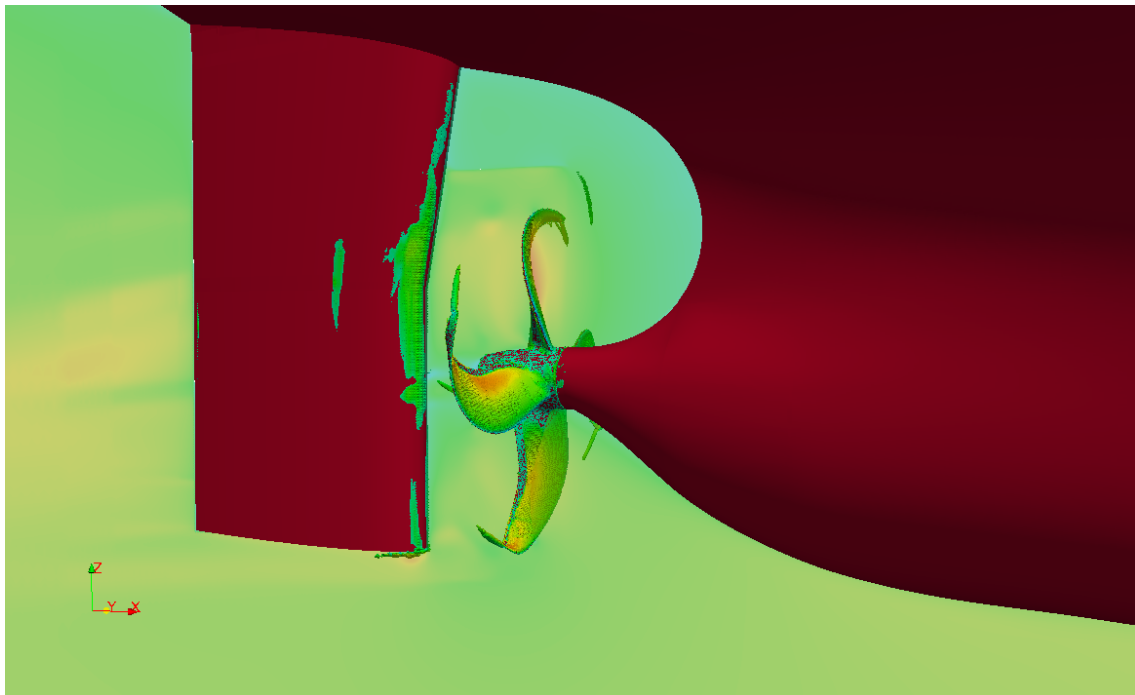# CHALMERS

Investigation of Effective Wake Scaling for Unconventional Propellers

Master's Thesis in Hydrodynamics

SURYA KIRAN PERAVALI

Department of Shipping & Marine Technology
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2015
Master's Thesis 2015:X-15/318

A THESIS FOR THE DEGREE OF MASTER OF SCIENCE

Investigation of Effective Wake Scaling for Unconventional Propellers

A comparative study on ITTC 78 scaling method using Open Source CFD

SURYA KIRAN PERAVALI

Department of Shipping and Marine Technology
Division of Hydrodynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2015

Investigation of Effective Wake Scaling for Unconventional Propellers
Using Open Source CFD
SURYA KIRAN PERAVALI

Cover: Flow structures (Iso Q) and axial velocity variation for a full scale unconventional
propeller behind a hull

Investigation of Effective Wake Scaling of Unconventional Propellers
Using Open Source CFD
SURYA KIRAN PERAVALI
Department of Shipping and Marine Technology
Chalmers University of Technology

Abstract

At SSPA model tests are evaluated and scaled to full scale using the ITTC 78 performace prediction method. The ITTC 78 method has been validated against the sea trials over more than 35 years and is very adapted to the conventional propellers. However, new type of propellers such as Kappel propeller seem to challenge the traditional method, in particular with wake scaling. Suspicions have been raised that the standard methods for evaluation of the model tests might give misleading results for Kappel propeller [1]. This report presents a comparative study in various aspects of the performance of a conventional propeller with an unconventional propeller (Kappel propeller). The study is carried out using OpenFOAM RANS solver for computing the viscous flow around a ship hull, propellers in open-water and propellers behind the hull. The numerical results are validated by comparing with the model scale experiments performed at SSPA. Further the computations are extended to full scale and the sources of scaling effects are identified. The differences in scaling between the conventional and unconventional propellers are examined and further this relative differences are compared with that of ITTC 78 scaling method.

The correlation of CFD results with the experimental study justify the accuracy of the CFD setup in predicting the complex flow between hull and propeller. In full scale the CFD results show a larger Reynolds scaling effect on the unconventional propeller compared with the conventional propeller in open-water condition. The scaling effects on various full scale performance factors such as the effective wake fraction, thrust deduction, hull and total efficiencies are evaluated for the propellers behind the hull using CFD and compared with the full scale estimations from ITTC 78 method. At the design condition the CFD study estimates the unconventional propeller has much lower effective wake and hull efficiency. However the propeller and total efficiencies are higher for unconventional propeller. The CFD estimations in full scale contradict the estimations from the ITTC 78 method. In this current report the differences between CFD (RANS) and ITTC 78 method are presented and the areas of further research are identified in order to uncover the other discrepancies.

Keywords: Unconventional/Kappel propeller, scale effect, ITTC-78 method, wake fraction, OpenFOAM

Acknowledgements

Contents

List of Figures

List of Tables

NOMENCLATURE

Abbrevation

AMI	Arbitrary Mesh Interface
BC	Boundary Condition
CFD	Computaional Fluid Dynamics
CFL	Courant-Friedrich-Lewy number
DILU	Diagonal incomplete-LU
DNS	Direct Numerical simulation
FVM	Finite Volume Method
GAMG	Generalized Geometric-Algebraic Multigrid
GUI	Graphical User Interface
ITTC	International Towing Tank Conference
LES	Large Eddy Simulation
MRF	Multiple Reference Frame
NVD	Normalised Variation Diminishing
OpenFOAM	Open source Field Operation And Manipulation
PCG	Preconditioned conjugate gradient
PBiCG	Preconditioned biconjugate gradient
PISO	Pressure-Implicit Split-Operator
RANS	Reynolds Averaged Navier-Stokes
SIMPLE	Semi-Implicit Method for Pressure Linked Equations
SST	Menter shear stress transport
VTK	Visualization Tool Kit
TVD	Total Variation Diminishing

Greek Symbols

δ	Kronecker's delta
ρ	Density
μ	Dynamic viscosity
ν	Kinematic viscosity
∂	Partial derivative
ω	Specific turbulent dissipation
τ_{ij}	Viscous stress tensor
ϵ	Turbulent dissipation
σ_{ij}	Total stress tensor
Ω	vorticity tensor

Roman Symbols

C_F	Frictional resistance coefficient
C_P	Pressure coefficient
C_T	Total resistance coefficient
D	Propeller diameter
ETA_0	Propeller efficiency
ETA_D	Propulsive efficiency/ Total efficiency
ETA_h	Hull efficiency
f_i	Volume forces
F_r	Froude number
J	Advance coefficient
k	Turbulent kinetic energy
k_p	Full scale blade roughness
K_Q	Torque coefficient
K_T	Thrust coefficient
n	Propeller revolution rate
p	Pressure
P_D	Delivered power
P_E	effective power
R_e	Reynold's number
S	wetted surface area of the hull
S_{ij}	Strain rate tensor
t	Time
t	Thrust deduction
u_i	Time averaged velocity vector
U_i	Instataneous velocity vector
W_T	Taylor wake fraction
y+	Non-dimenstional wall distance
x	Space variable

Subscripts

∞	Free stream property
i,j,k	tensor indices
t	turbulence
m	Model scale
s	Full scale

Chapter 1

Introduction

1.1 Background

The effort to find the solutions to the environmental and energy saving problems regarding the operation of ships is always a matter of concern. Such problems arise due to low propulsion efficiency and due to propeller induced noise and vibrations. Therefore there is a demand for the efficiency improving devices for a specific propeller in its operating profile. One of the sources for these induced noise and vibration is due to the tip vortices which are formed by the pressure differences between pressure side and suction side of the blade profile. This results in a vortex giving raise to unsteady pressure field at the tips thereby decreasing the efficiency of the tip region, inducing the vibrations and noise to the propeller. The propellers with non-planar lifting surfaces such as Kappel propeller have the designed approach to minimize the flow over the tip and outer region of the propeller. The aim is to retain a high efficiency increasing the total efficiency of the propeller and lowering the propeller induced noise and vibrations compared to conventional propellers [3].

For full scale performance evaluation for these unconventional propellers on the basis of model tests, the ITTC 1978, method which is a universal tool for evaluating conventional propellers has been normally used [1]. The predicted hull efficiency in full scale is claimed to be lower than the model scale due to the reduction in the frictional wake. This discrepancy is may be due to the methods applied for scaling the wake and open water tests. However the ITTC 78 method works well for the conventional propellers. Therefore it might be necessary to revise the ITTC 78 method for scaling of open water efficiency and wake fraction in order to account for all the discrepancies and to have a performance prediction method which can be applied to all types of propellers.

Figure 1.1: The Kappel propeller

1.2 Objective

The objective of this study is to perform CFD computations in open-water and behind conditions, at model and full scale, for two types of propellers: the conventional and the unconventional (kappel type). The propulsive quantities (in particularly wake) and their evolutions from model to full scale, are compared to what is predicted by the model tests and scaling using the ITTC 78 method for both the propellers.

