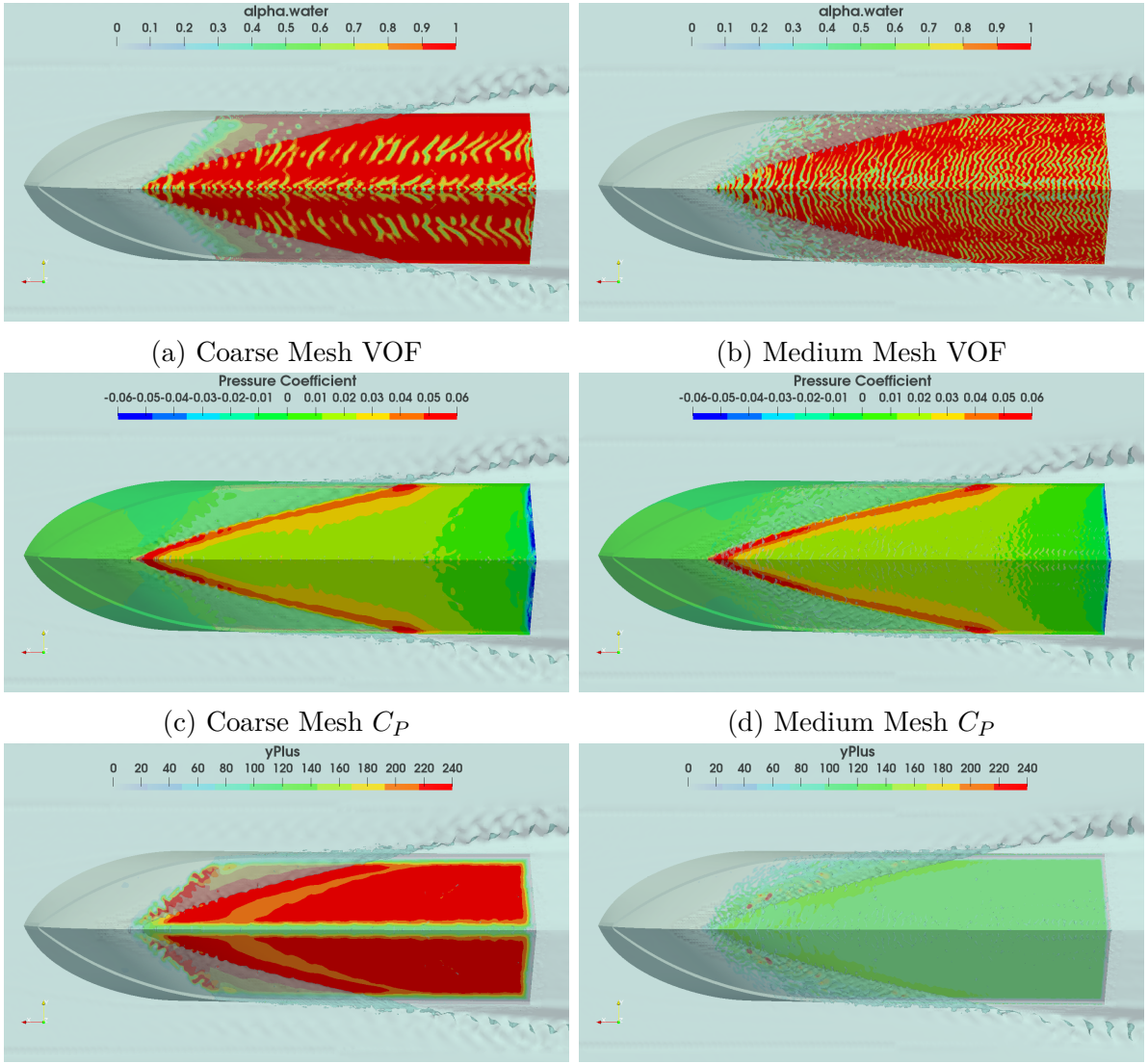


Figure 6.7: Computation Cost and Resistance Relative Error Comparison Plot at $Fn_\nabla = 5.72$ for Free Attitude Simulations

6.4 Free Attitude Simulation Results at $Fn_\nabla = 4.69$

The results of free attitude simulations at V = 10.03 m/s, corresponding to $Fn_\nabla = 4.69$, are summarized in Table 6.10 and compared in Figure 6.9 in terms of VOF, pressure coefficient and $y+$. As the hull moving slower, we can see the large decrease in friction drag and increase in pressure.

At this speed, the hull bottom numerical ventilation reduces a lot by increasing the mesh resolution, the medium mesh only have a small area of ventilation along the keel (Figure 6.9b)



(a) Coarse Mesh VOF

(b) Medium Mesh VOF

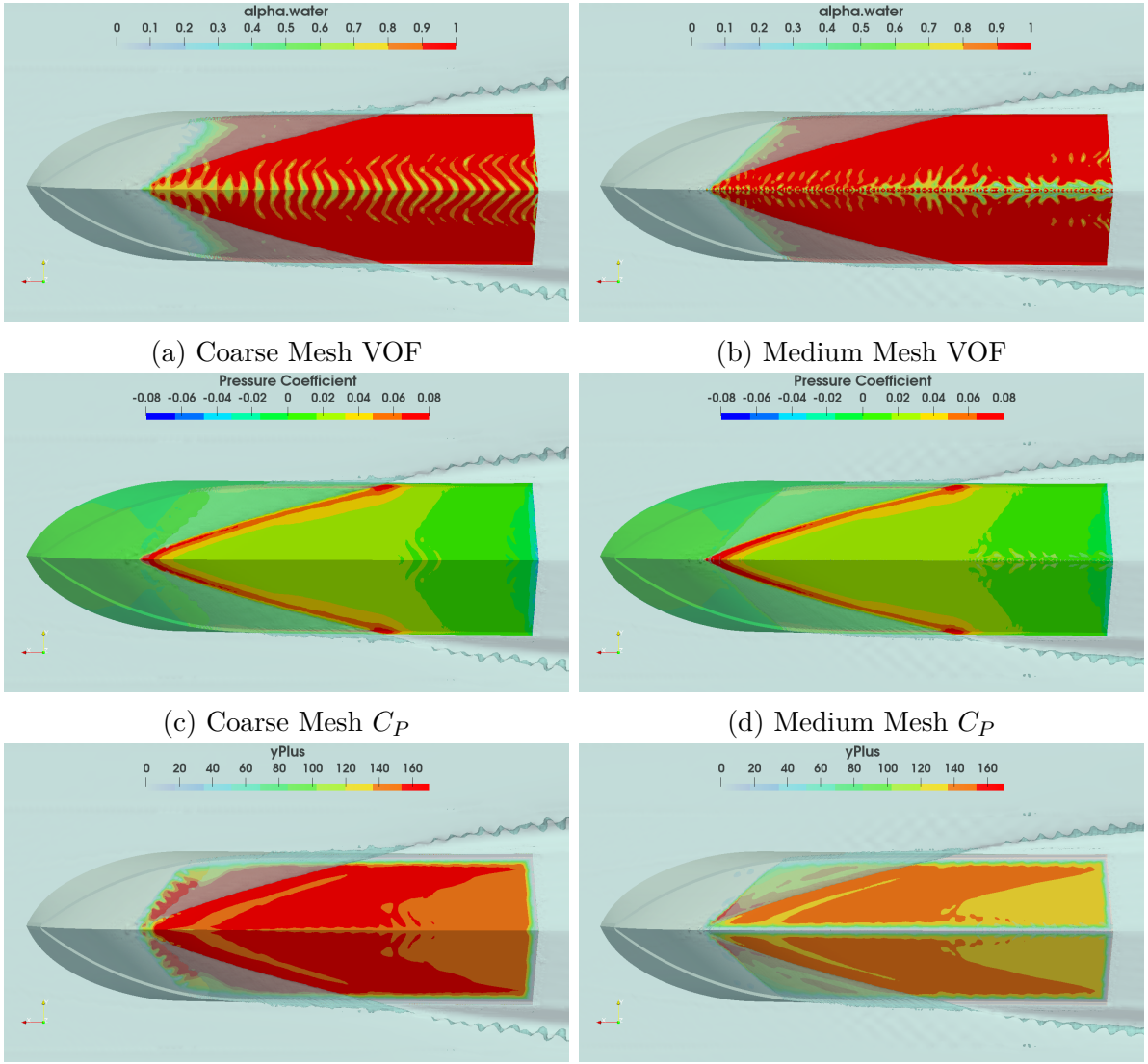
(c) Coarse Mesh C_P

(d) Medium Mesh C_P

(e) Coarse Mesh $y+$

(f) Medium Mesh $y+$

Figure 6.8: Numerical Solution Obtained Using Two Different Grids Described in Table 6.9. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at $\mathrm{Fn}_\nabla = 5.19$.



(a) Coarse Mesh VOF

(b) Medium Mesh VOF

(c) Coarse Mesh C_P

(d) Medium Mesh C_P

(e) Coarse Mesh $y+$

(f) Medium Mesh $y+$

Figure 6.9: Numerical Solution Obtained Using Two Different Grids Described in Table 6.10. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at $Fn_\nabla = 4.69$.

Table 6.9: Numerical Results for Free Attitude Simulations Obtained with Two Dynamic Meshes at $Fn_{\nabla} = 5.19$

	Coarse Mesh	Relative Error (%)	Medium Mesh	Relative Error (%)	Experimental Result	Savitsky Short Method
Mesh (Number of Cells)	1541478	—	2718115	—	—	—
Resistance [N]	201	14.4	203	13.6	235	206
Friction Drag [N]	137	—	139	—	—	151
Pressure [N]	64.1	—	64.0	—	—	54.9
Trim Angle [degree]	2.47	2.66	2.51	1.09	2.54	3.14
Heave [cm]	9.96	5.14	9.90	4.51	9.47	10.1
L_C [cm]	54.0	3.38	53.5	4.28	55.9	57.5
L_K [cm]	168	1.46	169	0.890	171	168

Table 6.10: Numerical Results for Free Attitude Simulations Obtained with Two Dynamic Meshes at $Fn_{\nabla} = 4.69$

	Coarse Mesh	Relative Error (%)	Medium Mesh	Relative Error (%)	Experimental Result	Savitsky Short Method
Mesh (Number of Cells)	1385778	—	2567245	—	—	—
Resistance [N]	191	7.64	193	6.67	207	184
Friction Drag [N]	124	—	125	—	—	120
Pressure [N]	67.3	—	67.9	—	—	63.9
Trim Angle [degree]	3.06	3.39	3.09	4.39	2.96	3.67
Heave [cm]	9.59	3.42	9.39	1.26	9.27	9.88
L_C [cm]	61.7	8.36	64.4	4.35	67.3	66.4
L_K [cm]	167	0.880	170	0.530	169	163

and the simulated resistance is close to the experimental benchmark data. The hull bottom pressure coefficient increases comparing with the highest speed pressure coefficient and it's in accordance with the increase in pressure resistance (Table 6.10). Meanwhile, unlike $y+$ solution change through refinement in 11.11 m/s and 12.24 m/s free body simulations, $y+$ doesn't decrease so much when adding more cells.

6.5 Free Attitude Simulation Results at $Fn_{\nabla} = 4.21$

The free attitude simulation results for V = 9 m/s, which corresponds $Fn_{\nabla} = 4.21$, are summarized in Table 6.11 and also compared in Figure 6.10. Trim angle increases with the

decrease of hull speed, the hull also have more sinkage than higher speeds.

Table 6.11: Numerical Results for Free Attitude Simulations Obtained with Two Dynamic Meshes at $Fn_\nabla = 4.21$