1.3 Method

As mentioned the objective is to analyse two different propellers working behind a same hull using CFD. The detailed list of tasks are as follows:

1. Problem definition and case setup in OpenFoam.

2. Open-source mesh development for the test cases.

3. Resistance simulation of the bare Hull in model scale.

4. Open water simulations of conventional propeller in model scale.

5. Open water simulations of unconventional propeller in model scale.

6. Self-propulsion simulation with conventional propeller in model scale.

7. Self-propulsion simulation with unconventional propeller in model scale.

8. Mesh dependency tests for the models above.

9. Full scale open water simulation for conventional propeller.

10. Full scale open water simulation for unconventional propeller.

11. Full scale self-propulsion simulation with conventional propeller.

12. Full scale self-propulsion simulation with unconventional propeller.

The data from the model scale simulations are compared with the experimental model tests in order to check the accuracy of the simulation. The ITTC-78 method is used to evaluate the model scale tests and to scale the full scale test cases. The full scale wake (W_{Ts}) is calculated from the model scale wake (W_{Tm}) and thrust deduction factor (t) according to:

$$W_{Ts} = (t + 0.04) + (W_{Tm} - t - 0.04)\frac{(1+k)C_{Fs} + \Delta C_F}{(1+k)C_{Fm}} \tag{1.1}$$

This formula was developed in the 70's keeping conventional propellers in mind. The outcome from this method is compared with the results of full scale simulations in order to study how the scaling effect the flow details and its reflection in the ITTC wake scaling formula.

1.4 Limitations and Assumptions

The time frame for this project is around 9 months and it is performed at SSPA with the computational resources at SSPA. The focus is kept on the CFD simulations in model scale and full scale test cases. The RANS methods (k-ω SST) have been applied for predicting the performance and flow patterns around the propellers and the hull. Two incompressible solvers in OpenFOAM are used for the computations. `simpleFOAM` solver is used for steady state simulations and `pimpleDyMFoam` solver is used for unsteady computations for test cases with dynamic sliding mesh interfaces. The free-surface and trim of the hull can play an important role in evaluation of performance. Since, a comparative study is performed between the performance of two propellers, the effects of trim and free surface should reduce. Further, the Froude number and trim of the test cases is very low. Therefore, the effects of trim and the free surface are not considered for this study.

1.5 Composition of this thesis

This thesis comprises of three major parts: numerical set up of the test cases in OpenFoam and dicretize them using open-source meshing tools: Numerical simulation of test cases in model and full scale and Validating the simulation by comparing them with model scale experiments and further comparing the full scale numerical results with the results from ITTC 78 performance prediction method. Figure 3.7 shows the entire process flow of the work performed in the thesis.

This report contains six chapters. The present chapter deals with the background and objectives in the project and also gives a brief outline about the approach of the work along with the limitations and the assumption in the work. Chapter 2 contains a summary about the numerical methods applied and their implementations along with few additional considerations in OpenFOAM. Chapter 3 gives a overview about the open-source meshing and the simulation setup in OpenFoam. In this chapter the dicretized mesh is described for different cases and the grid study. The various solver settings, post processing and the entire work flow of the project is described in this chapter. In chapter 4 the results are presented and discussed. Chapter 5 show the summary of the work performed and various conclusions in the work. The future scope of the work is presented in chapter 6.

Chapter 2

Theory

2.1 Governing Equations

The fluid is presumed to be incompressible and Newtonian. For this incompressible hydrodynamic flow studies, the Navier-Stokes equations in cartesian coordinate system are written as:

$$\rho\Big(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}\Big) = -\frac{\partial p}{\partial x} + \mu\Big(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\Big) \tag{2.1}$$

$$\rho\Big(\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z}\Big) = -\frac{\partial p}{\partial x} + \mu\Big(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\Big) \tag{2.2}$$

$$\rho\Big(\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z}\Big) = -\frac{\partial p}{\partial x} - \rho g_z + \mu\Big(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\Big) \tag{2.3}$$

Here, gravity is the only body force and it acts along the negative z-axis. The continuity equation for an incompressible fluid is given by:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.4}$$

The Navier-Stokes equations are mathematically classified as second order, elliptic partial differential equations. Therefore, there is a need for boundary conditions for the computational domain. The boundary conditions are specified for the unknowns u, v, w and p in the Navier-Stokes equations and continuity equations. The common boundary conditions are: inflow, outflow, no-slip and symmetry.

2.2 Modeling

The flow in many cases is mainly turbulent because of the viscosity. At critical Reynolds number the flow transits from laminar to turbulent. The turbulent flow displays a chaotic and unpredictable flow field consisting of eddies of various length scales. Because of the chaotic behavior it is nearly impossible to predict the flow around a body. The equations for turbulent flow are solved using different approaches such as Direct Numerical Simulation (DNS), Large Eddy Simulation (LES), Reynolds Averaged Navier Stokes equations (RANS) etc. In this thesis the RANS approach is chosen for the turbulence modeling since it is relatively efficient. The other models like DNS and LES are computationally very costly. Thus, they are not covered in this thesis.

2.2.1 RANS

In Reynolds Averaged Navier Stokes approach, all variables are split into an averaged part and a fluctuating part, $\phi = \bar{\phi} + \phi'$. Inserting the decomposed variables into the Navier-Stokes equations and introducing Einstein notation the governing equations become:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \bar{\rho} \bar{u}_i}{\partial x_i} = 0 \tag{2.5}$$

$$\frac{\partial \bar{U}_i}{\partial t} + \frac{\partial \bar{U}_i \bar{U}_j}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{P}}{\partial x_i} + \frac{\partial}{\partial x_i} \left[\nu \left(\frac{\partial \bar{U}_i}{\partial x_j} + \frac{\partial \bar{U}_j}{\partial x_i} \right) - \overline{u'_i u'_j} \right] \tag{2.6}$$

The time averaging of the Navier-Stokes equations yields to six additional Reynolds stresses. For solving these Reynolds stresses, the turbulent flow has to be modeled. This can be done using an appropriate turbulence model based on assumptions.

2.2.2 Turbulence Model

The turbulent fields in RANS equations are modeled based on Boussineq's hypothesis. Here, the Reynolds stress tensor is assumed to be proportional to the mean strain rate tensor, where the turbulent viscosity(μ_t) is taken as the constant of proportionality (2.7). There are several models for the calculation of turbulent viscosity. Here the case is reviewed using the k-ω SST model.

2.2.2.1 k-ω SST Model

The k-ω SST model is a blend of k-ω model and the k-ϵ model. In the near wall region the k-ω model is used and in the fully turbulent regions, i.e. away from the wall, the k-ϵ model is used. The turbulent viscosity/eddy viscosity is given by,

$$\mu_t = \frac{\rho k}{\omega}.$$

Using the Boussineq's assumption the Reynolds stresses are calculated as,

$$\tau_{ij} = -\rho \overline{u'_i u'_j} = \mu_t \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij}. \tag{2.7}$$

The transport equation of k is given by,

$$\frac{\partial (\rho k)}{\partial t} + \frac{\partial (\rho k \bar{u}_i)}{\partial x_i} = \frac{\partial}{\partial x_i} \left[\left(\mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_i} \right] + 2\mu_t S_{ij} S_{ij} - \frac{2}{3} \rho k \frac{\partial \bar{u}_i}{\partial x_j} \delta_{ij} - \beta^* \rho k \omega, \tag{2.8}$$

where,

$$S_{ij} = \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \ .$$

The transport eauation for ω is,

$$\frac{\partial (\rho \omega)}{\partial t} + \frac{\partial (\rho \omega \bar{u}_i)}{\partial x_i} = \frac{\partial}{\partial x_i} \left[\left(\mu + \frac{\mu_t}{\sigma_{\omega,1}} \right) \frac{\partial \omega}{\partial x_i} \right] + \gamma_2 \left(2\rho S_{ij} S_{ij} - \frac{2}{3} \rho \omega \frac{\partial \bar{u}_i}{\partial x_j} \delta_{ij} \right)$$
$$-\beta_2 \rho \omega^2 + 2 \frac{\rho}{\sigma_{\omega,2} \omega} \frac{\partial k}{\partial x_k} \frac{\partial \omega}{\partial x_k}. \tag{2.9}$$

The model constants $\sigma_k = 1.0$, $\sigma_{\omega,1} = 2.0$, $\sigma_{\omega,2} = 1.17$, $\gamma_2 0.44$, $\beta_2 = 0.083$ and $\beta* = 0.09$. A smooth transition is assured from k-ω model to k-ϵ model with the use of some blending functions.

2.3   OpenFOAM Solvers

OpenFOAM has a collection of libraries dedicated for the solution of partial differential equations. In OpenFOAM solvers used, the continuity, momentum and the turbulence equations are solved separately and the pressure-velocity coupling is done based on an algorithm. The OpenFOAM uses a collocated grid approach and the Rhie-chow interpolation is used for the pressure velocity coupling. Among the standard solvers available in the OpenFoam library, the `simpleFoam` and `pimpleDyMFoam` solvers are used for reviewing the test cases.

2.3.1   simpleFoam

The `simpleFoam` solver is a steady-state incompressible flow solver. The continuity and momentum equations are solved separately. The momentum equation is solved and pressure correction is performed. This solver uses SIMPLE algorithm (Semi-Implicit Method for Pressure Linked Equations) in order to take care of pressure-velocity coupling problem.

2.3.2   pimpleFoam

The `pimpleFoam` is a large time-step transient solver for incompressible flow. The SIMPLE algorithm neglects the unknown velocity corrections, which result a slow convergence. The PISO (Pressure Implicit with Splitting of Operator) algorithm on other hand takes care of the velocity corrections and thus have a faster convergence. The PISO is quite sensitive to initial conditions and has a lesser stability compared to SIMPLE. Therefore in order to retain the accuracy of PISO with the stability of SIMPLE approach, the PIMPLE approach was developed. PIMPLE is a hybrid approach which is a blend of PISO and SIMPLE algorithms.

2.3.2.1   pimpleDyMFoam

This solver is a transient solver for incompressible, flow of Newtonian fluids on a moving/-dynamic mesh using the PIMPLE algorithm. The transient cases such as the open-water propeller condition and self-propulsion condition are solved using this solver.

2.4   Finite Volume Method

Finite Volume Methods are based on a discretization of the intergral forms of the conservation equations. In a finite volume method, the solution is divided into a finite number of control volumes and each control volume is treated as separate system where the governing equations hold. The finite volume approach ensures that the discretization is conservative in both locally and globally. It doesn't require a coordinate transformation for applying for irregular meshes (unstructured meshes). Depending on the type of the mesh (structured or unstructured) the control volumes for every cell node is created differently.

2.4.1   CFL Number

CFL means Courant-Friedrich-Lewy condition. It is a numerical constraint which determine the maximum allowed time step for a specific grid size. It states that the physical domain of dependence must be contained in the numerical domain of dependence. This condition is very necessary for convergence in explicit time integration schemes.

For n dimensional case the CFL has a form:

$$C = \Delta t \sum_{i=1}^{n} \frac{u_{xi}}{\Delta x_i} \leq C_{max} \tag{2.10}$$

where u is the velocity, Δt is the time step and Δx is the length interval.
C_{max} is the maximum value of the CFL number and its value changes with the method used to solve the discretised equation. For an explicit (time-marching) solver the $C_{max} = 1$. For implicit (matrix) solvers the value of C_{max} can be larger.

2.4.2   y+

y^+ refers to the non-dimensional wall distance. It is commonly used for modeling the boundary layer flow by defining the law of wall. The y^+ is given by:

$$y^+ = \frac{u_* y}{\nu} \tag{2.11}$$

Where $u_* = \sqrt{\frac{\tau_w}{\rho}}$ is the friction velocity. For $y^+ < 5$ the flow is in the viscous sub layer ($u/u_* = y^+$). For $30 < y^+ < 200$ the flow is in the log-law region.

2.4.3   Wall Functions

It is often very expensive to resolve the viscous sublayer close to the walls particularly in 3-D. Using fine grid near the wall lead to high aspect ratio cells to which the OpenFoam solvers are very sensitive. One of the traditional way is to use wall functions. The wall functions are functional in the log-law region. This allows to resolve the first computational node around $30 < y^+ < 100$.

2.5   Rotating Motion

There are different approaches available in OpenFOAM for simulating flow around rotating machinery. In the current project, the test cases are analysed using the Multiple Reference Frame technique (MRF) and Sliding Mesh interface based on Arbitrary Mesh Interface technique (AMI).

2.5.1   MRF

In the MRF approach, the steady state solver simpleFoam computes the flow using the rotating and stationary reference frames. Here the Navier-Stokes equations are modified such that the flux is calculated by the relative velocity (u_R) and an additional source term (corolis force, $\Omega \times u_I$) is included in the equations.

2.5.1.1   Governing Equations :

- Rotating zone

$$\bigtriangledown .(u_R u_I) + \Omega \times u_I = - \bigtriangledown p + \bigtriangledown .(\nu_{eff}(\bigtriangledown u_I + (u_I)^T)) \tag{2.12}$$

$$\bigtriangledown .u_R = 0$$

- Stationary zone

$$\nabla.(u_I u_I) = -\nabla p + \nabla.(\nu_{eff}(\nabla u_I + (u_I)^T)) \tag{2.13}$$

$$\nabla.u_I = 0$$

2.5.2 AMI

AMI technique allows the simulation across disconnected adjacent, mesh domains. The domains can be stationary or move relative to one another. This enables to set up a transient simulation with mesh motion such that the propeller movement is realized by the moving part of mesh around propeller geometry.

The patches of the adjacent need to be fairly conformal to one another, such that the sum of weights defining the contribution as a fraction of the intersecting areas shouldn't be less than a threshold value. A zero-gradient condition can be applied to patch faces whose weights are below a user-specified threshold by using `lowWeightCorrection` keyword in the mesh boundary file.

Chapter 3

Open-source Grid generation and Simulation setup

3.1 Grid Development

The generation of accurate mesh is a very crucial aspect for the reliable prediction of flow around the different geometries. The grids can be either structured or unstructured. The structured grid usually considered to have a higher accuracy, but due to geometrical complexities it is quite expensive to generate a properly structured grid around a geometry. Thus, a better solution can be a combination of these two.
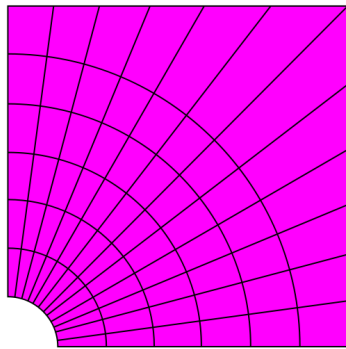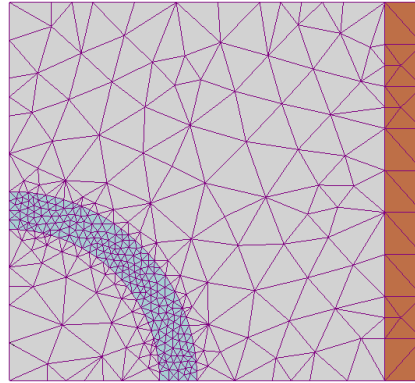
(a) Structured Grid   (b) Unstructured Grid

Figure 3.1: Structured and Unstructured grid patterns

OpenFOAM supports structured and unstructured mesh cells of any shape. OpenFOAM allows the freedom to have cells with multiple number of faces and edges. This creates the flexibility for the generation of meshes around complex shapes and embeded refinement. In this project the test cases are meshed using the `blockMesh` and `snappyHexMesh` utilities in OpenFOAM.

3.1.1 The blockMesh tool

The `blockMesh` is a multi-block mesh generator that generates meshes of hexahedra type from a text configuration file (`blockMeshDict`). In the configuration file the mesh inputs such as the reference vertices of the mesh, mesh grading definition and cell count over are

defined. Additionally, once can assign specific edge definitions on blocks such as arcs and splines. The faces of the blocks can be assigned to the patch names along with the option of face merging or merging of patches. The `blockMesh` tool is generally used for creating a structured background domain mesh initially before snapping of the geometry.
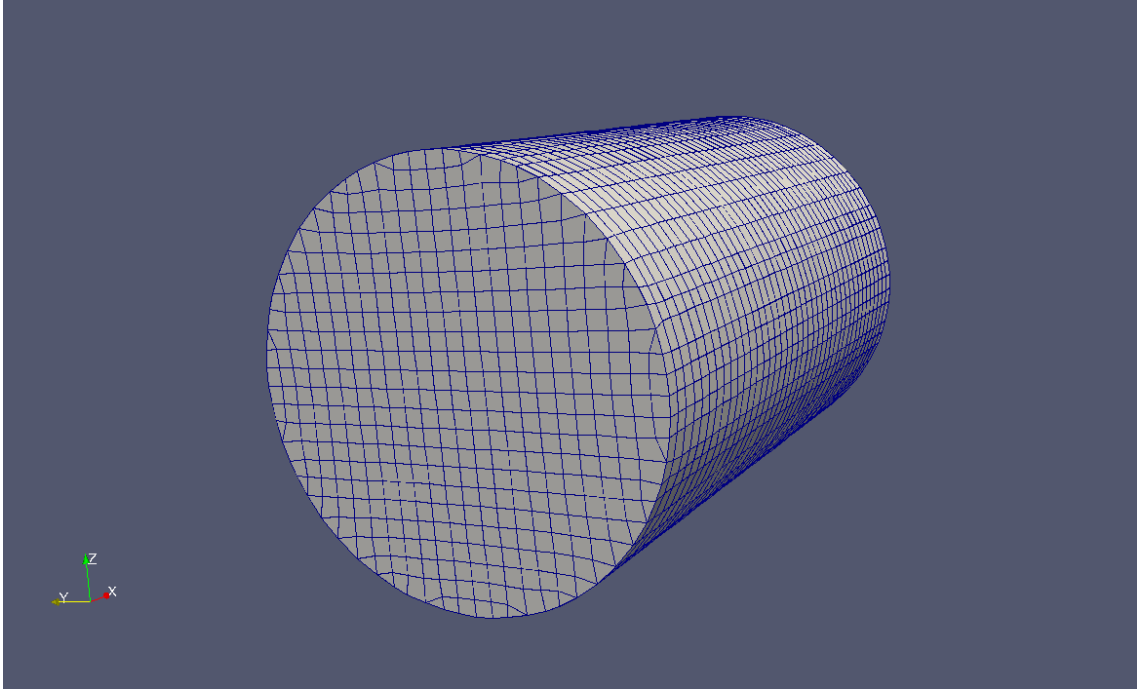
Figure 3.2: Cylindrical mesh domain for open-water test case generated using blockMesh

3.1.2 The snappyHexMesh tool

The `snappyHexMesh` is an automatic, parallel, octree-refinement based mesh generation utility in OpenFOAM. This tool creates Cartesian hexa-dominant meshes and can adequately handle complex geometries. This process is done iteratively refining the background mesh (created using blockMesh) and morphing the mesh to the surface. The mesh generation is based on four steps: background mesh, castellation, snapping and layer addition. As discussed in earlier section, the background mesh is created using blockMesh. In the castellation phase the cells which do not belong to the computational domain are removed (eg. for external flows the cells inside of the geometry are removed). In the third phase, the cells close to the geometry surface are projected according to the geometry and further smoothed according to the mesh quality definitions, this can can implemented both explicitly or implicitly. In the layer addition phase, the prism layers are inserted by pushing the mesh away from the surface with user specified thickness and expansion ratio. The snappyHexMesh can be implemented in parallel and is very convenient for large cases (upto 100M cells).

The `snappyHexMesh` is a very easy tool to generate a mesh but still lacks some important features. Since snappyHexMesh cuts through the surface and generates its own mesh, its sometimes generate a poor quality mesh especially in creating the boundary layer regions. To yield a correct solution this mesh region needs to be very fine. The snappyHexMesh works well for simple geometries but for more complex surfaces (propeller trailing edges and

transom region of hull) it might not create the layers in few regions or the mesh may have bad properties (negative cells, orthogonality etc.). The source code of the snappyHexMesh has been modified based on a tutorial by Christofer Järpner(2011)[10], so that boundary layer mesh with adequate quality is generated for the test cases. The modified mesh tool from the snappyHexMesh works well for the test cases but sometimes generate high aspect ratio cells to which OpenFOAM is quite sensitive. Therefore, one need to take care of the refinement and by validating the mesh in every phase of meshing. OpenFOAM has several utilities that perform mesh checking and manipulation. The meshes for the test cases described in section 3.1.3 are generated using snappyHeshMesh. As described in section 1.3, there are ten test cases which can be classified into three types: bare hull, open-water and self-propulsion cases.