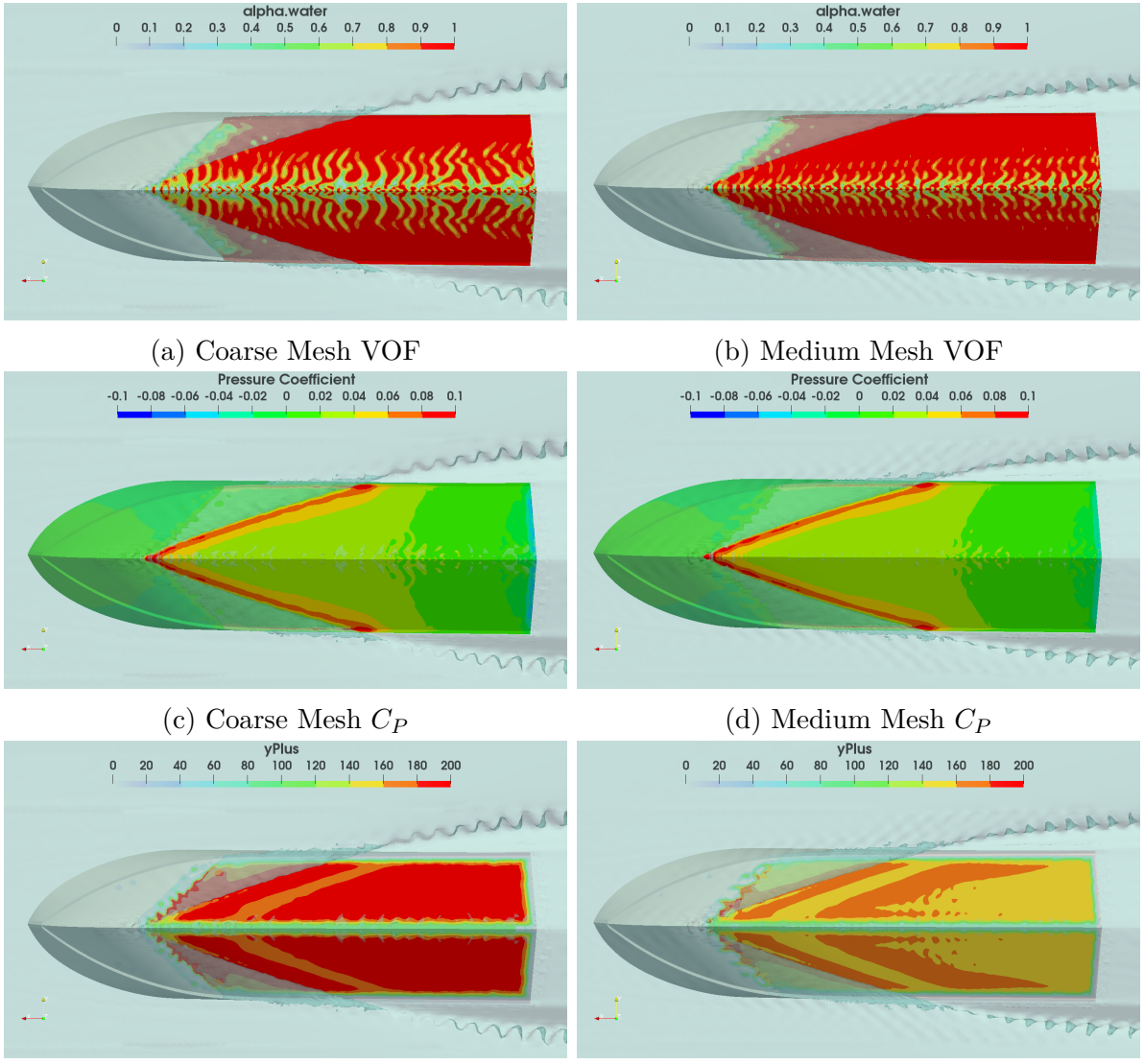
	Coarse Mesh	Relative Error (%)	Medium Mesh	Relative Error (%)	Experimental Result	Savitsky Short Method
Mesh (Number of Cells)	1239609	—	2312403	—	—	—
Resistance [N]	166	9.99	170	7.82	184	171
Friction Drag [N]	92.9	—	98.3	—	—	97.1
Pressure [N]	73.1	—	71.7	—	—	74.1
Trim Angle [degree]	3.49	1.99	3.62	5.79	3.42	4.25
Heave [cm]	9.40	3.75	9.25	2.09	9.06	9.70
L_C [cm]	70.9	6.99	71.8	5.74	76.2	71.4
L_K [cm]	165	0.610	168	1.14	166	154

In Table 6.11 and Figure 6.10, the dynamic medium mesh yields better results in total resistance and heave. In addition, the medium mesh simulation also has less numerical ventilation at hull bottom and smaller $y+$ value than coarse mesh.

6.6 Free Attitude Simulation Results at $Fn_\nabla = 3.56$

Similarly, free attitude simulations at speed V = 7.61 m/s, corresponding to $Fn_\nabla = 3.56$ also use the coarse and medium mesh. Results obtained are summarized in Table 6.12 and plotted in Figure 6.11. The hull is moving at a semi-planing regime and pressure gradually becomes the dominant component instead of friction drag in total resistance.

Figure 6.11a and 6.11b provide a clear view of difference in numerical ventilation, which is not reduced on the hull bottom by more refined grids. It's very hard to say that the medium mesh simulation has total area of numerical ventilation larger than the coarse mesh simulation since the numerical ventilation is more scattering on the bottom surface instead of aggregated along the keel. The $y+$ for medium mesh just decreases by approximately 20 than the coarse mesh. Therefore, the medium mesh free body simulation doesn't greatly



(a) Coarse Mesh VOF

(b) Medium Mesh VOF

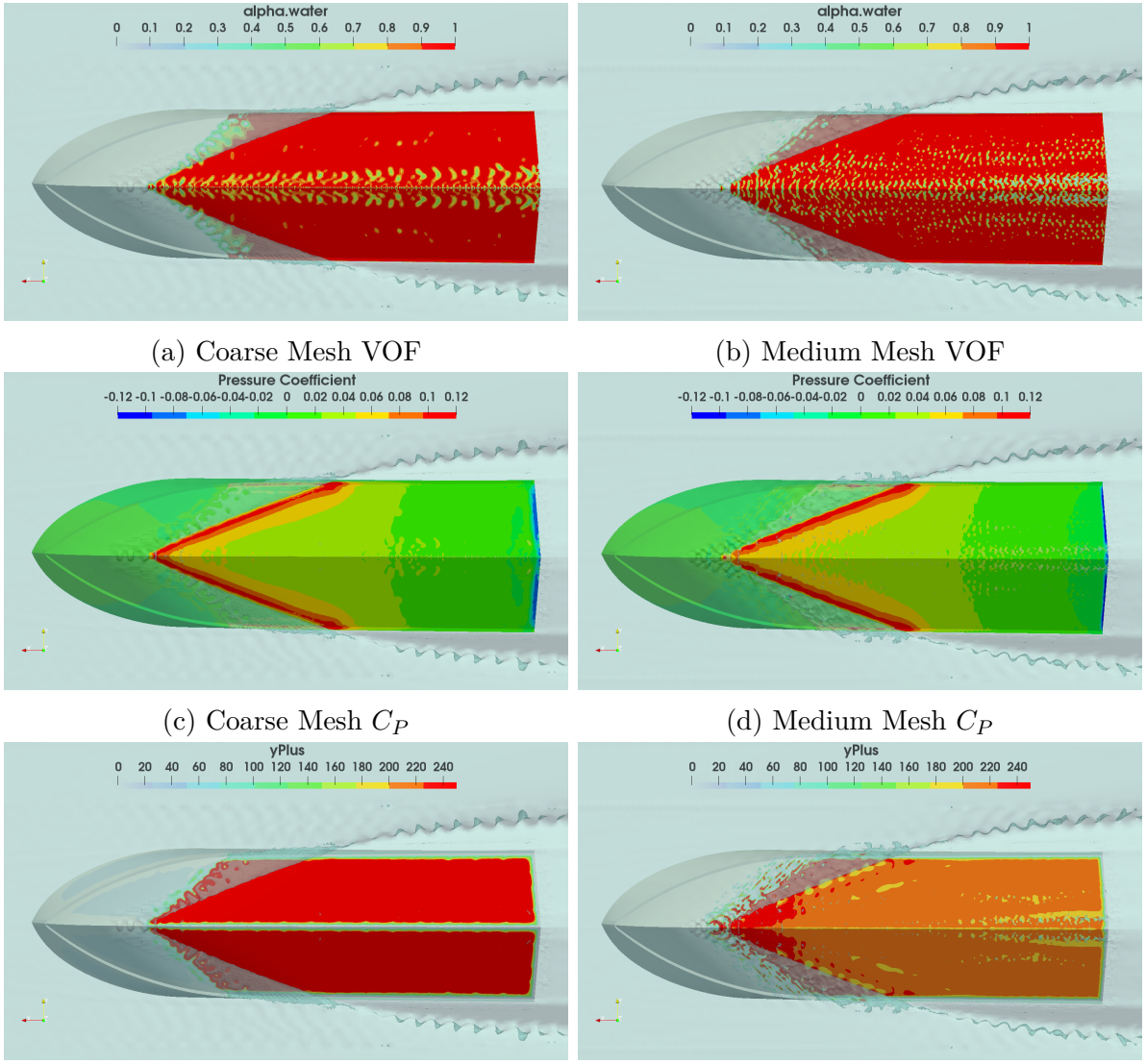
(c) Coarse Mesh C_P

(d) Medium Mesh C_P

(e) Coarse Mesh $y+$

(f) Medium Mesh $y+$

Figure 6.10: Numerical Solution Obtained Using Two Different Grids Described in Table 6.11. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at $\mathrm{Fn}_\nabla = 4.21$.



(a) Coarse Mesh VOF

(b) Medium Mesh VOF

(c) Coarse Mesh C_P

(d) Medium Mesh C_P

(e) Coarse Mesh $y+$

(f) Medium Mesh $y+$

Figure 6.11: Numerical Solution Obtained Using Two Different Grids Described in Table 6.12. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at $Fn_\nabla = 3.56$.

Table 6.12: Numerical Results for Free Attitude Simulations Obtained with Two Dynamic Meshes at $Fn_\nabla = 3.56$

	Coarse Mesh	Relative Error (%)	Medium Mesh	Relative Error (%)	Experimental Result	Savitsky Short Method
Mesh (Number of Cells)	1396575	—	2139392	—	—	—
Resistance [N]	151	8.48	154	6.82	165	160
Friction Drag [N]	66.7	—	69.6	—	—	70.5
Pressure [N]	84.3	—	84.4	—	—	89.3
Trim Angle [degree]	4.45	5.47	4.33	2.63	4.22	5.12
Heave [cm]	8.81	3.72	8.65	1.84	8.49	9.00
L_C [cm]	82.1	7.64	85.1	4.27	88.9	80.8
L_K [cm]	163	0.810	165	0.450	164	149

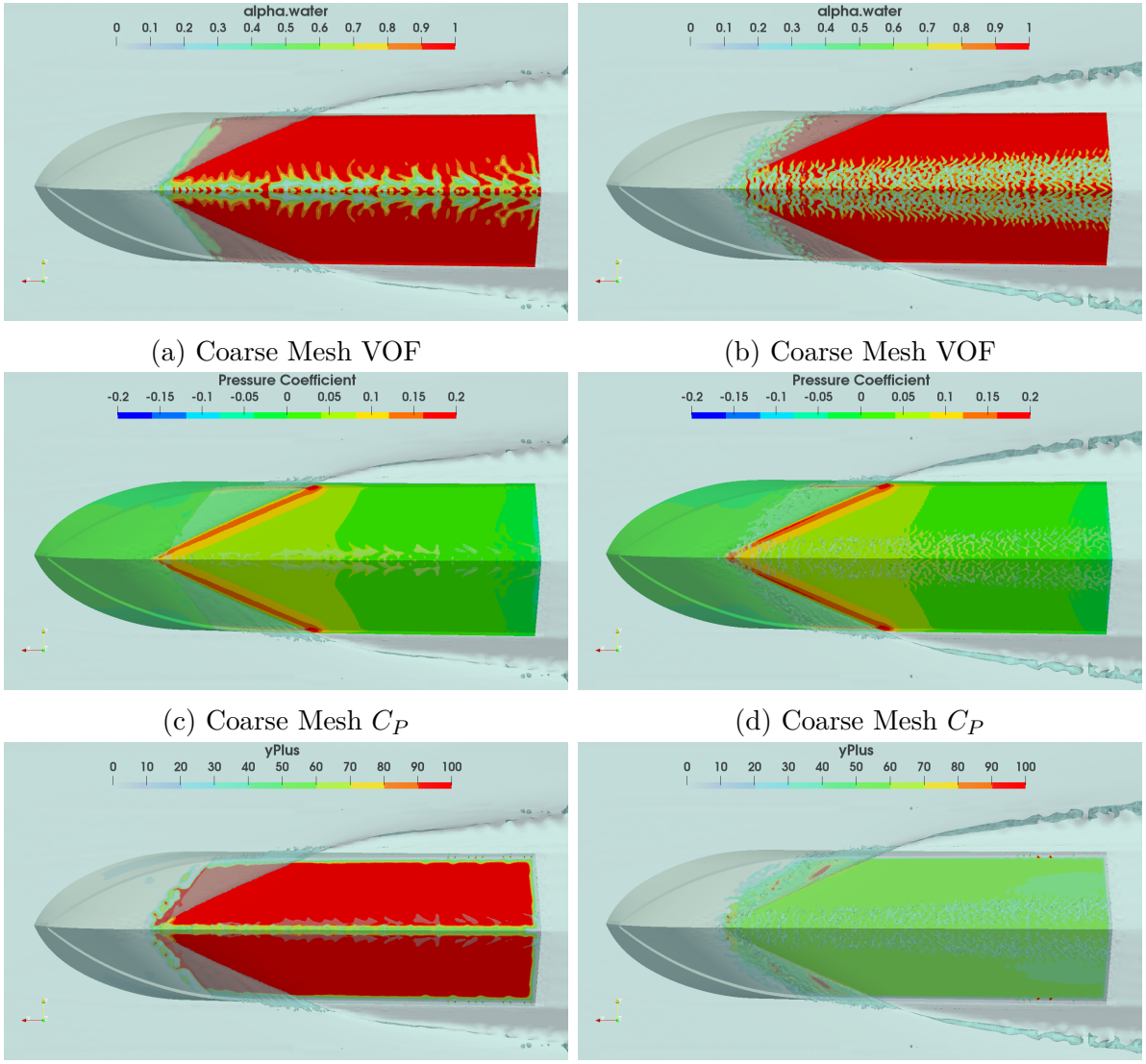
improve the solutions at this speed.

6.7 Free Attitude Simulation Results at $Fn_\nabla = 3.13$

When the hull is advancing at speed V = 6.69 m/s, which corresponds $Fn_\nabla = 3.13$, the hull is almost getting into the pre-planing ($Fn_\nabla < 3$) regime. Table 6.13 summarizes the free attitude simulation results using the coarse and medium mesh. The results are also well compared in Figure 6.12. The pressure resistance is almost 2 times as large as friction drag at this speed.

Table 6.13: Numerical Results for Free Attitude Simulations Obtained with Two Dynamic Meshes at $Fn_\nabla = 3.13$

	Coarse Mesh	Relative Error (%)	Medium Mesh	Relative Error (%)	Experimental Result	Savitsky Short Method
Mesh (Number of Cells)	1368457	—	2243883	—	—	—
Resistance [N]	149	6.49	150	6.30	160	161
Friction Drag [N]	54.3	—	53.1	—	—	55.4
Pressure [N]	94.9	—	96.4	—	—	105
Trim Angle [degree]	5.13	4.81	5.01	2.66	4.88	6.03
Heave [cm]	8.00	2.10	8.06	1.35	8.17	8.46
L_C [cm]	93.8	4.10	95.3	2.58	97.8	89.4
L_K [cm]	161	0.840	163	0.280	163	149



(a) Coarse Mesh VOF

(b) Coarse Mesh VOF

(c) Coarse Mesh C_P

(d) Coarse Mesh C_P

(e) Coarse Mesh $y+$

(f) Coarse Mesh $y+$

Figure 6.12: Numerical Solution Obtained Using Two Different Grids Described in Table 6.13. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at $Fn_\nabla = 3.13$.

It's very unusual for the medium mesh simulation that the numerical ventilation at the hull bottom actually increases but the results happen to be a little better than the coarse mesh. However, the little improvement can be negligible due to the error caused by averaging the last 5 oscillations and it could be even worse than the coarse mesh results. The drop in friction drag corresponds to the increase of numerical ventilation at the bottom. The only thing that medium mesh changes is the $y+$ value, which decreases by 50 from the coarse mesh. In general, the medium mesh simulation doesn't improve the solution with more grids.

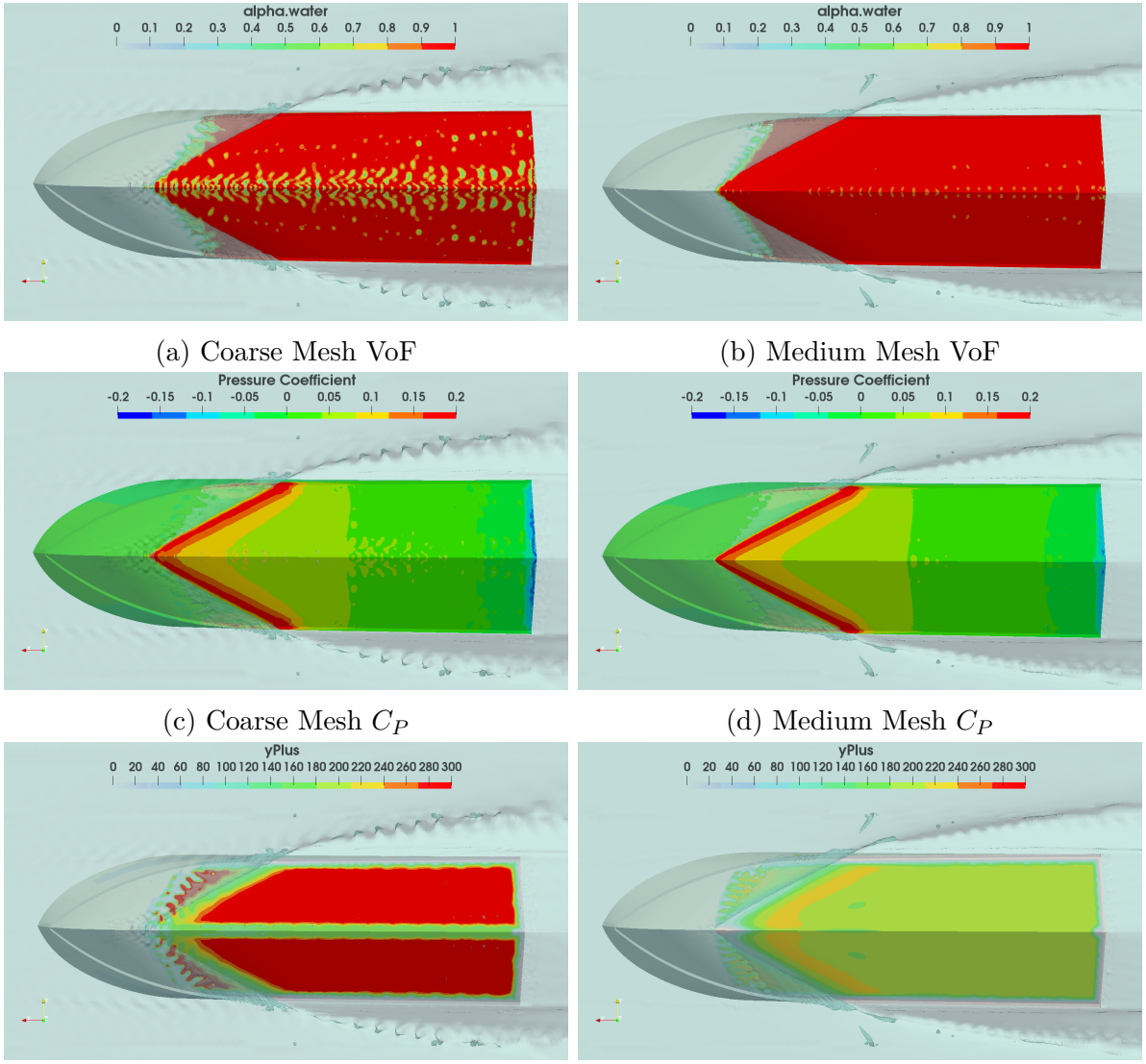
6.8 Free Attitude Simulation Results at $Fn_\nabla = 2.60$

The lowest speed is V = 5.56 m/s, corresponding to $Fn_\nabla = 2.60$ and the hull is at pre-planing regime. Free body simulations at this speed also use the coarse and medium mesh and results obtained are summarized in Table 6.14 and compared in Figure 6.13. As shown in Table 6.14, pressure is almost 3 times as large as friction drag in the simulation results.

Table 6.14: Numerical Results for Free Attitude Simulations Obtained with Two Dynamic Meshes at $Fn_\nabla = 2.60$

	Coarse Mesh	Relative Error (%)	Medium Mesh	Relative Error (%)	Experimental Result	Savitsky Short Method
Mesh (Number of Cells)	1226229	—	2125594	—	—	—
Resistance [N]	151	6.46	154	4.60	161	162
Friction Drag [N]	35.3	—	40.2	—	—	40.6
Pressure [N]	116	—	114	—	—	122
Trim Angle [degree]	6.08	4.73	5.88	1.29	5.81	6.97
Heave [cm]	6.70	10.1	6.85	8.08	7.45	7.44
L_C [cm]	105	3.94	106	3.29	109	107
L_K [cm]	161	2.64	163	1.49	165	158

By comparing Figure 6.13a and 6.13b, the medium mesh simulation did a good job in reducing the numerical ventilation at the hull bottom. Also, Table 6.14 indicates that all the other results are improved by using increasing the grid resolution. Taking a glance at



(a) Coarse Mesh VoF

(b) Medium Mesh VoF

(c) Coarse Mesh C_P

(d) Medium Mesh C_P

(e) Coarse Mesh $y+$

(f) Medium Mesh $y+$

Figure 6.13: Numerical Solution Obtained Using Two Different Grids Described in Table 6.14. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at $Fn_\nabla = 2.60$.

all the VOF plots of free body simulations at seven speeds, the medium mesh simulation at 5.56 m/s is actually the best simulation that yield the least numerical ventilation. The non-dimensional wall distance $y+$ decreases from about 300 in the coarse mesh to 200 in the medium mesh.

In short, increasing the grid refinement doesn't guarantee better solutions for free attitude simulations. The free attitude simulation results are largely depends on the speeds and the model itself. Since the seven models are rotated and translated to seven corresponding speeds' steady state, respectively, the mesh could be changed a lot even using the same mesh strategy.

6.9 Summary of Simulations at All Speeds

After the separate analysis for free attitude simulation results described in previous sections, it's imperative to get a overview of all numerical solutions together with experimental benchmark data and empirical calculations. The results of all speeds' free attitude simulations are compared with the experimental benchmark data from NSWCCD tests and Savitsky Short Method in terms of total resistance over displacement $\frac{R}{\Delta}$, heave, trim, wetted chine length, and wetted keel length. As the focus in this study, total resistance is compared in terms of $\frac{R}{\Delta}$, which is calculated with $\Delta = 101.5kg$ and summarized in Table 6.15. For the purpose of more intuitive comparison, $\frac{R}{\Delta}$ is also plotted as Figure 6.14 with experimental data and Savitsky Short Method resisitance.

From Table 6.15 and Figure 6.14, the medium mesh simulations generally provide better total resistance solutions than coarse mesh simulations. However, at $Fn_{\nabla} = 5.19$, the medium mesh results don't follow the general trend, so does the coarse mesh results, which indicates that the simulations at 11.11 m/s fail to give satisfying results and the meshing strategy

Table 6.15: $\frac{R}{\Delta}$ for All Speeds' Free Attitude Simulations

Fn_∇	Speed [m/s]	Coarse Mesh (1 million cells)	Medium Mesh (2 million cells)	Fine Mesh (4 million cells)	Experimental Result	Savitsky Short Method
5.72	12.24	2.17	2.40	2.48	2.66	2.28
5.19	11.11	1.98	2.00	—	2.31	2.03
4.69	10.03	1.88	1.90	—	2.04	1.81
4.21	9.00	1.64	1.67	—	1.82	1.69
3.56	7.61	1.49	1.52	—	1.63	1.57
3.13	6.69	1.47	1.47	—	1.57	1.58
2.60	5.56	1.49	1.52	—	1.59	1.60

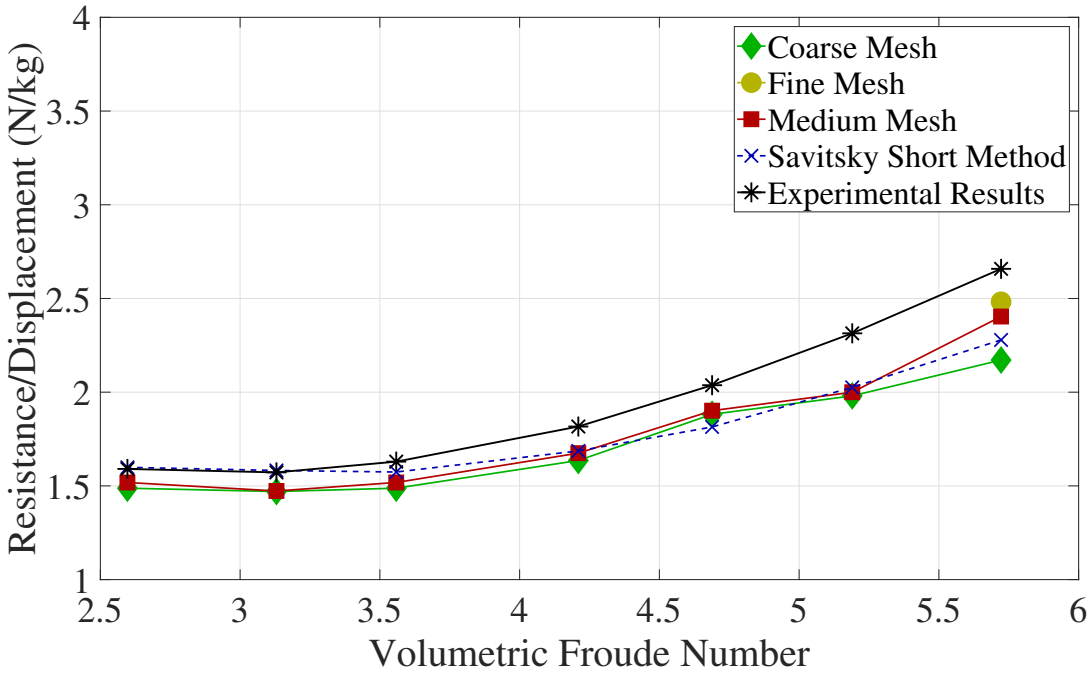


Figure 6.14: All Speeds' Free Attitude Simulations $\frac{R}{\Delta}$

doesn't work so well at this speed. The fine mesh result shown as a single dot at Fn_∇ = 5.72, is the best one among all the highest-speed simulation results. However, the fine mesh simulation is only conducted at the highest speed due to large computational cost, thus, there is no way to know how the fine mesh works at other speeds. Savitsky Short Method gives very good estimations at low speeds, i.e., $Fn_\nabla = 2.6$ and $Fn_\nabla = 3.13$, which corresponds to 5.56 m/s and 6.69 m/s, respectively. With the increase of Fn_∇, Savitsky Method no longer yields satisfying estimations. It can be also observed that the Savitsky

Method generally gives better estimations than the coarse mesh simulations, which requires much more computational effort than Savitsky's equations. Therefore, more refinement or a new mesh strategy is needed in future study to improve the simulation results.

Trim angle is summarized in Table 6.16 and plotted as Figure 6.15. Figure 6.15 provides us a good view of all trim angle results including trim angles from simulations, Savitsky Short Method and experiments. In general, both coarse mesh and medium mesh simulations yield good results for trim. Savitsky Method gives high values of trim and the estimations are far from the experimental results. The medium mesh simulations have trim angles close to the experiment and are generally better than the coarse mesh results especially at low speeds. At $Fn_\nabla = 5.19$ and $Fn_\nabla = 5.72$, medium mesh trim is almost the same as experimental trim and fine mesh trim at$Fn_\nabla = 5.72$ also overlaps the experiment and medium mesh trim. Coarse mesh simulation at $Fn_\nabla = 5.72$ yields a much smaller trim angle than the experiment.