3.1.3 Bare Hull
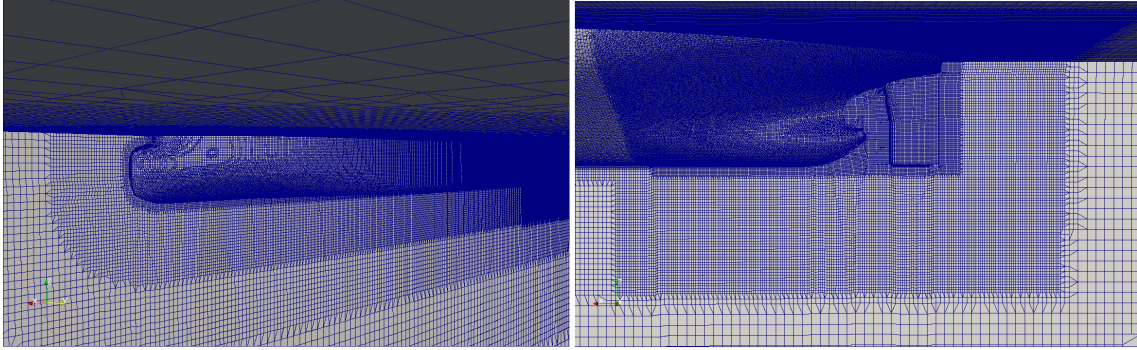
Figure 3.3: Computational mesh at fore and aft body of the bare hull generated using snappyHexMesh

This test case has a box shaped computational domain of depth and width 2 times of the model length. The length of the domain extends to 1 model length forward of the bow and 1.5 model lengths aft of the transom . The domain was meshed from 6 million cells to 16 million cells for model scale case for the mesh dependency studies. For a full scale case the domain was meshed with around 67 million cells. The mesh was a combination of both hexa and tetra cells. The mesh is much refined near the transom region of the hull because of the complexity of the geometry. The model scale bare hull geometry was also meshed using the ICEM cfd tool in order to check the accuracy of the open source mesh and simulation with commercial software(Fluent).

Table 3.1: Average values of y+ for the bare hull mesh

Case	Scale	Avg. y+
bareHull (without transom)	model	30-40
bareHull transom	model	≈ 1
bareHull (ICEM)	model	≈ 1
bareHull	Full scale	90-120

3.1.4 Openwater

The openwater simulation has two cases, one with conventional propeller and one with unconventional propeller. The test case has a cylindrical shaped domain of diameter 3 times the propeller diameter and its length extends to 3 times the model length. The mesh domain is split into two domains, stationary domain and rotating domain. The high density mesh region in Figure 3.4 is the rotating domain. The rotating domain is a cylindrical domain of 1.5 times the diameter of the propeller and length equal to the propeller diameter. Both domains are meshed to have AMI capability. The domains are meshed from 2 million cells to 6 million for model scale cases (both conventional and unconventional). For a full scale the domains are meshed from 10 million to 20 million cells.
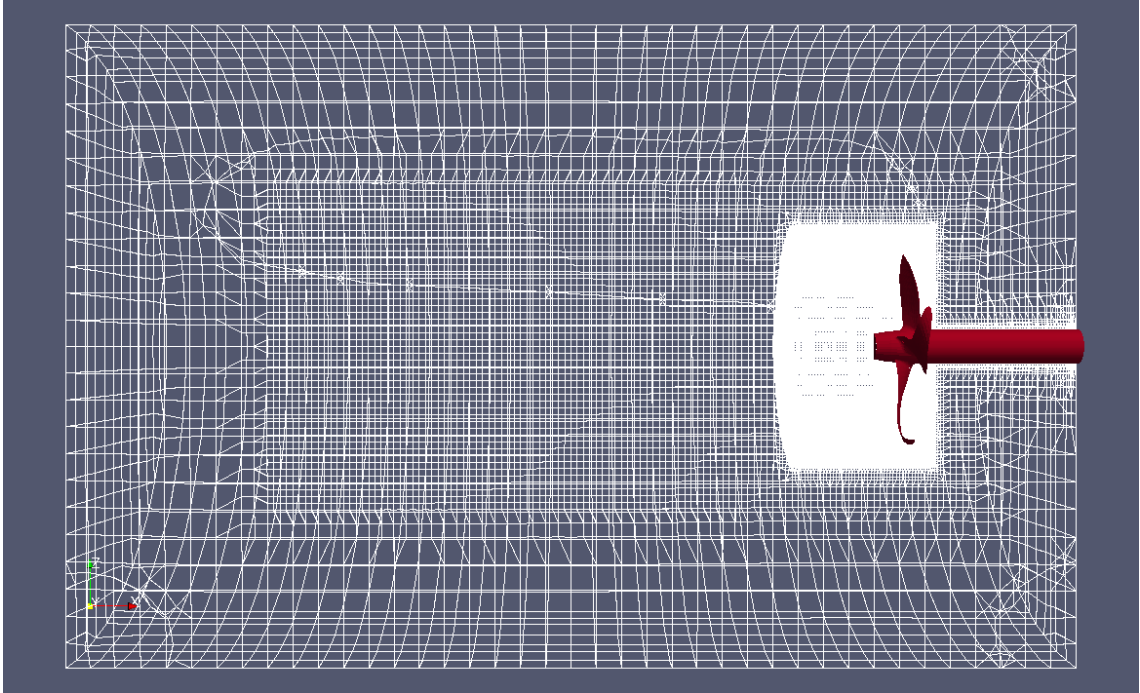
Figure 3.4: Computational mesh for an open-water case (AMI)

Table 3.2: Average values of y+ for the Openwater case

Case	Scale	Avg. y+
Open-Water(conv)	model	≈ 1
Open-water(unconv)	model	≈ 1
Open-Water(conv)	full scale	90-95
Open-Water(unconv)	full scale	90-95

3.1.5   Self-propulsion

This test case contains two geometries, the bare hull and the propeller, and the simulation has two cases, one with conventional and other with unconventional propeller. The size of the computational domain is much same as the bare hull case. Like the open-water case the computational domain is split into two domains, stationary for bare hull and rotational for propeller. The high density mesh region in Figure 3.5 is the rotating domain. The rotating domain is a cylindrical domain of 1.5 times the diameter of the propeller and of adequate length such that its fits between the hull-rudder region. The whole domain is meshed have AMI capability. The domain is meshed from 12 million cells to 18 million cells for the model scale cases and for full scale case its is meshed with around 70 million cells.
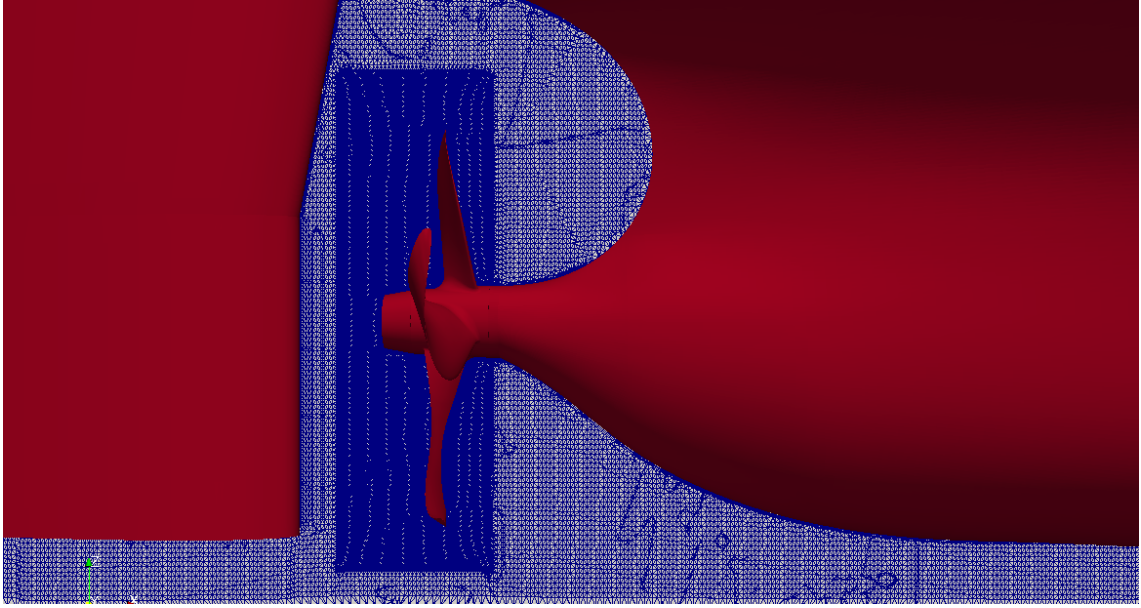
Figure 3.5: Computational mesh near propeller region for self-propulsion case

Table 3.3: Average values of y+ for the self-propulsion case

Case	Scale	Avg. y+
Self-propulsion (without transom and propeller)	model	30-40
Self-propulsion (transom and conv. propeller)	model	≈ 1
Self-propulsion (transom and unconv. propeller)	model	≈ 1
Self-propulsion conv.	Full scale	90-120
Self-propulsion unconv.	Full scale	90-120

3.1.6 Grid Study

In a CFD analysis, its often very important to ensure that the solution is independent of the grid size. This study is required in order to quantify the accurate level of grid refinement for a case, so that one can choose an efficient mesh in which the solution is invariant with much finer meshes. Table 3.4 shows the number of mesh cases and range to cells for the grid studies. The variable chosen for this study is the total resistance/thrust force of the test case. The maximum tolerance of this variable is 2%.