Table 6.16: Trim Angle (degrees) for All Speeds' Free Attitude Simulations

Fn_∇	Speed [m/s]	Coarse Mesh (1 million cells)	Medium Mesh (2 million cells)	Fine Mesh (4 million cells)	Experimental Result	Savitsky Short Method
5.72	12.24	1.87	2.14	2.19	2.17	2.84
5.19	11.11	2.47	2.51	—	2.54	3.14
4.69	10.03	3.06	3.09	—	2.96	3.67
4.21	9.00	3.49	3.62	—	3.42	4.25
3.56	7.61	4.45	4.33	—	4.22	5.12
3.13	6.69	5.13	5.01	—	4.88	6.03
2.60	5.56	6.08	5.88	—	5.81	6.97

Heave (CoG rise) values for all free body simulations are summarized in Table 6.17 and plotted in Figure 6.16 with Savitsky Method heave and experimental heave results. For heave calculations, Savitsky Method is also too much higher than the experimental results. Both coarse mesh and medium mesh simulations have smaller heave at low speeds $Fn_\nabla = 2.60$ and $Fn_\nabla = 3.13$ while simulated heave is higher than the experimental benchmark

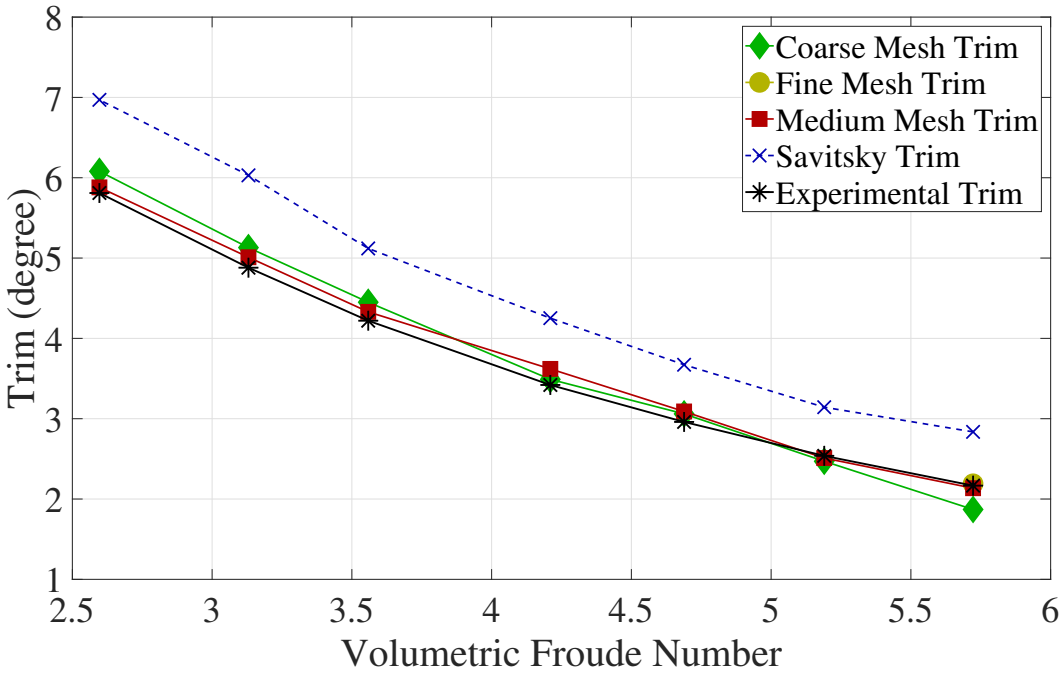


Figure 6.15: Trim Angle of All Speeds' Free Attitude Simulations

data at higher speeds. Also, the difference between the simulations and experimental data increases with the increase of speed. Generally speaking, the medium mesh simulations yield the best results than the other predictions.

Table 6.17: Heave (cm) for All Speeds' Free Attitude Simulations

Fn_∇	Speed [m/s]	Coarse Mesh (1 million cells)	Medium Mesh (2 million cells)	Fine Mesh (4 million cells)	Experimental Result	Savitsky Short Method
5.72	12.24	10.3	9.99	9.89	9.60	10.3
5.19	11.11	9.96	9.90	—	9.47	10.1
4.69	10.03	9.59	9.39	—	9.27	9.88
4.21	9.00	9.40	9.25	—	9.06	9.70
3.56	7.61	8.81	8.65	—	8.49	9.00
3.13	6.69	8.00	8.06	—	8.17	8.46
2.60	5.56	6.70	6.85	—	7.45	7.44

Wetted chine length results of free attitude simulations, Savitsky Method and benchmark data are shown in Table 6.18 and compared in Figure 6.17. Similarly, Wetted keel length results of free attitude simulations, Savitsky Method and benchmark data are summarized

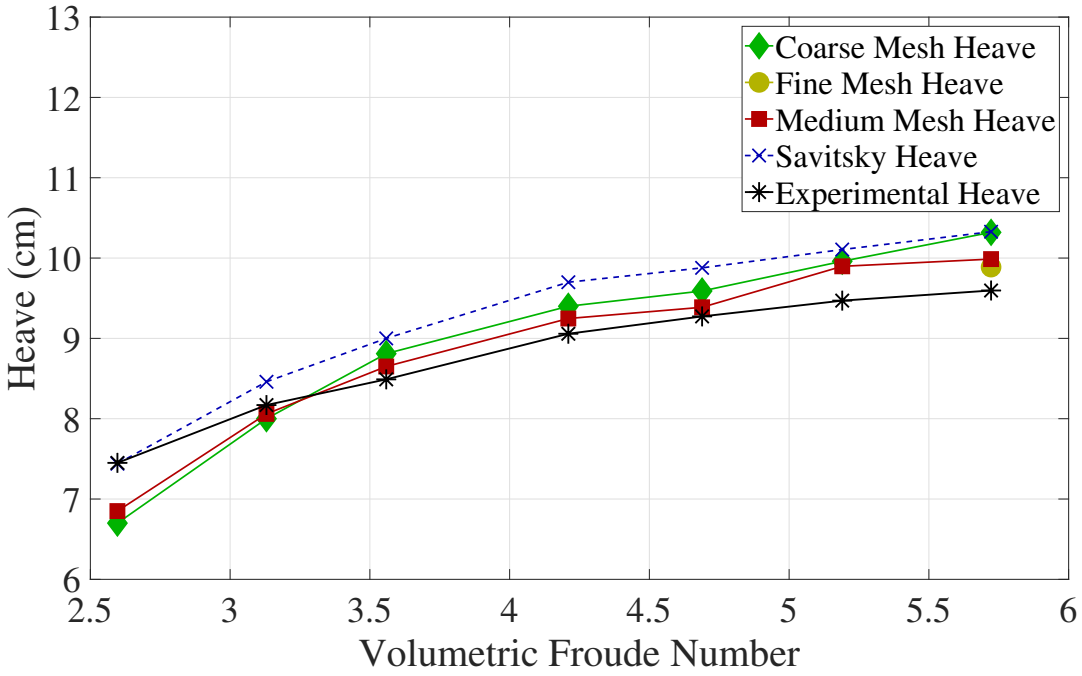


Figure 6.16: Heave (CoG rise) of All Speeds' Free Attitude Simulations

in Table 6.19 and plotted in Figure 6.18.

Table 6.18: Wetted Chine Length (cm) for All Speeds' Free Attitude Simulations

Fn_∇	Speed [m/s]	Coarse Mesh (1 million cells)	Medium Mesh (2 million cells)	Fine Mesh (4 million cells)	Experimental Result	Savitsky Short Method
5.72	12.24	43.5	43.1	43.0	44.5	50.5
5.19	11.11	54.0	53.5	—	55.9	57.5
4.69	10.03	61.7	64.4	—	67.3	66.4
4.21	9.00	70.9	71.8	—	76.2	71.4
3.56	7.61	82.1	85.1	—	88.9	80.8
3.13	6.69	93.8	95.3	—	97.8	89.4
2.60	5.56	105	106	—	109	107

As shown in figure 6.17, the three different mesh free body simulations yield wetted chine length close to the experiment while Savitsky Method estimations are a little away. The medium mesh simulations' wetted chine length is generally higher than the coarse mesh and closer to the experimental benchmark data.

For the wetted keel length, both coarse mesh and medium mesh simulations results are very

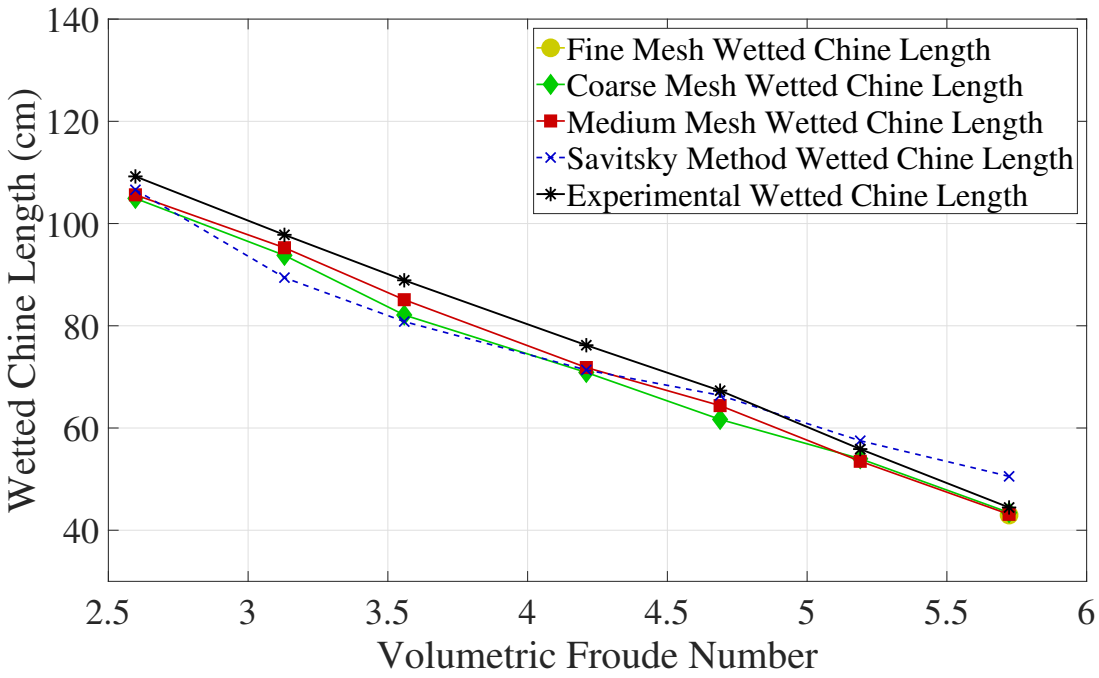


Figure 6.17: Wetted Chine Length of All Speeds' Free Attitude Simulations

close to the experiment while Savitsky Method only yields satisfying results at high speeds, i.e., $Fn_{\nabla} = 5.19$ and $Fn_{\nabla} = 5.72$. The simulation wetted chine length and wetted keel length are measured in *paraView* by plotting the pressure and the contour plot of the free surface. Thus, it may not be the exact value due to the error and uncertainty of measurement.

In conclusion, medium mesh free body simulations at 7 different speeds yield better results than the coarse mesh simulations in total resistance, trim angle, heave, wetted chine length and wetted keel length. Since the coarse mesh simulations total resistance results are no better than the Savitsky Method estimations, it shouldn't be considered for future simulations. All simulations at the same speed have very similar pressure resistance and different friction drag. The variation of friction drag depends largely on the ventilation on the hull bottom. The main focus of improving the simulation solutions is reducing the numerical ventilation and increase the accuracy of solving for friction drag. Even it's very hard to completely eliminate the ventilation, ventilation can still be reduced by increasing the resolution of the

Table 6.19: Wetted Keel Length (cm) for All Speeds' Free Attitude Simulations

Fn_∇	Speed [m/s]	Coarse Mesh (1 million cells)	Medium Mesh (2 million cells)	Fine Mesh (4 million cells)	Experimental Result	Savitsky Short Method
5.72	12.24	169	173	174	173	175
5.19	11.11	168	169	—	171	168
4.69	10.03	167	170	—	169	163
4.21	9.00	165	168	—	166	154
3.56	7.61	163	165	—	164	149
3.13	6.69	161	163	—	163	149
2.60	5.56	161	163	—	165	158

hull bottom and prism layers. Further study should start on the basis of medium mesh and increase the resolution at the hull bottom and chine area where the free surface undergoes violent deformations.

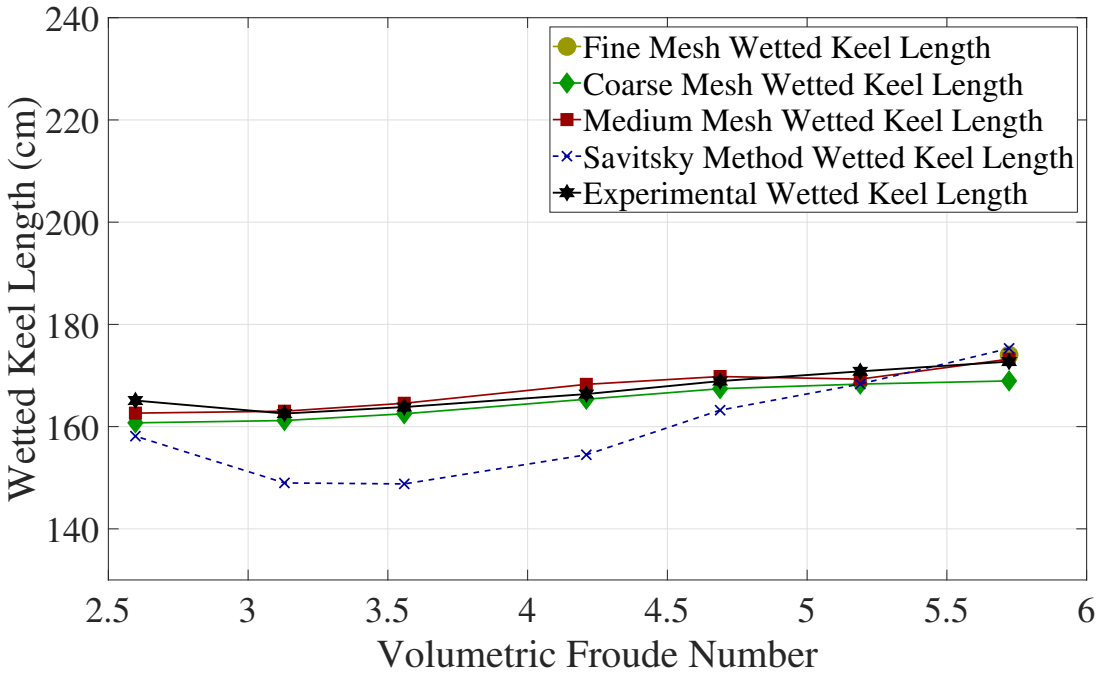


Figure 6.18: Wetted Keel Length of All Speeds' Free Attitude Simulations

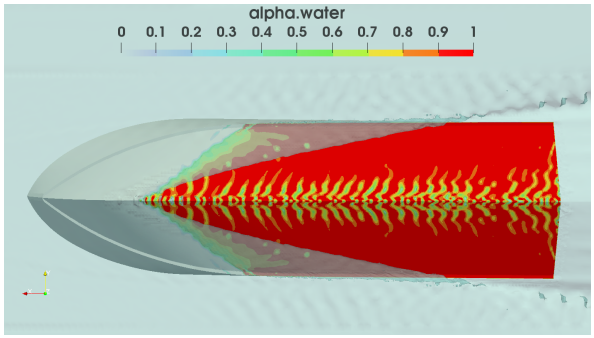
Chapter 7

Simulation Improvements

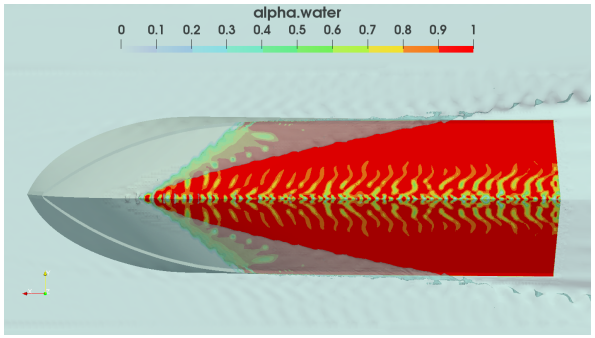
7.1 Increase the Pressure-momentum Coupling Loops

In OpenFOAM *fvSolution*, if set the *nOuterCorrectors* larger than 1, the PIMPLE loop would be activated and the pressure-momentum coupling would be calculated the specified value times in one single time step. In order to check the effect from the change of *nOuterCorrectors* defined in *PIMPLE*, we use the coarse mesh (1 million cells) at the highest speed to test, since it is very steady and can complete for shorter time. Start from the original coarse mesh simulation at the highest speed $V = 12.24m/s$, only change the *nOuterCorrectors* under *PIMPLE* in *fvSolution* from 1 to 3. The results for this new simulation are compared with the original coarse mesh simulation. The plots for VOF, pressure coefficient and $y+$ are compared in Figure 7.1. The results comparison is summarized in Table 7.1 and the iterative history plot for resistance is shown as Figure 7.2. In Figure 7.1, it can be observed that the pressure coefficient plots and the $y+$ plots are almost the same, which indicates that the change in *nOuterCorrectors* has not obvious effects on the calculation for pressure and the non-dimensional wall distance. However, let's take a look at 7.1a and 7.1b, the modified case with high *nOuterCorrectors* has less ventilation on the bottom of the hull.

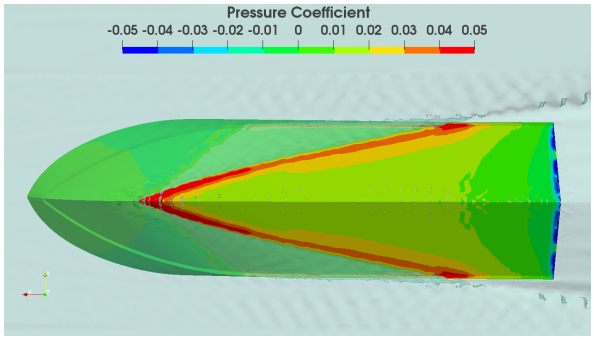
From Table 7.1, we can see that the relative error with respect to the experimental results are largely decreased in resistance and trim angle by 7.45% and 32.12%, respectively. All the calculated results for *nOuterCorrectors*=3 simulation is closer to the experiment data than



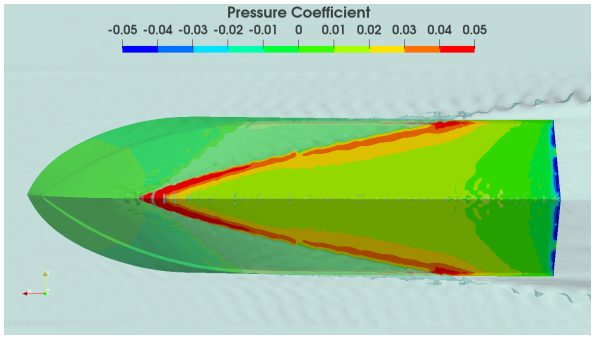
(a) Coarse Mesh VOF with *nOuterCorrectors*=1



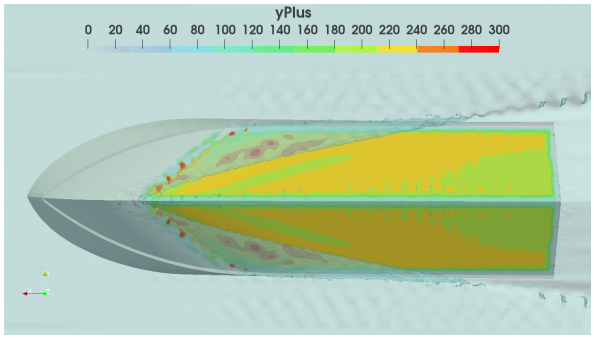
(b) Coarse Mesh VOF with *nOuterCorrectors*=3



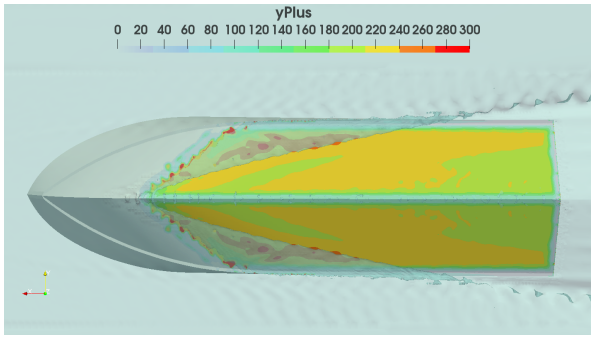
(c) Coarse Mesh C_P with *nOuterCorrectors*=1



(d) Coarse Mesh C_P with *nOuterCorrectors*=3



(e) Coarse Mesh $y+$ with *nOuterCorrectors*=1



(f) Coarse Mesh $y+$ with *nOuterCorrectors*=3

Figure 7.1: Comparison of Numerical Solution Between Original Coarse Mesh with *nOuterCorrectors*=3 and the Modified Coarse Mesh with *nOuterCorrectors*=1. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom, Steady-state Solution at highest speed $\mathrm{Fn}_\nabla = 5.72$

Table 7.1: Free Attitude Simulation Results Comparison Between Coarse Mesh with Different *nOuterCorrectors* Obtained at $Fn_{\nabla} = 5.72$

	nOuterCorrectors 1	Relative Error (%)	*nOuterCorrectors* 3	Relative Error (%)	Experimental Result
Resistance [N]	220.4	18.32	240.5	10.87	269.83
Friction Drag [N]	158.0	—	179.1	—	—
Pressure [N]	62.4	—	61.4	—	—
Trim Angle [degree]	1.87	34.15	2.03	5.70	2.17
Heave [cm]	10.32	7.54	10.01	4.27	9.60
L_C [cm]	43.45	2.25	43.62	1.87	44.45
L_K [cm]	168.94	2.19	170.90	1.82	172.72

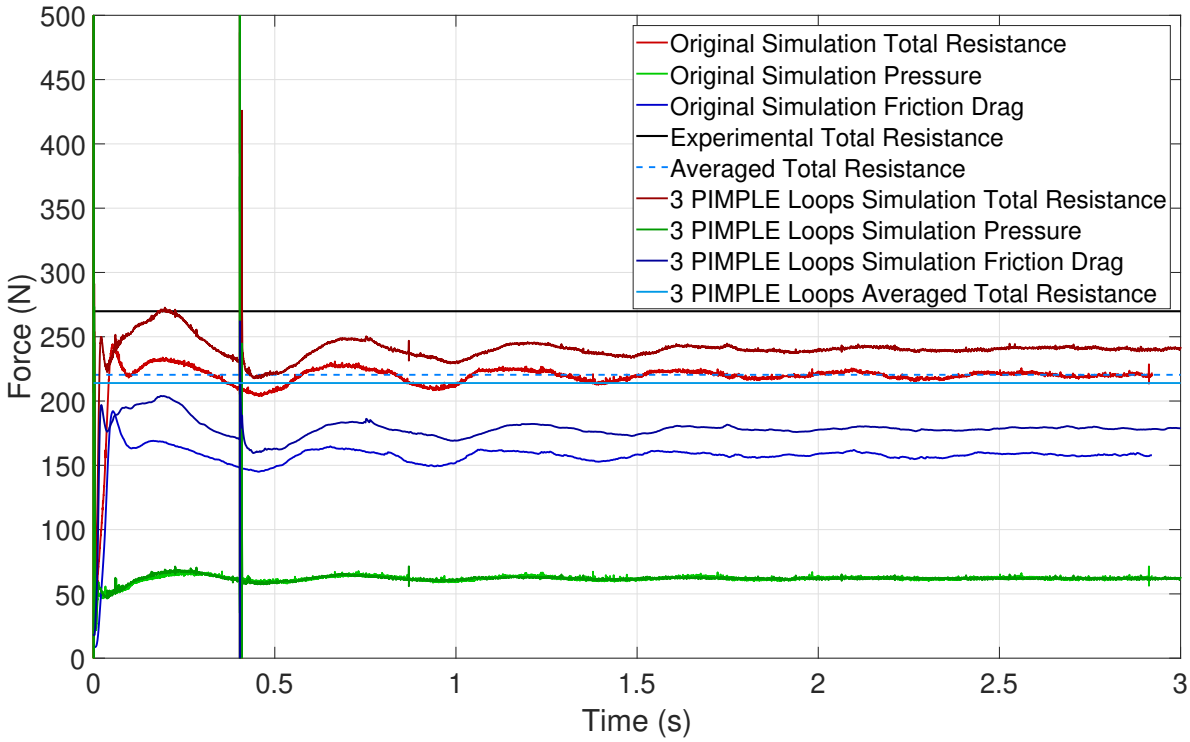


Figure 7.2: Time History of Different *nOuterCorrectors* Cases Resistance

nOuterCorrectors=1. Since the experimental result for friction drag and pressure resistance are unknown, the error can't be computed. However, there is a significant increase in friction drag by 13.35% of the original *nOuterCorrectors*=1 simulation result, which could be explained by the decrease of numerical ventilation on the bottom of the hull shown in Figure 7.1a and 7.1b. The increase of total resistance and friction drag is clearly shown in Figure

7.2, the pressure is almost the same for the two simualtions. The convergence history of the two cases are similar to each other. Moreover, the execution time for original *nOuterCorrectors*=1 is 209.61 hours while the modified *nOuterCorrectors*=3 case takes 429.70 hours, which is more than two times the execution time for *nOuterCorrectors*=1.

In conclusion, the increase of *nOuterCorrectors* can greatly improve the precision of simulation, but it also takes much more time to finish the simulation. For the medium mesh and fine mesh that already take a very long period to complete the simulation, increasing *nOuterCorrectors* might not be a good choice for high resolution grids considering the big computational cost.

7.2 Change Alpha Coefficients

By going through all the simulations, it's very easy to find that all the plots of VOF described as α_{water} in the plots, have different numerical ventilation on the hull bottom, indicating that α is solved incorrectly. The phase fraction α_{water} is used to describe the fraction of water and air. The definition is described in (7.1).

$$\begin{aligned} \alpha_{water} &= 1 \;\; for\ water\ phase \\ \alpha_{water} &= 0 \;\; for\ air\ phase \end{aligned} \tag{7.1}$$

In OpenFOAM, the solver for α_{water} is defined in *fvSolution* under *system* folder. The coefficients and solvers are defined in *solvers* sub-dictionary. In this study, all fixed and free attitude simulations use *smoothSolver* as the solver and *symGaussSeidel* as the smoother.*smoothSolver* means that a solver that uses a smoother and *symGaussSeidel* means symmertic Gauss-Seidel with the number of sweeps set to 1 by default. The tolerance is set to $1e^{-10}$. There are two

important coefficients defined in *solver*, one is *cAlpha* and the other is *nAlphaSubCycles*. The two simulations performed to improve the results are based on changing the *cAlpha* and *nAlphaSubCycles*.

cAlpha is defined as an extra term in the equation of phase fraction convention. The extra term known as Artificial Compression physically introduce a pressure on the free surface to keep it from dispersing, thus the transport equation with the Artificial Compression is (7.2) where u_r is the velocity field normal to the free surface [1].