Table 3.4: No of mesh cases for grid dependency studies

Case	Scale	No. of cells range	No. of mesh cases
Bare Hull	model	6-16 M	4
Open-water conv.	model	2-6 M	3
Open-water unconv.	model	2.4-6.2 M	2
Self-propulsion conv.	model	12-18 M	3
Self-propulsion unconv.	model	12-18 M	2
Bare Hull	full scale	67 M	1
Open-water conv.	full scale	10-20 M	3
Open-water unconv.	full scale	20 M	2
Self-propulsion conv.	full scale	70 M	1
Self-propulsion unconv.	full scale	73 M	1
		total	22

3.2 Simulation setup

This section deals with the case set up of the simulation along with the file structure and organisation in OpenFOAM. In OpenFOAM a certain directory structure for a set of files to run the applications is expected. Normally a case is setup in three directories, `constant`, `system` and a Time directory which is usually named 0, it 0 is the starting time. Figure 3.6 shows the directory setup for the simulations.
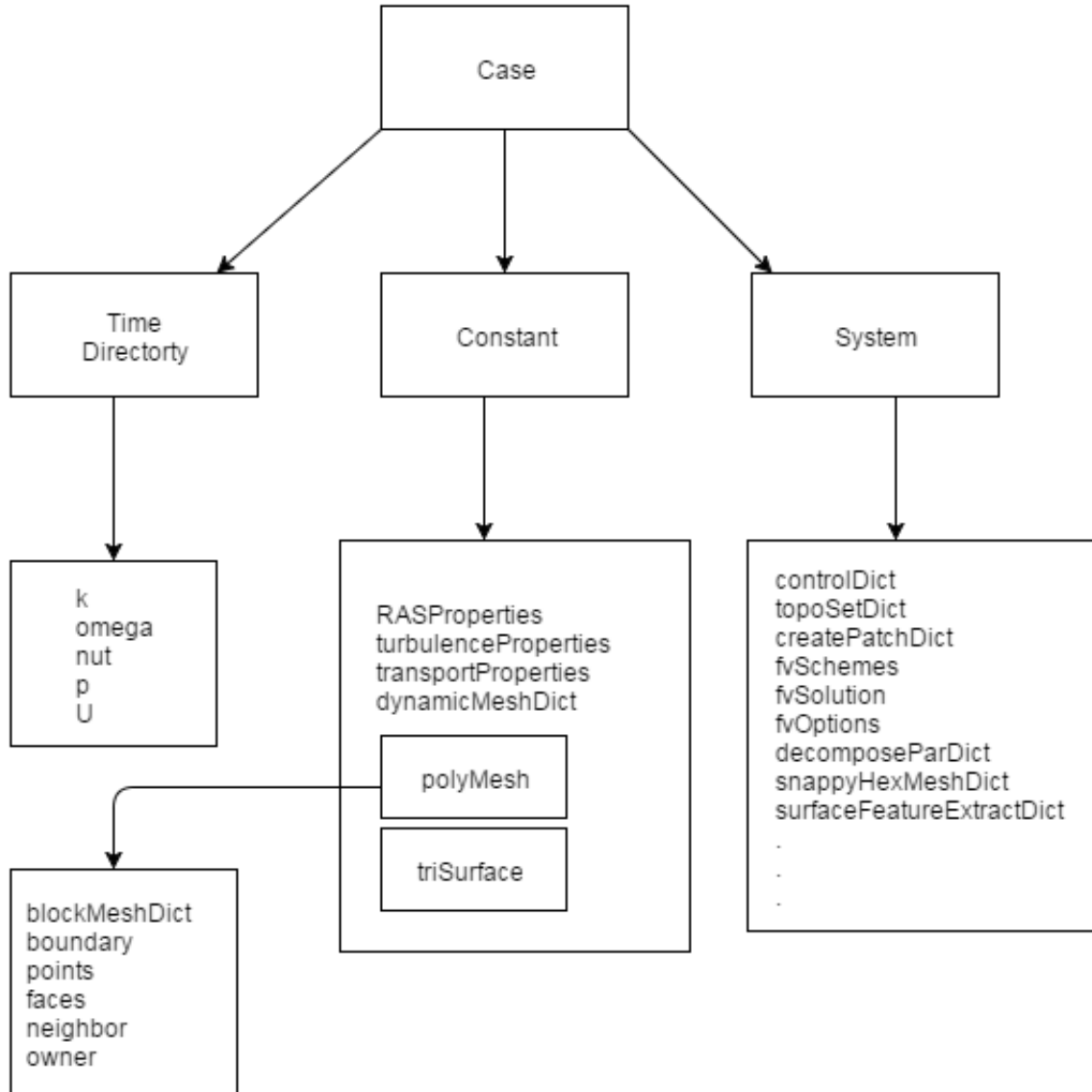
Figure 3.6: Case setup in OpenFOAM

3.2.1 Time directory

The 0 folder contains the files with the initial and boundary conditions of the flow variables. For simulating incompressible flows using $k - \omega$ SST RANS model, the files U, p, nut, k and omega are needed. The boundary conditions for turbulence quantities are calculated assuming an arbitrary turbulent intensity (I) and model constants.

3.2.2   Constant directory

This folder contains turbulence and transport properties of the fluid. The turbulence model is defined in `RASProperties` and kinematic viscosity (ν) is defined in the `tranportProperties`. The `dynamicMeshDict` is used to define the conditions required for the dynamic mesh operations.

The `polyMesh` folder contains the block mesh dictionary and also contains the files which contain the data of the mesh. The `boundary` file in one of such file, where the names of the boundaries an be assigned (empty, symmetry etc.). In the `triSurface` folder the geometry files required for the simulation are store in `.stl` format.

3.2.3   System directory

The files which define the specifications of the simulation are contained in the system folder. The `snappyHexMesh` dictionary and other mesh manipulations dictionaries must be defined inside system folder. The `decomposeParDict` file contains the mesh decomposition definitions where mesh can be decomposed into adequate number of parts for parallel simulations. Apart from the simulations the `snappyHexMesh` tool also can be implemented in parallel for large number of cells (around 100M).

As discussed earlier, the OpenFOAM solvers begin to run by setting up a database. The `controlDict` dictionary defines the time and data input/output control and sets the input parameters essential for creating a database. The inputs like run time, time steps and Courant number etc. are assigned in controlDict.
The `fvOptions` file contains additional finite volume options which represent additional source terms or constraints (such as Coriolis forces etc.) on the governing equations. This file is very much essential in order to define the Multiple reference frame technique (MRF) simulation type.

3.2.3.1   Solver settings

The solvers for the equation, tolerances and the algorithms are controlled by the `fvSolution` dictionary. OpenFOAM uses linear solvers for system of equations and these are of 4 types; Preconditioned (bi-)conjugate gradient (PCG/PBiCG), smooth solver, Generalised geometric-algebraic multi-grid (GAMG), and Diagonal solver for explicit systems (diagonal). The pressure correction equation is solved using GAMG algorithm and turbulence equations are solved using smoothSolver. The velocity equation is solved by using PBiCG algorithm and the preconditioning of matrices in this algorithm is done by Diagonal incomplete-LU (DILU) preconditioner. The Gauss-seidel smoother is used for multi-grid smoothing. The Under-relaxation factors are used in order to improve the stability of the computation and the residual levels are controlled by setting the solution tolerances (tolerance and relTol keywords).

3.2.3.2   Dicretisation

The numerical discretisation schemes for the modeled equations are assigned in `fvSchemes` file. The numerical schemes can be either for gradient, divergence, laplacian, interpolation or time dicretisation derivatives. Additionally this file has a sub-dictionary (`fluxRequired`) that lists the fields for which the flux is generated in the application. The first/second order schemes were used for the dicretisation of governing equations.

For the gradient terms, first order Gauss linear scheme is used and for the momentum divergence term ($\bigtriangledown.(\rho UU)$) , the second order scheme Gauss linear upwind is used for the steady state cases and Gamma V scheme (NVD scheme) is used for transient cases. The turbulent terms are discretised using bounded Gauss upwind scheme.

For the laplacian term ($\bigtriangledown.(\nu \bigtriangledown U)$) in the governing equations, the Gauss scheme is the only choice of discretisation and requires an interpolation scheme for the diffusion coefficient. The second order conservative Guass linear corrected scheme is used for the laplacian term.

For the transient cases, the time derivative terms ($\frac{\partial}{\partial t}$) are dicretised using the first order bounded implicit Euler scheme. The interpolation of values from cell centres to face centres are assigned in `interpolationSchemes` sub-dictionary. Out of the available schemes the liner interpolations scheme is chosen as it is more general.

3.2.4 Boundary Conditions

The treatment of the boundary is an essential part in flow simulation which can effect the solution. In OpenFOAM the case is generally broken into set of patches and the boundary conditions are then assigned as attributes to the patches. The patch attributes in OpenFOAM are of three types: Base type, primitive type and derived type. The base type attributes are described in terms of geometry (e.g. wall, patch, cyclic, symmetry etc.). The base type are generally defined in the `boundary` file in the `polyMesh` folder. The primitive type attributes are assigned to the field variables on a patch (e.g. fixedValue, zeroGradient, calculated etc.). The derived type is also assigned to the field variable but is of a complex patch condition which is derived form primitive type (e.g. inletOutlet and outletInlet). Both primitive and derived type are defined in the field files in the time directory.

3.2.5 Convergence

A criterion is enforced in such a way that the change in a chosen calculated value (residuals or force monitors) is evaluated for each time step and when the criterion meets the assigned residual tolerance for N number of time steps (say 1000) the solution is evaluated as converged. In this study, the forces like thrust and resistance are monitored for many of the cases. The fluctuations between the residuals is around 1%.

3.2.6 Post Processing

The main post-processor used in this study is Paraview, which is open-source, multi-platform data analysis and visualization application. Paraview is based on the visualization tool kit (VTK) and in OpenFOAM the results can be converted to the VTK by using `foamToVTK` utility. A wrapper of Paraview (`paraFoam`) is distributed along with Open-FOAM which makes paraview GUI modified according to the specifications of OpenFOAM.

Another post-processing tool used in this study is Tecplot. Tecplot is a commercial visualization software which can post process flow data and grid quantities. In this study, Tecplot is used for creating the polar contour plots for wake field.