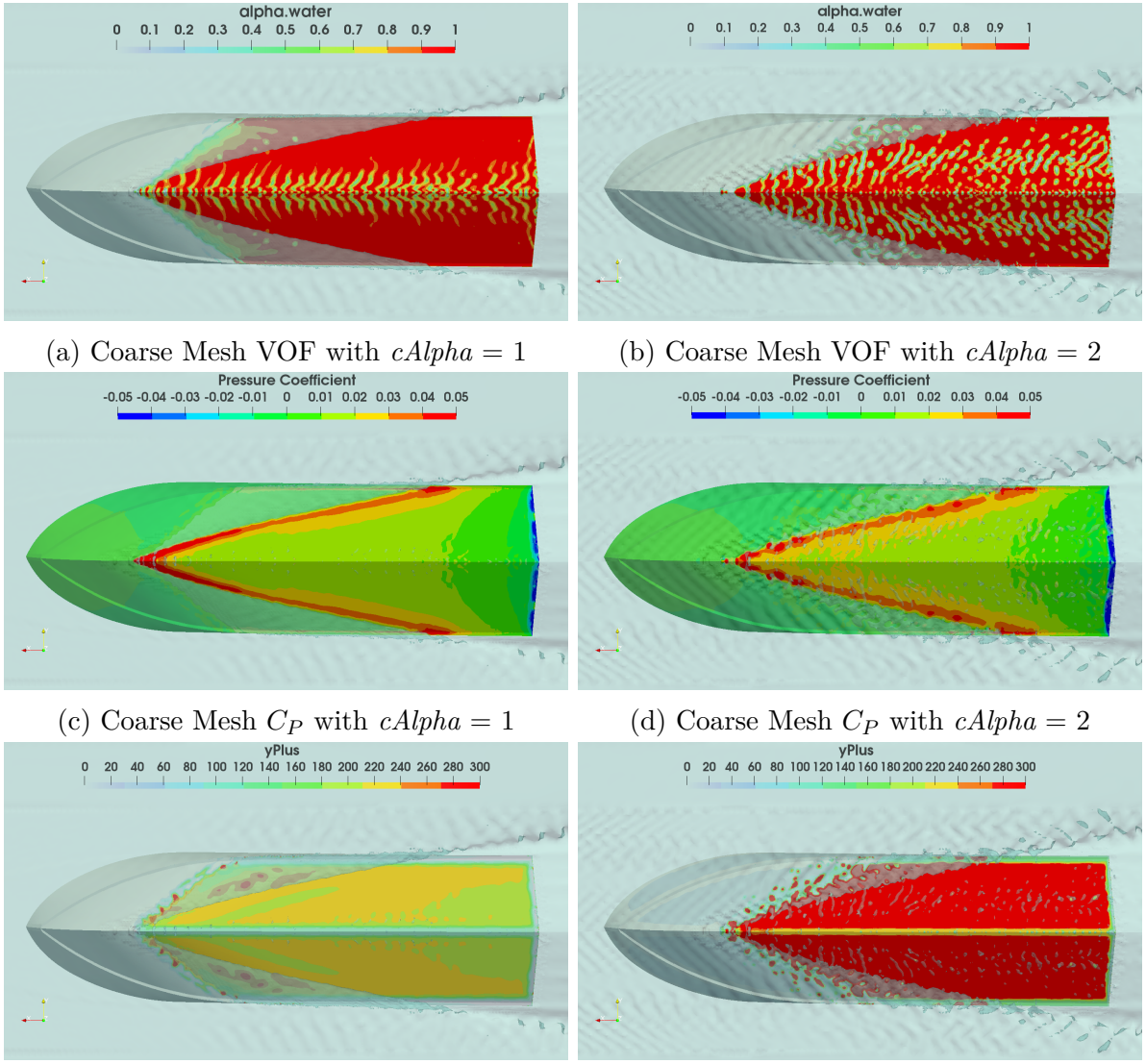
$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \big(\alpha u\big) + \nabla \cdot \big(\alpha\big(1-\alpha\big)u_r\big) \tag{7.2}$$

In OpenFOAM, *cAlpha* is defined as below [14]:

$$cAlpha = 0 : No\ Artificial\ Compression$$
$$cAlpha = 1 : Conservative\ Artificial\ Compression$$
$$cAlpha > 1 : Enhanced\ Artificial\ Compression$$

We still use the coarse mesh case for the purpose of fast computing. The original case has $cAlpha = 1$ indicating a conservative Artificial Compression, and the modified case has $cAlpha = 2$ indicating an enhanced compression on the interface. All the other settings are the same for these two cases. Again, the hull bottom plots for VOF, pressure coefficients and $y+$ are generated for the $cAlpha = 2$ case and compared with the original results shown in Figure 7.5.

From Figure 7.3, 7.3a and 7.3b, the numerical ventilation at the hull bottom is greatly reduced due to the increase of *cAlpha*. Numerical ventilation aggregating at keel decreases while large ventilation area appears between the chine and the keel. In our simulation,



(a) Coarse Mesh VOF with $cAlpha = 1$

(b) Coarse Mesh VOF with $cAlpha = 2$

(c) Coarse Mesh C_P with $cAlpha = 1$

(d) Coarse Mesh C_P with $cAlpha = 2$

(e) Coarse Mesh $y+$ with $cAlpha = 1$

(f) Coarse Mesh $y+$ with $cAlpha = 2$

Figure 7.3: Comparison of Numerical Solution Between Original Coarse Mesh with $cAlpha$=1 and the Modified Coarse Mesh with $cAlpha$=2. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom. Steady-state Solution at Highest Speed $\mathrm{Fn}_\nabla = 5.72$

adding the enhanced compression doesn't improves the precision of solving bottom VOF, on the contrary, the results are even worse than the original simulation. Another main difference appears at the calculation of $y+$, the non-dimensional wall distance $y+$ of the modified case is much larger than the original case. So, increasing *cAlpha* could enlarge $y+$ in our simulations.

Table 7.2: Results Comparison Between Coarse Mesh with Different *cAlpha* Obtained at $\mathrm{Fn}_\nabla = 5.72$

	cAlpha 1	Relative Error (%)	*cAlpha* 2	Relative Error (%)	Experimental Result
Resistance [N]	220.4	18.32	231.0	14.39	269.83
Friction Drag [N]	158.0	—	169.3	—	—
Pressure [N]	62.4	—	61.7	—	—
Trim Angle [degree]	1.87	34.15	1.95	10.14	2.17
Heave [cm]	10.32	7.54	10.26	6.88	9.60
L_C [cm]	43.45	2.25	44.27	0.40	44.45
L_K [cm]	168.94	2.19	169.05	2.12	172.72

As described in Table 7.2, it's surprisingly to find that even the increase of *cAlpha* does no improvements on the solution of hull bottom VOF, the total resistance still improves 3.93% of experimental total resistance. Also the friction drag increases with the increment of numerical ventilation at the bottom, which is hard to explain physically. This might be due to the increase of numerical errors and the co-effect from the other coefficients. Time history plot 7.4 shows that the $cAlpha = 2$ case oscillates a lot and finally gained oscillatory convergence. The friction drag oscillates a lot while pressure resistance is stabler, this might be resulted from the large ventilation at the bottom. The computation time for $cAlpha = 2$ case is 226.38 hours which is 18 times longer than the original coarse mesh simulation. Further study should be conducted to understand effect of *cAlpha* to our simulations, like $cAlpha < 1$, $cAlpha > 1$ with medium mesh, etc.

We still have one coefficient *nAlphaSubCycles* to discuss. This coefficient represents the

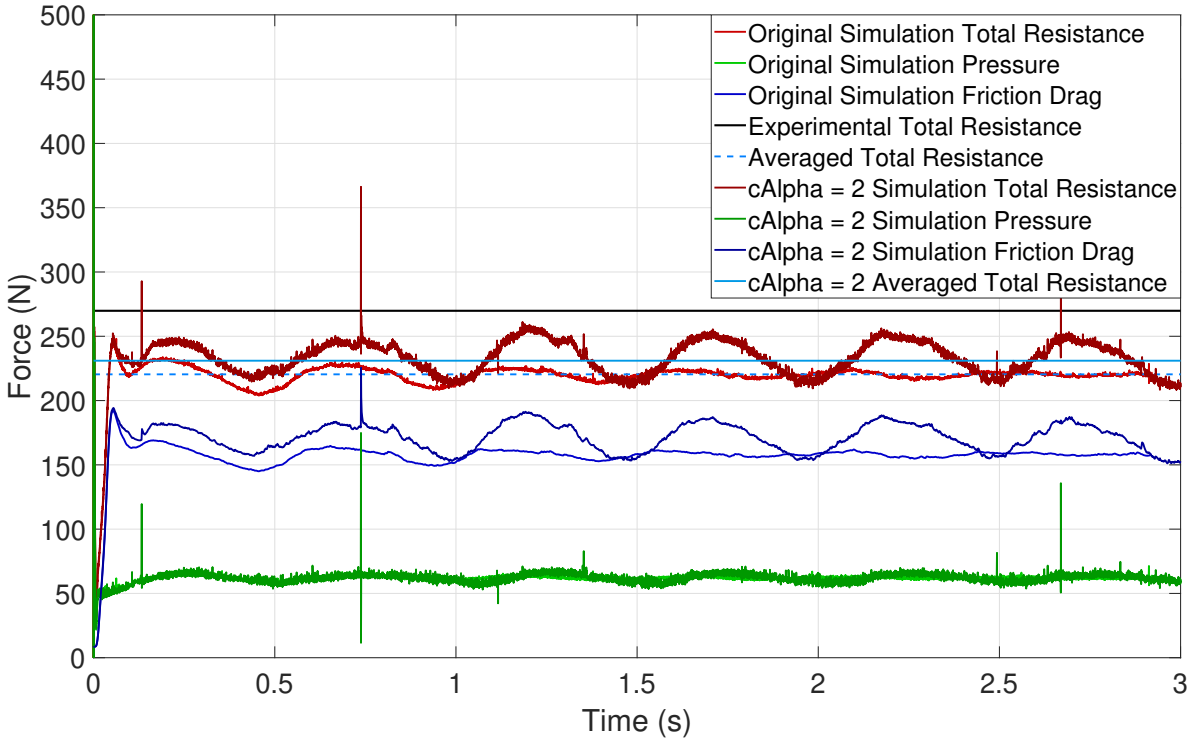
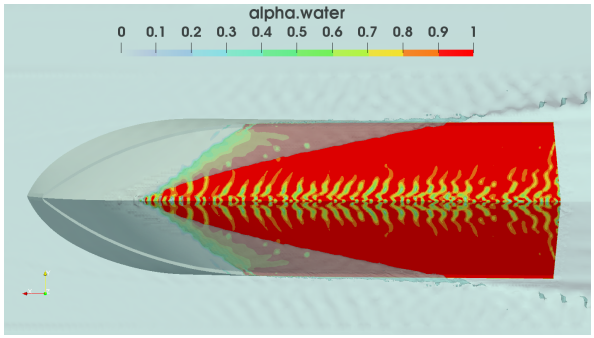


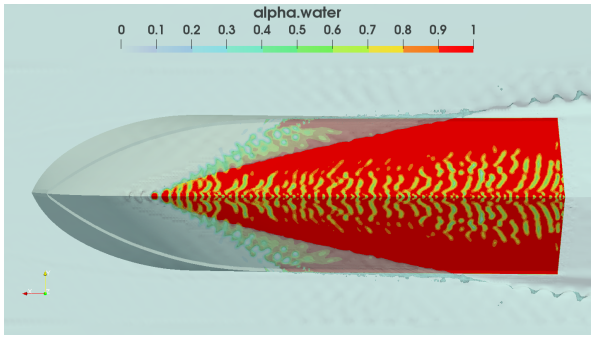
Figure 7.4: Time History of Different *cAlpha* Cases Resistance

number of sub-cycles which are additional solutions to α equation within a given time step. Adding the sub-cycles would stabilize the solution without reducing the time step and greatly increase the execution time. In a new modified case, the *nAlphaSubCycles* is changed from the original 1 to 2, meaning that breaking one true time step into two half-length time steps and solve for α equation in these two half time steps [14]. The execution time is 74 hours and it's about 6 times as long as the original simulation. The results are summarized in Table 7.3 and compared in Figure 7.5.

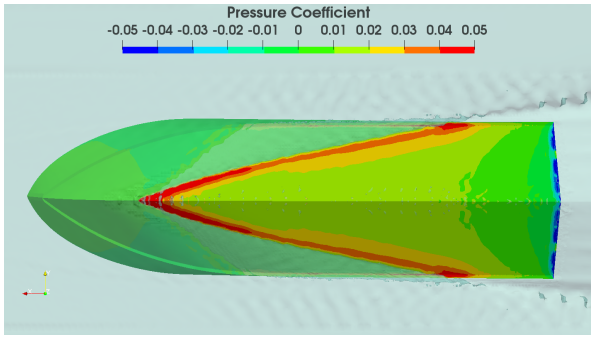
By comparing the VOF of the hull bottom, increasing *nAlphaSubCycles* from 1 to 2 also induces more numerical ventilation at the bottom as shown in 7.5a and 7.5b. In addition to it, the $y+$ is also larger than the original simulation. The results shown in Table 7.3 show that the modified case failed to yield improved solution. The oscillations of the modified case shown in plot 7.6 are close to the original case, but the results of total resistance and



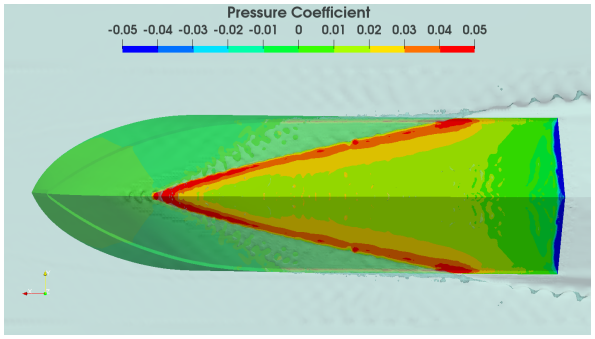
(a) Coarse Mesh VOF with *nAlphaSubCycles* = 1