3.2.6.1 Flow structures

The visualization of the flow is presented using Iso-surfaces of the second invariant of velocity gradient (Q). Introducing Einstein's notation the Q is represented as follows;

$$Q = \frac{1}{2}(\Omega_{ij}\Omega_{ij} - S_{ij}S_{ij}) \tag{3.1}$$

where,

$$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right)$$

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

Here, Ω is the vorticity tensor and S is the rate of strain.

3.2.7 Process flow

Figure 3.7 shows the entire process on how the cases are analysed. Since the Open-FOAM is very sensitive to the quality of the mesh, the mesh quality is evaluated using the `checkMesh` utility. The meshes in this study are created such a way that the quality of parameters like mesh topology, patch topology, boundary openness, non-orthogonality, minimum volume, face pyramids are retained.

The experimental results are obtained from the tank tests at SSPA and are used for validating the simulation results for models scale. This is done in order to ensure the accuracy of the solution and the solving method in RANS. Further the same solving methods are used for setup of full scale simulations. The scaling effects on different propulsive coefficients and efficiencies are studied and compared with the outcomes of the ITTC 78 method.
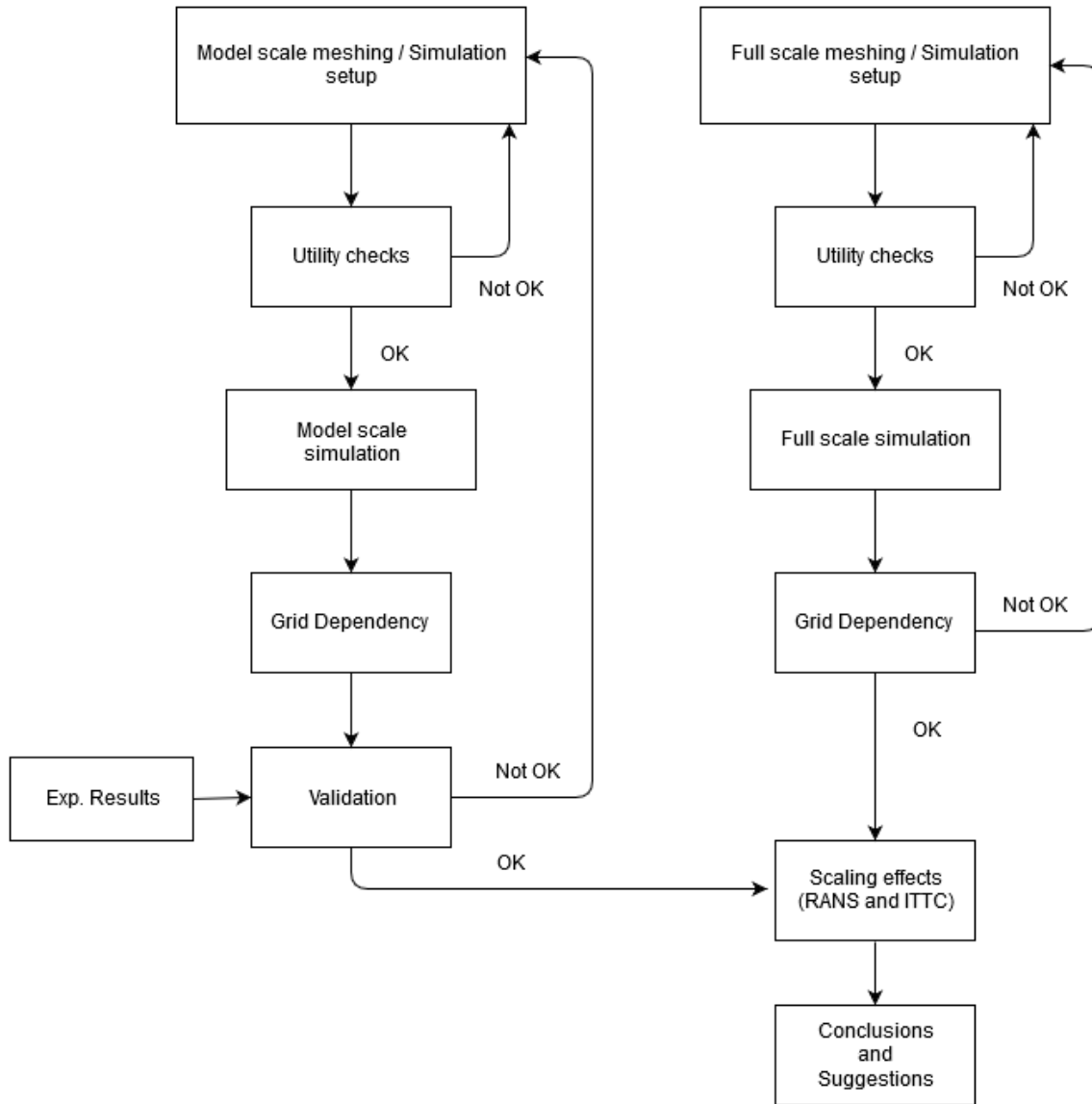

Figure 3.7: Process flow

Chapter 4

Results and Discussion

4.1 Bare Hull Model

In this section, the results of the bare hull simulation are presented. The bare hull used in this study is from a tanker vessel at which the two propellers (conventional and Unconventional) operate. As discussed in the previous chapter, the domain is a box shaped domain. The boundary conditions are assigned to the flow domain in such a way that the Inlet and Outlet have `inletOutlet` and `outletInlet` respectively. The side, upper and bottom faces are assigned with slip/symmetry boundary conditions. For the bare hull, the wall boundary conditions is used (no slip). The flow is simulated using steady state solver (simpleFoam) without a free surface at the design froude number (0.15). The effect of free surface is neglected as the vessel is steaming with lower froude number and in order to reduce the complexities in meshing, solution process and convergence. The results such as resistance and nominal wake field at the propeller disk are compared with the towing tank test at SSPA. Further the test case is simulated using commercial CFD tool fluent and compared with the results of OpenFOAM. Figures 4.1 and 4.2 show the axial velocity field around the hull. This is performed in order to ensure the accuracy of the open-source tools. Table 4.1 shows the comparison of the resistance of the simulations and experimental tests. The results from OpenFoam is quite consistent with results from Fluent. However, there is a difference of 5% with experimental result which may be partially due the effect of free surface. Figure 4.3 shows the comparison of nominal wake field at propeller disk between experimental tests and the simulations. The results show similar trend in the wake with some differences. This could be due to various effects like the free surface and other interferences in experimental setup. However the simulated wake would be considered for the study of the scaling effect.

Table 4.1: Comparison of resistance coefficients using different methods

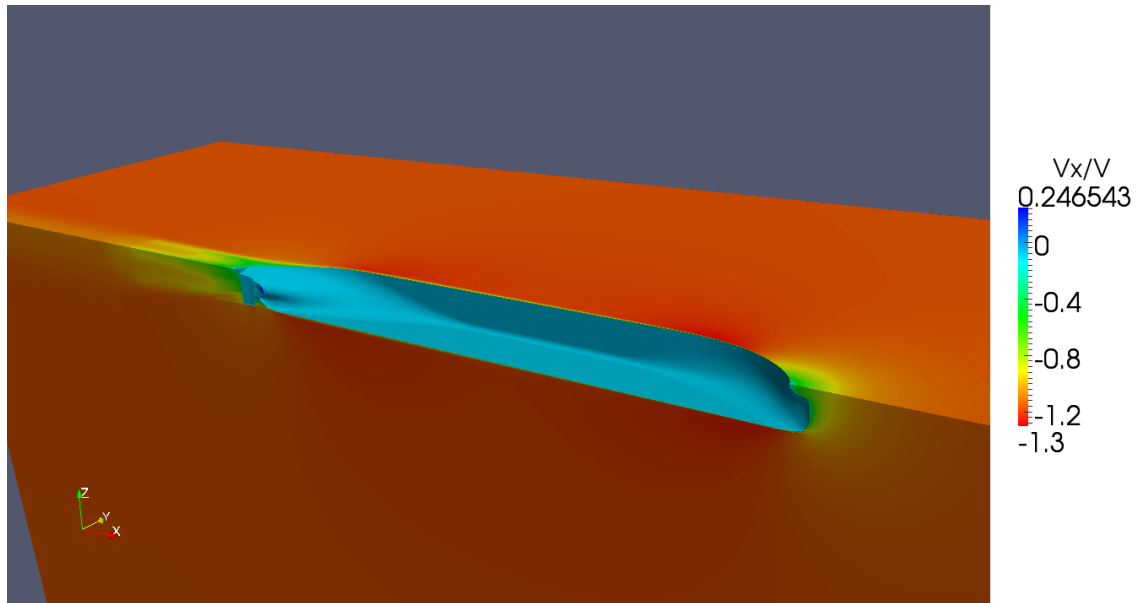| *	OpenFOAM (snappy)	Fluent (ICEM)	Experiment
Resistance coeff.	0.042165	0.042246	0.044343