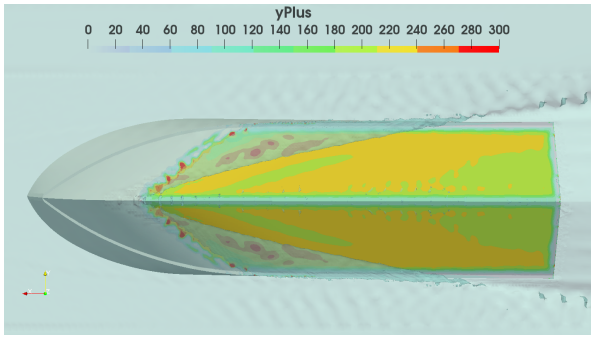
(b) Coarse Mesh VOF with *nAlphaSubCycles* = 2



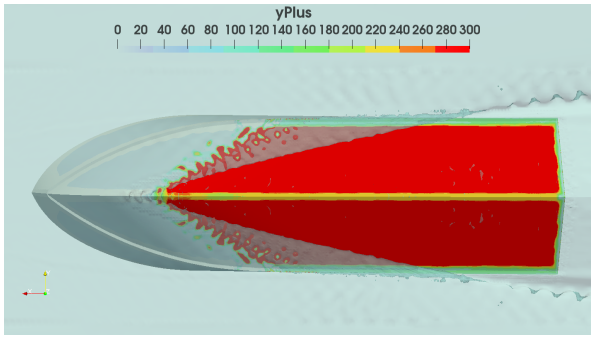
(c) Coarse Mesh C_P with *nAlphaSubCycles* = 1



(d) Coarse Mesh C_P with *nAlphaSubCycles* = 2



(e) Coarse Mesh $y+$ with *nAlphaSubCycles* = 1



(f) Coarse Mesh $y+$ with *nAlphaSubCycles* = 2

Figure 7.5: Comparison of Numerical Solution Between Original Coarse Mesh with *nAlphaSubCycles*=1 and the Modified Coarse Mesh with *nAlphaSubCycles*=2. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom. Steady-state Solution at Highest Speed $\mathrm{Fn}_\nabla = 5.72$

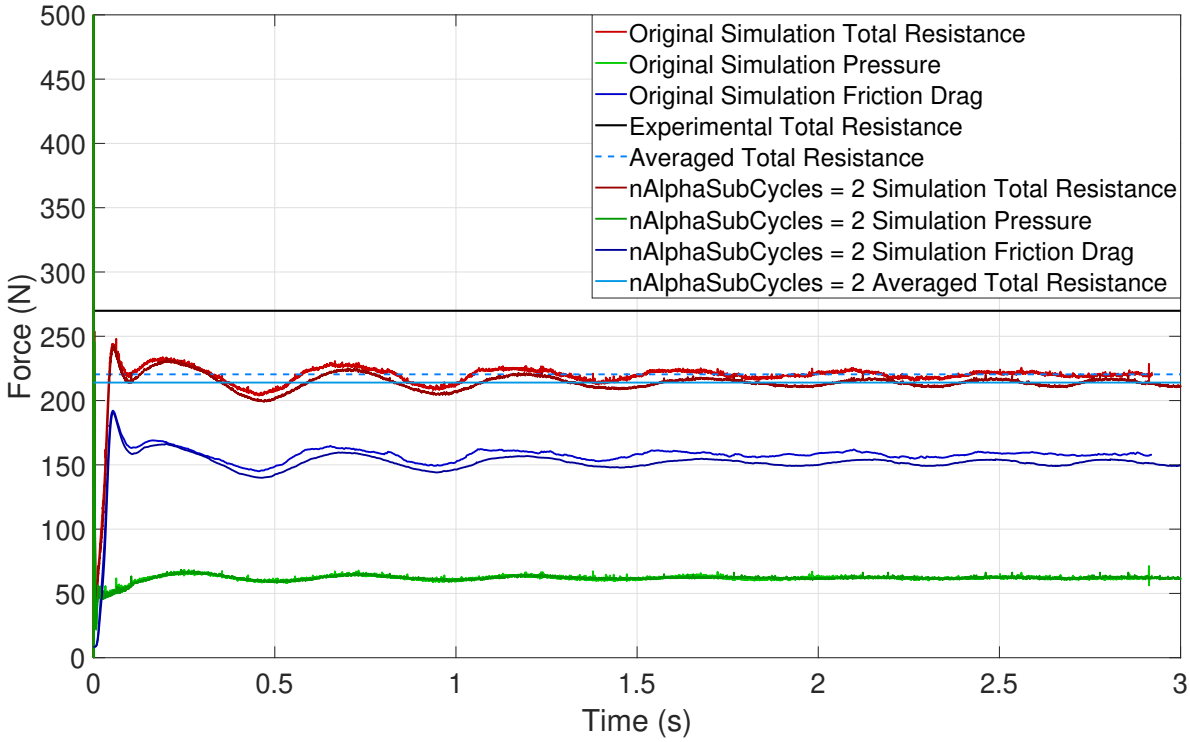


Figure 7.6: Time History of Different *cAlpha* Cases Resistance

friction drag are smaller than the original case.

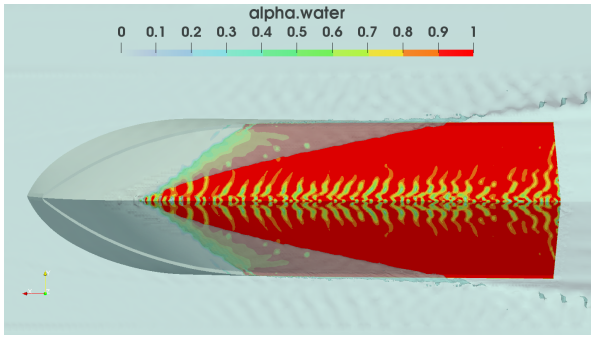
Table 7.3: Results Comparison Between Coarse Mesh with Different *nAlphaSubCycles* Obtained at $\mathrm{Fn}_\nabla = 5.72$

	nAlphaSubCycles 1	Relative Error (%)	*nAlphaSubCycles* 2	Relative Error (%)	Experimental Result
Resistance [N]	220.4	18.32	214.0	20.69	269.83
Friction Drag [N]	158.0	—	151.8	—	—
Pressure [N]	62.4	—	62.2	—	—
Trim Angle [degree]	1.87	34.15	1.85	14.75	2.17
Heave [cm]	10.32	7.54	10.32	7.50	9.60
L_C [cm]	43.45	2.25	41.92	5.96	44.45
L_K [cm]	168.94	2.19	167.62	1.75	172.72

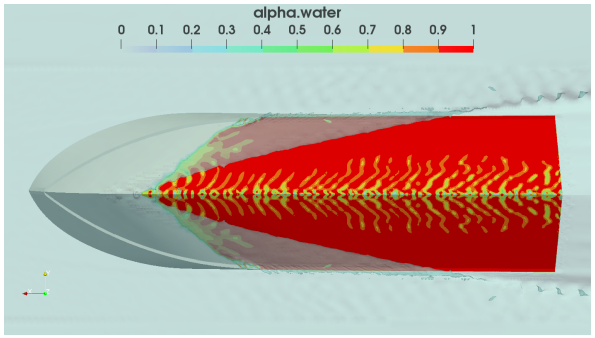
7.3 Change Hull Surface Roughness

It is noted that all the results of our simulations are smaller than the experimental results, a guess arouse that the roughness setting of the hull surface might be small which would result in a small viscous force. In OpenFOAM 0 folder, the initial conditions are defined and *nutkWallFunction* is used for the hull in *nut*. Wall function *nutkWallFunction* provides a turbulence viscosity condition based on the turbulence kinetic energy[13]. In order to get a rough hull surface, the wall function is changed to *nutkRoughWallFunction*. User can define two parameters, one is K_s that defines the roughness height and setting to 0 means a smooth wall, the other is C_s which is the roughness constant usually varies from 0.5 to 1. To see how the roughness will change the simulation results, we use K_s equals 0.0001(m) and C_s is 0.5. The results are plotted and compared with the original case with smooth hull surface in Figure 7.7 and Table 7.4. The time history is also plotted in Figure 7.8, and it's very obvious that the friction drag and total resistance are very large due to the unreasonable surface roughness.

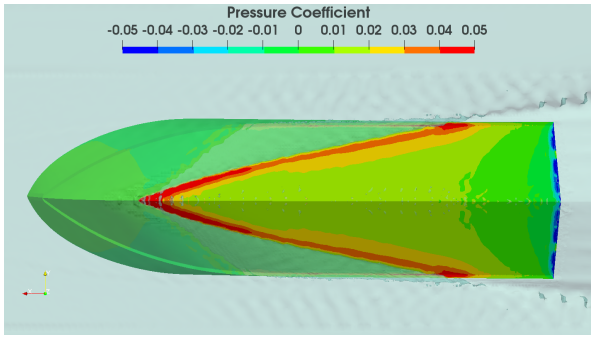
By changing the *nut* wall function from *nutkWallFunction* to *nutkWallFunction*, the ventilation at hull bottom decreased a little as shown in 7.7a and 7.7b. However, the non-dimensional wall distance $y+$ increased a lot as shown in 7.7e and 7.7f. Let's also take a look at Table 7.4 where the results are summarized, the friction drag increases too much and results in the big increment in the total resistance. The pressure is very close to the smooth hull surface simulation. Except the unwanted solution for friction drag and total resistance. The other results are all improved, especially the trim angle. Making the surface a little bit rougher might improve the results under the circumstance that the K_s and C_s is set to the correct value, not too rough like the default value in our simulation. However, it's very hard to know how rough the surface is of NSWCDD test model, hence the correct values of K_s and



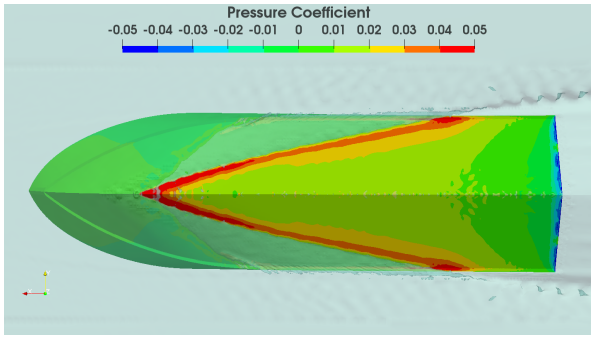
(a) Original Coarse Mesh VOF with *nutkWallFunction*



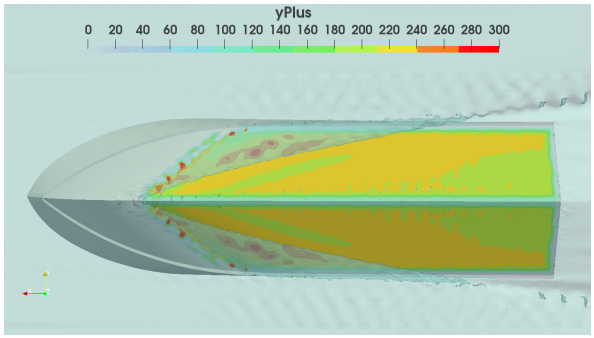
(b) Coarse Mesh VOF with *nutkRoughWallFunction*



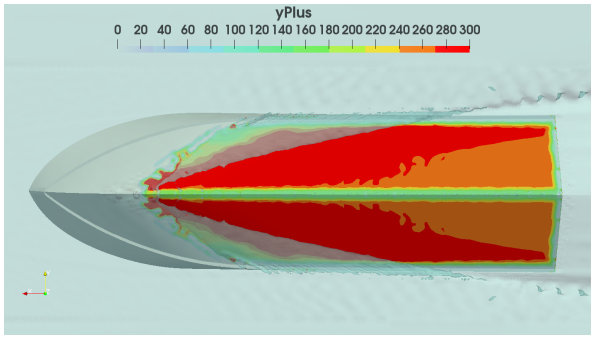
(c) Original Coarse Mesh C_P with *nutkWallFunction*



(d) Coarse Mesh C_P with *nutkRoughWallFunction*



(e) Original Coarse Mesh $y+$ with *nutkWallFunction*



(f) Coarse Mesh $y+$ with *nutkRoughWallFunction*

Figure 7.7: Comparison of Numerical Solution Between Original Coarse Mesh with *nutkWallFunction* and *nutkRoughWallFunction*. Results Presented in Terms of Volume of Fluid, Pressure Coefficient and Non-dimensional Wall Distance ($y+$) Contours on the Hull Bottom. Steady-state Solution at Highest Speed $\mathrm{Fn}_\nabla = 5.72$

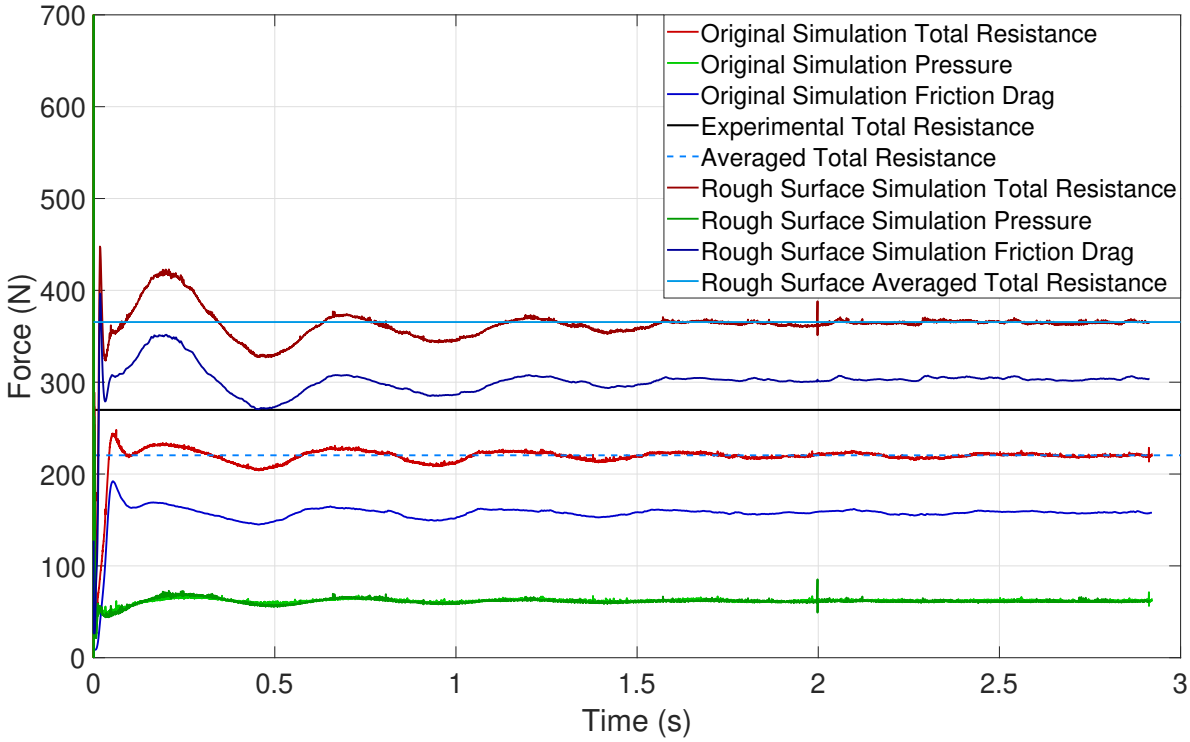


Figure 7.8: Time History of *nutkWallFunction* and *nutkRoughWallFunction* Cases Resistance

C_s are unknown. In conclusion, changing the *nut* wall function to *nutkRoughWallFunction* could be a way to improve the simulation results providing the correct roughness of the test model. But this method is not suggested to use for further study as a result of unknown roughness.

Table 7.4: Results Comparison Between Coarse Mesh with *nutkWallFunction* and *nutkRough-WallFunction* Obtained at $\mathrm{Fn}_\nabla = 5.72$

	nutkWallFunction	Relative Error (%)	*nutkRoughWallFunction*	Relative Error (%)	Experimental Result
Resistance [N]	220.4	18.32	365.5	35.46	269.83
Friction Drag [N]	158.0	—	303.5	—	—
Pressure [N]	62.4	—	62.0	—	—
Trim Angle [degree]	1.87	34.15	2.05	5.38	2.17
Heave [cm]	10.32	7.54	10.10	5.21	9.60
L_C [cm]	43.45	2.25	44.58	0.29	44.45
L_K [cm]	168.94	2.19	170.61	1.22	172.72

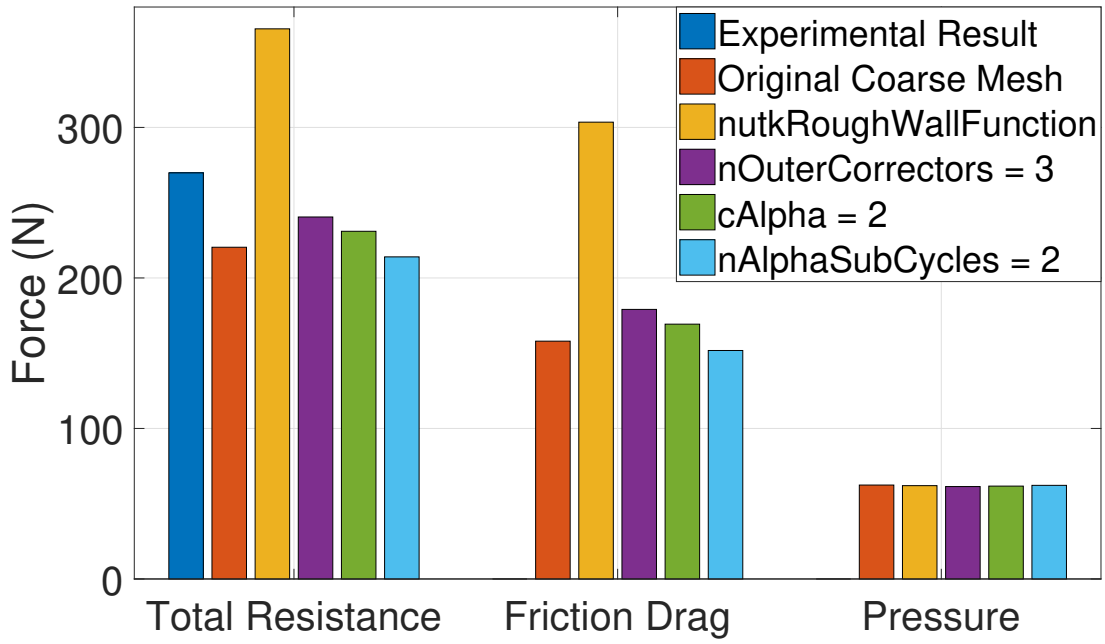


Figure 7.9: Resistance Comparison of Four Modified Simulations with the Original Coarse Mesh Simulation and the Experimental Result

7.4 Discussion and Future Development

Summarize the four different modified simulations described in Section 7.1, 7.2 and 7.3, a bar plot 7.9 is generated to compare the different resistance components of these modified cases with the original coarse mesh simulation and the experimental results.

The bar plot shows that the four modified simulations and the original coarse mesh simulation have a little difference in calculation of pressure. The pressure variation between the five simulations are within 1 N, which is very small and negligible comparing to the change of friction drag. The main difference is the solution for friction drag which results in the large variation in the total resistance consequently. Since the *nutkRoughWallFunction* case is physically not acceptable, the other three cases show how the modifications of different coefficients would effect the friction drag solution. Among the three cases, the case

with$nOuterCorrectors$=3 and case with $cAlpha = 2$ increase the friction drag while the case with $nAlphaSubCycles = 2$ provides decreased friction drag solution. In further study to get more accurate resistance results, increasing $nOuterCorrectors$ and $cAlpha$ could be an idea. But for the purpose of finding the best combinations of these coefficients for the solver as well as acceptable computational cost, more simulations should be conducted and tested using different settings.

Chapter 8

Conclusion

The Generic Prismatic Planing Hull (GPPH) tested at the NSWCCD provides a valuable opportunity to test numerical models and tune their parameters against a modern set of experiments. In this thesis, the meshing strategy was introduced and the mesh convergence study, as well as verification and validation study, were performed for the fixed attitude simulations and free attitude simulations at the highest speed. Two grids were chosen to perform a serious of free attitude numerical simulations of GPPH model at 7 tested speeds from pre-planing to fully planning regime using open source software OpenFOAM. Results of the simulations have shown agreement with experiment benchmark data and have been compared with both experimental data and Savitsky Short Method. This thesis also discusses the simulation improvements by performing free attitude simulations of modified cases for the reference of future work.

Firstly, the GPPH model has been rotated and translated to experimental steady state, four different grids were generated for different resolutions with OpenFOAM *snappyHexMesh* mesh generation program. The coarse mesh (1 million cells) and medium mesh (2 million cells) used 6 *topoSet* boxes for refinement while the fine mesh(4 million cells) and finest mesh (8 million cells) had extra refinement box around the hull bottom and free surface area except for the 6 *topoSet* boxes.

Using the four generated mesh, fixed attitude simulations were performed at the highest speed of 12.24 m/s with minimum relative error 1.79% w.r.t. experiment. The mesh conver-

gence study and verification and validation (V&V) study for medium, fine and finest mesh have yield monotonic convergence with 6.60% verification uncertainty and 6.31% validation uncertainty w.r.t finest grid solution. However, mesh convergence study for coarse, medium and fine mesh showed divergence. Considering the computational cost, coarse mesh is still kept to be applied in free attitude simulations.

Then, free attitude simulations with three meshes were performed at the highest speed and mesh convergence and V&V study were performed based on the results. The V&V study obtained 7.90% simulation numerical uncertainty and 7.39% validation uncertainty w.r.t fine grid solution. For the purpose of cost-effective simulation, the coarse mesh and medium mesh were finally used in the free attitude simulations at all seven tested speeds. The free body dynamic mesh simulation results were compared with the experimental benchmark data and in terms of total resistance, pressure, friction drag, trim angle, heave, wetted keel length and wetted chine length. Same speed hull bottom Volume of Fluid (VOF), pressure coefficient and non-dimensional wall distance were plotted and compared. The medium mesh simulations in the whole speed range yield 13.58% maximum error, 4.6% minimum error and 7.91% average error in total resistance solution. The medium mesh simulations also provide 5.74% maximum error, 2.58% minimum error and 3.93% average error in wetted chine length. For the solution of wetted keel length, a maximum error of 1.49%, a minimum error of 0.28% and an average error of 0.72% are attained. These errors might be caused by the insufficient grid resolution, inappropriate algorithms and the usage of wall functions. In general, the simulation results have been improved by increasing the mesh resolution from coarse mesh to medium mesh. The problem of numerical ventilation was partially mitigated by increasing the grid resolution while keeping relatively high values of $y+$.

Finally, in order to improve the simulations, four modified cases were simulated with *nutkRough-WallFunction* (was *nutkWallFunction*), *nOuterCorrectors* = 3 (was 1), *cAlpha* = 2 (was 1)

and $nAlphaSubCycles = 2$ (was 1). The change in $nOuterCorrectors$ improved the total resistance by 3.93% w.r.t experimental data and the other changes had minor improvements on the results which might give reference to future work. Also, further studies are needed to understand the influence of other wall functions on VOF diffusion on the pressure area of planing hulls and to characterize the importance of numerical solution of the partial differential equation describing the transport of VOF. Different mesh strategies are also needed to be tested to improve the results.

Bibliography

[1] Mostafa Amini Afshar. *Numerical Wave Generation In OpenFOAM®*. Chalmers tekniska högskola, 2010. Master of Science Thesis.

[2] R. Broglia and D. Durante. Accurate prediction of complex free surface flow around a high speed craft using a single-phase level set method. *Computational Mechanics*, 62: 421–437, September 2018. doi: 10.1007/s00466-017-1505-1.

[3] R. Featherstone. *Rigid body dynamics algorithms*. Springer, 2014.

[4] Gaggero S. Ferrando M. and Villa D. Open source computations of planing hull resistance. *Transactions of the Royal Institution of Naval Architects Part B: International Journal of Small Craft Technology*, 157:83–98, 2015.

[5] Stefano Gaggero, Diego Villa, and Marco Ferrando. An open source approach for the prediction of planing hull resistance. 10 2014.

[6] 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.

[7] R. Issa. Solution of the implicitly discretized fluid flow equations by operator splitting. *Journal of Computational Physics*, 62(1):40–65, 1985.

[8] H. Jasak. Dynamic Mesh Handling in OpenFOAM. In *47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, 5-8 January, Orlando, Florida*, 2009.

[9] H. Jasak and Z. Tukovic. Automatic Mesh Motion for the Unstructured Finite Volume Method. *Transactions of FAMENA*, 30(2):1–20, 2007.

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

[11] Lars Larsson, Frederick Stern, and Michel Visonneau. *Numerical ship hydrodynamics: an assessment of the Gothenburg 2010 workshop*. Springer, 2013.

[12] Weil C.R. Lee E. and Fullerton A. Experimental results for the calm water resistance of the generic prismatic planing hull (gpph). Report NSWCCD-80-TR-2017/015, Naval Surface Warfare Center Carderock Division, 2017.

[13] F Liu. A thorough description of how wall functions are implemented in openfoam. *Proceedings of CFD with OpenSource Software*, pages 1–33, 2016.

[14] 2015-2019 CFD Direct Ltd. Openfoam v6 user guide: 2.3 breaking of a dam. https://cfd.direct/openfoam/user-guide/v6-dambreak/, 2018.

[15] Agostino De Marco, Simone Mancini, Salvatore Miranda, Raffaele Scognamiglio, and Luigi Vitiello. Experimental and numerical hydrodynamic analysis of a stepped planing hull. *Applied Ocean Research*, 64:135 – 154, 2017. ISSN 0141-1187. doi: https://doi.org/10.1016/j.apor.2017.02.004. URL http://www.sciencedirect.com/science/article/pii/S0141118716302620.

[16] F.R. Menter. Zonal two equation k-ω Turbulence Models for Aerodynamic Flows. In *24th AIAA Fluid Dynamics Conference, July 6-9, Orlando, Florida*, page 2906, 1993.

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

[18] OpenFOAM Foundation. *OpenFOAM user guide*. OpenFOAM Foundation, 2014.

[19] Brizzolara S. and Federici A. Cfd modeling of planning hulls with partially ventilated bottom. In *The William Froude Conference: Advances in Theoretical and Applied*

Hydrodynamics – Past and Future, pages 1–17, London, 2010. Royal Institution of Naval Architects.

[20] Brizzolara S. and Villa D. Cfd simulations of planing hulls. In *Proceedings of the Seventh International Conference on High-Performance Marine Vehicles, HIPER 2010*, pages 16–24, Melbourne, 2010. Florida Institute of Technology.

[21] Brizzolara S. and Serra F. Accuracy of cfd codes in the prediction of planning surfaces hydrodynamic characteristics. In *Proceedings of the 2nd International Conference on Marine Research and Transportation, ICMRT'07*, pages A:1–12, Italy, 2007. U. of Napoli Federico II.

[22] Daniel Savitsky. Hydrodynamic design of planing hulls. *Marine technology*, 1(1), 1964.

[23] Fred Stern, Robert V Wilson, Hugh W Coleman, and Eric G Paterson. Verification and validation of cfd simulations. Technical report, IOWA INST OF HYDRAULIC RESEARCH IOWA CITY, 1999.

[24] D.C. Wilcox. Reassessment of the Scale-Determining Equation for Advanced Turbulence Models. *The American Institute of Aeronautics and Astronautics (AIAA)*, 26:1299–1310, 1988.