Figure 4.1: Resistance prediction of bare hull (OpenFoam) coloured by axial velociy variation
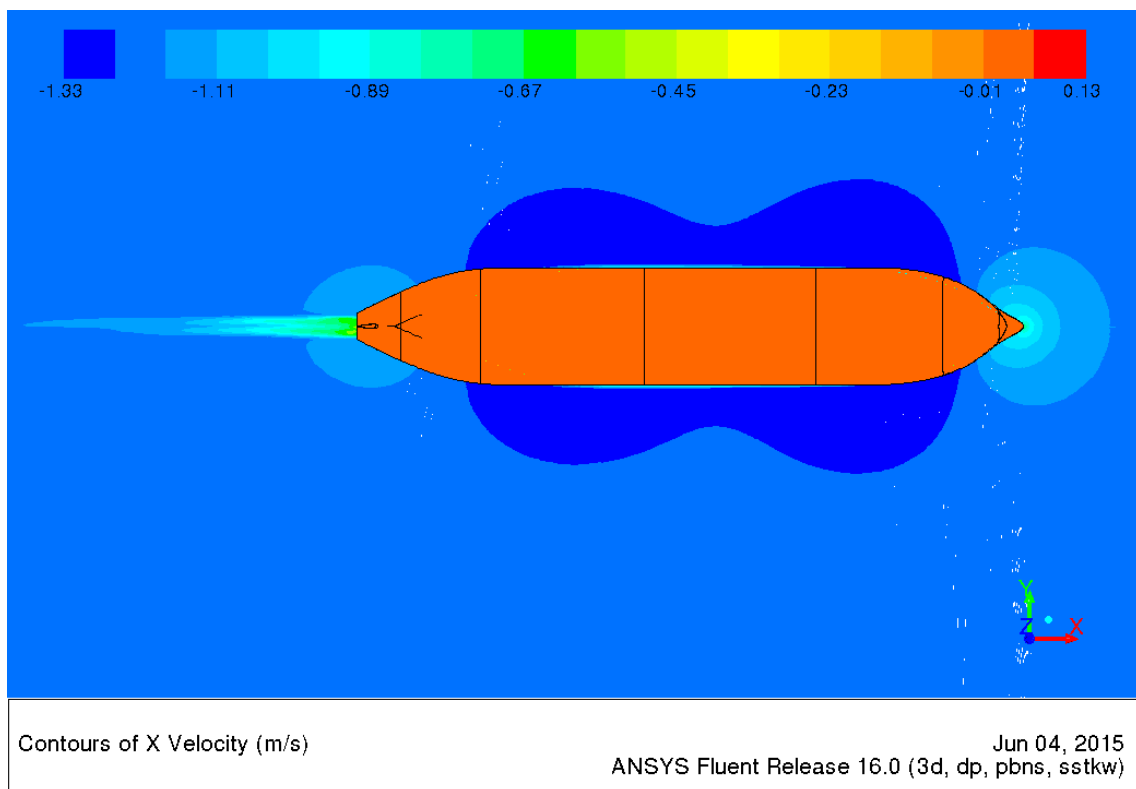

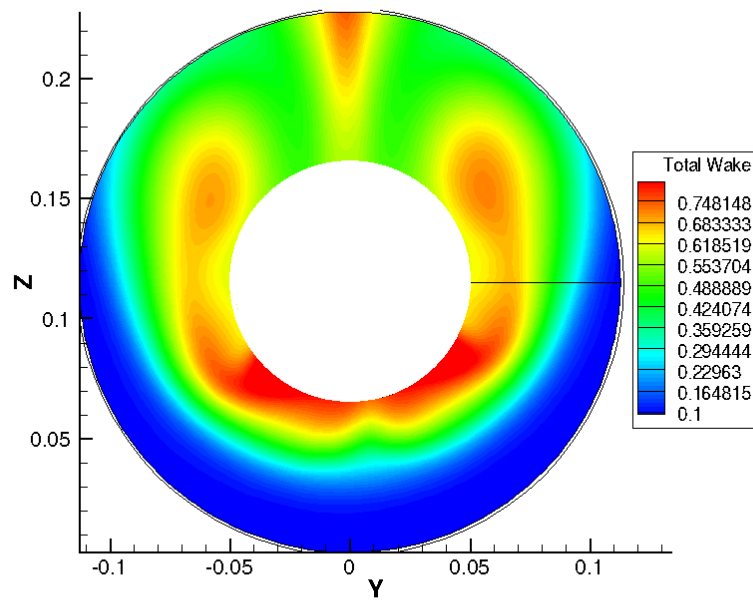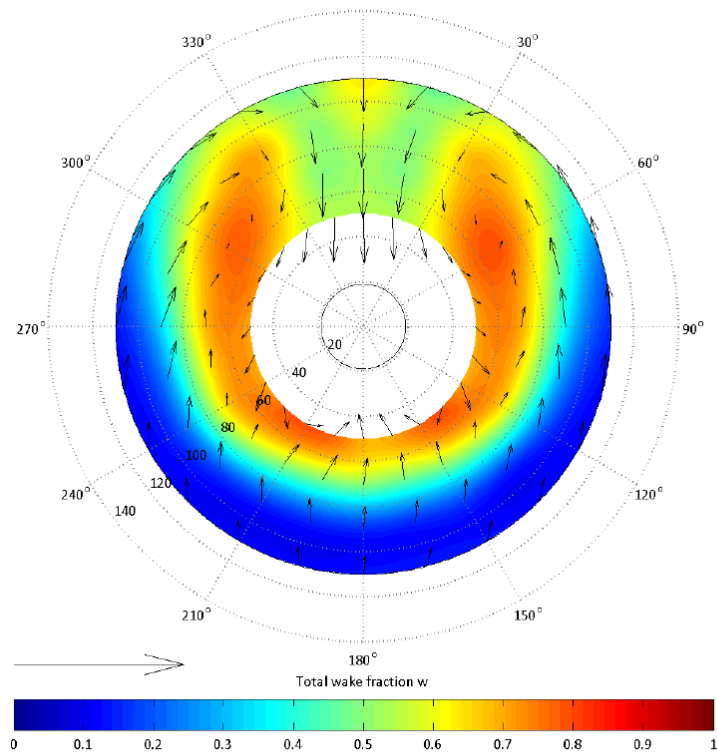**Figure 4.2:** Resistance prediction of bare hull (Fluent) coloured by axial velociy variation

Figure 4.3: Wake at the propeller disk from towing tank tests (above) and through simulations (below)

4.2 Open-water Model scale

In this section, the model scale open-water simulations of both conventional and unconventional propellers which are designed for same operating condition are described. Both the propellers are four bladed and have a shaft length of 1.5 times of the propeller diameter. The computational domain (Figure 3.2) is of cylindrical in shape and mesh has two interfaces (stationary and translational). Inlet and outlet boundaries are assigned to the faces of the cylinder as `fixedValue` (free stream velocity is assigned) for inlet and `inletOutet` for outlet. The cylindrical surface has a symmetry/slip boundary condition. The propeller geometry has a wall type patch attribute with `movingWallVelocity` as velocity boundary condition and `zeroGradient` for pressure boundary conditions. The turbulent variables (k, ω and ν_t) for the wall have a fixed assigned values. The flow around the propellers is simulated using RANS method with translational grids (AMI) using the pimpleDyMFoam solver. As methioned in the previous chapter the Euler scheme is choosen for the time discretization.

In order to evaluate the accuracy of the RANS simulations, it is necessary to compare the thrust (T), torque (Q) and efficiency of the propellers both by computation and experiments. The flow around the test cases are computed at different advance ratios (J) in order to plot the thrust, torque and efficiency curves of both the propellers. The nomenclature for the coefficients of thrust (K_T), torque (K_Q), and efficiency (ETA_0) are as follows:

$$K_T = \frac{T}{\rho n^2 D^4} \tag{4.1}$$

$$K_Q = \frac{Q}{\rho n^2 D^5} \tag{4.2}$$

$$ETA_0 = \frac{J * K_T}{2\pi * K_Q} \tag{4.3}$$

where D is the diameter of the propeller, n is the rotational speed of the propeller and ρ is the density of the water. Figures 4.5 and 4.7 show the K_T, K_Q, and ETA_0 curves for both the propellers. The solid lines represent the results of model scale RANS computations and solid symbols indicate the experimental data. The transient RANS computations showed a good correlation with the experimental data (diff. < 2%) for both conventional and unconventional propeller. Further the propeller test cases are simulated using the steady-state multiple reference frame technique (MRF, see section 2.5.1), in order to compare the two methods (AMI and MRF). The steady-state computations also showed a good correlation with the experimental data and the transient simulations.

At a higher advance ratio (J > 1) the RANS simulations show a small separated leading edge vortex apart from the usual tip vortex (Figure 4.8 and 4.9). The effect of this phenomenon is not studied as the advance ratios above 0.6 doesn't fall under the design conditions. Further, in order to resolve this separation properly, the mesh has to be much refined at the separation regions and it necessitates the use of higher order models (e.g. eddy viscosity models).

Figure 4.4: Iso-Surface Q coloured by Ux magnitude for conventional propeller
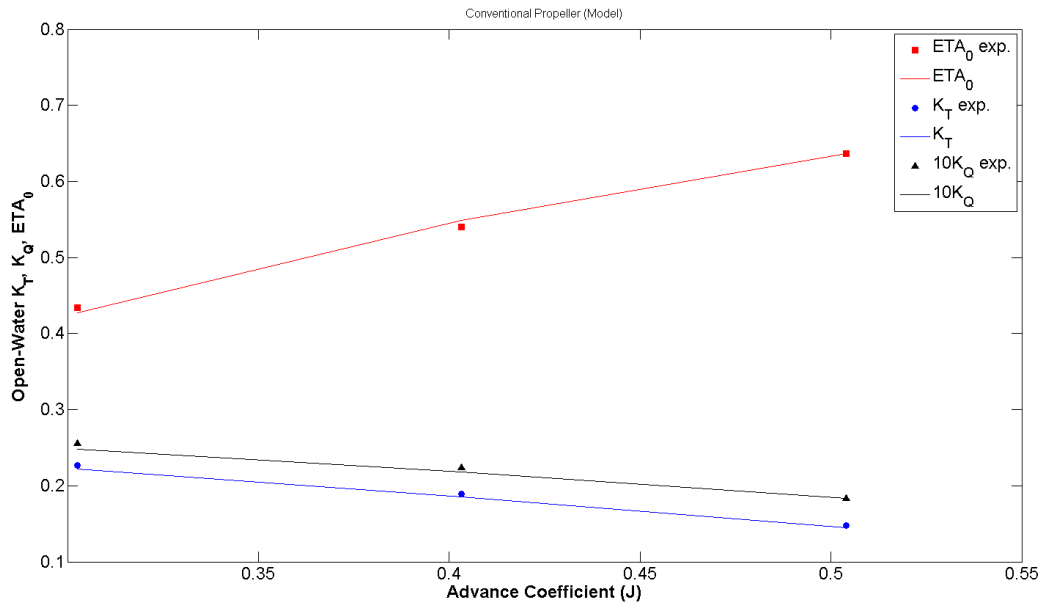
Figure 4.5: Open-Water thrust, torque and efficiency for conventional propeller at model scale

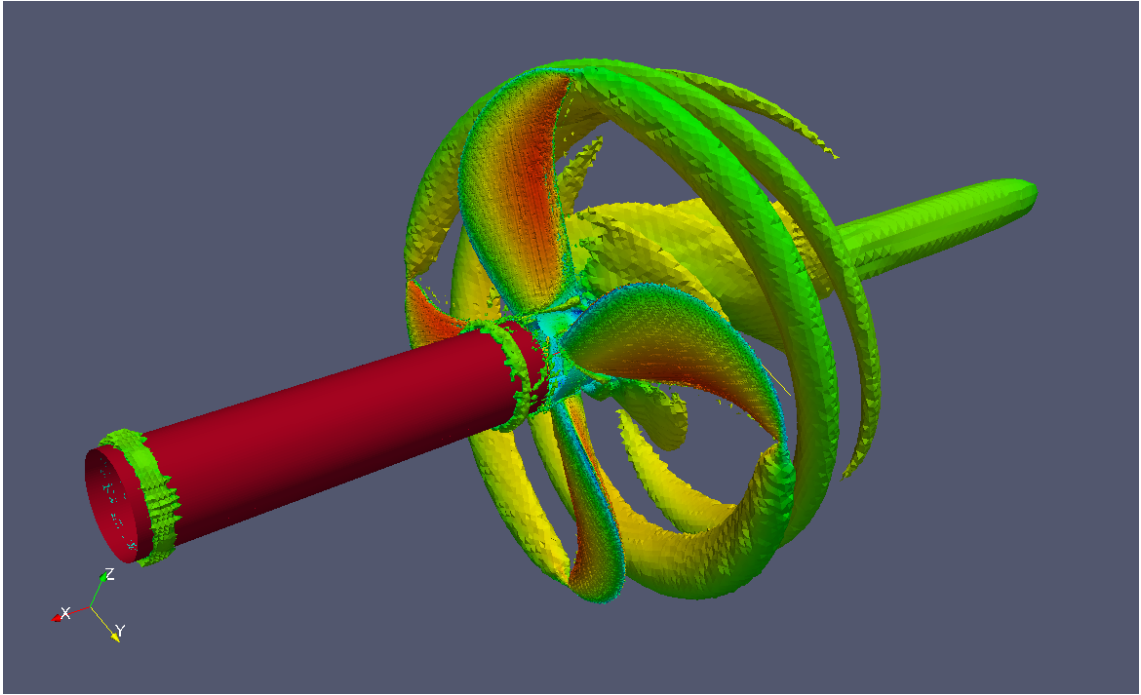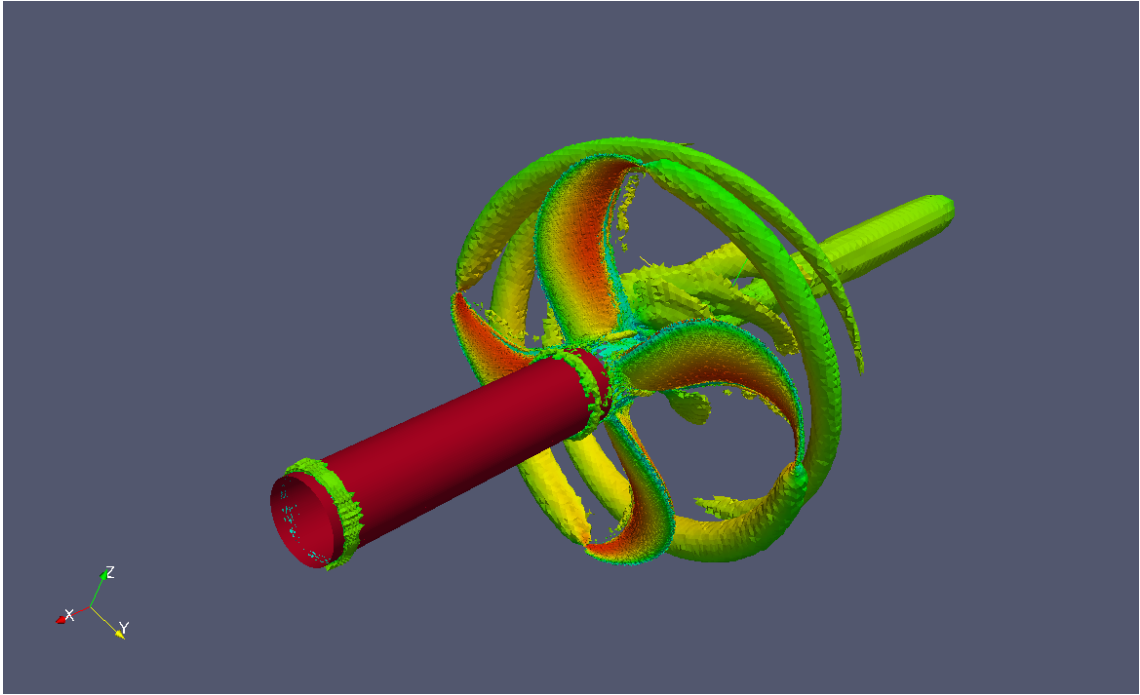**Figure 4.6:** Iso-Surface Q coloured by Ux magnitude for unconventional propeller
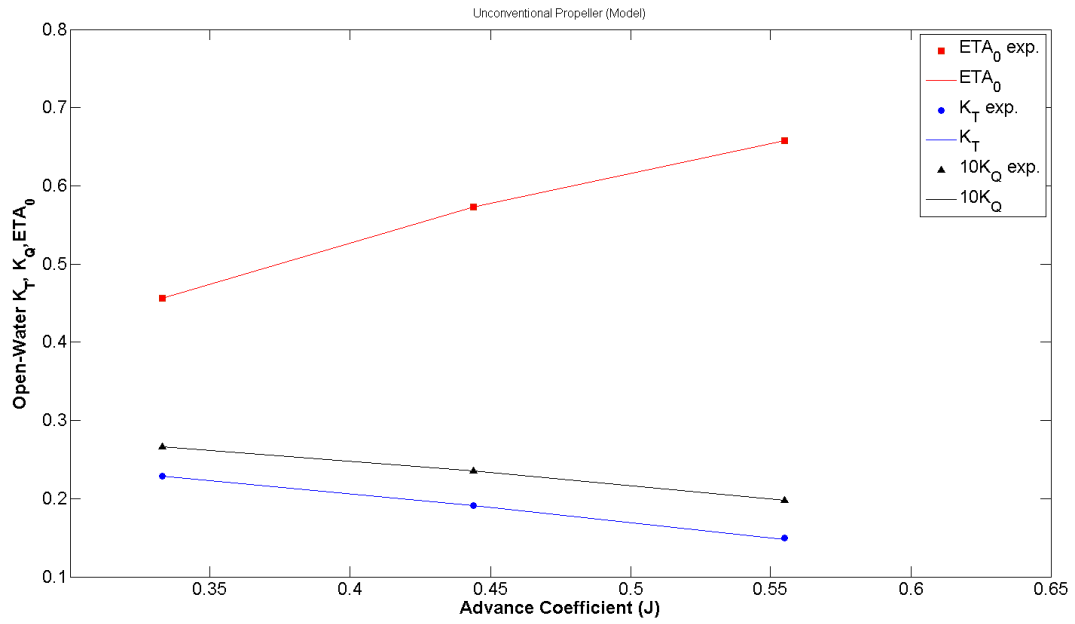

Figure 4.7: Open-Water thrust, torque and efficiency for unconventional propeller at model scale
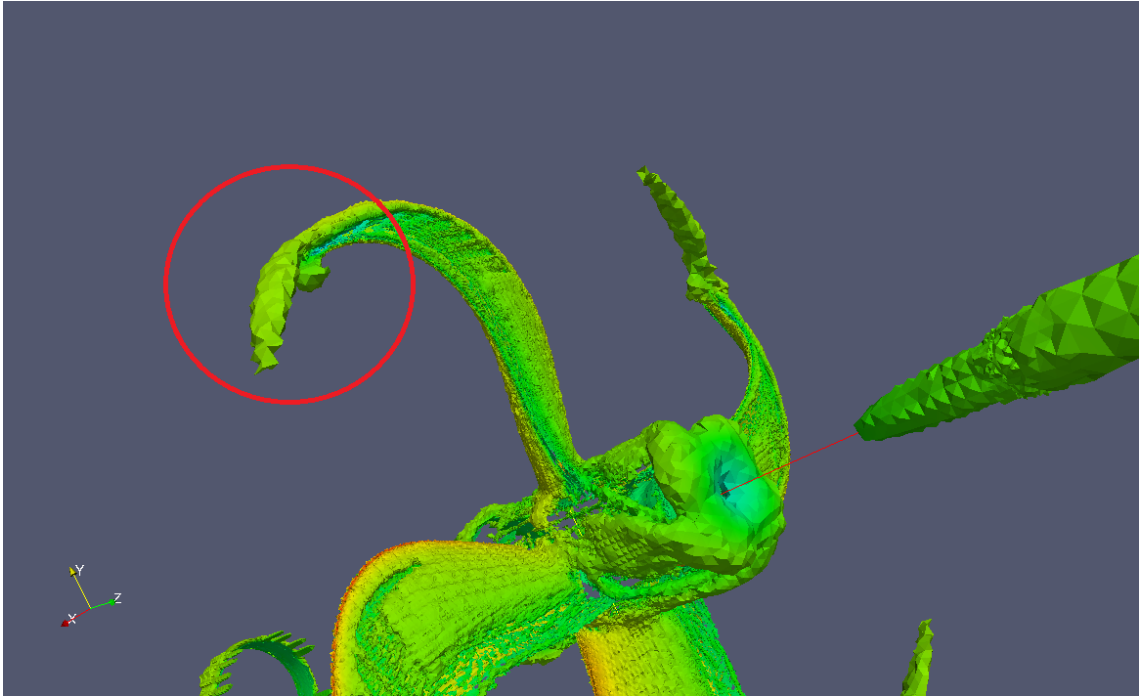
Figure 4.8: Separated leading edge vortex (red circle) of the unconventional propeller at J ≈ 1
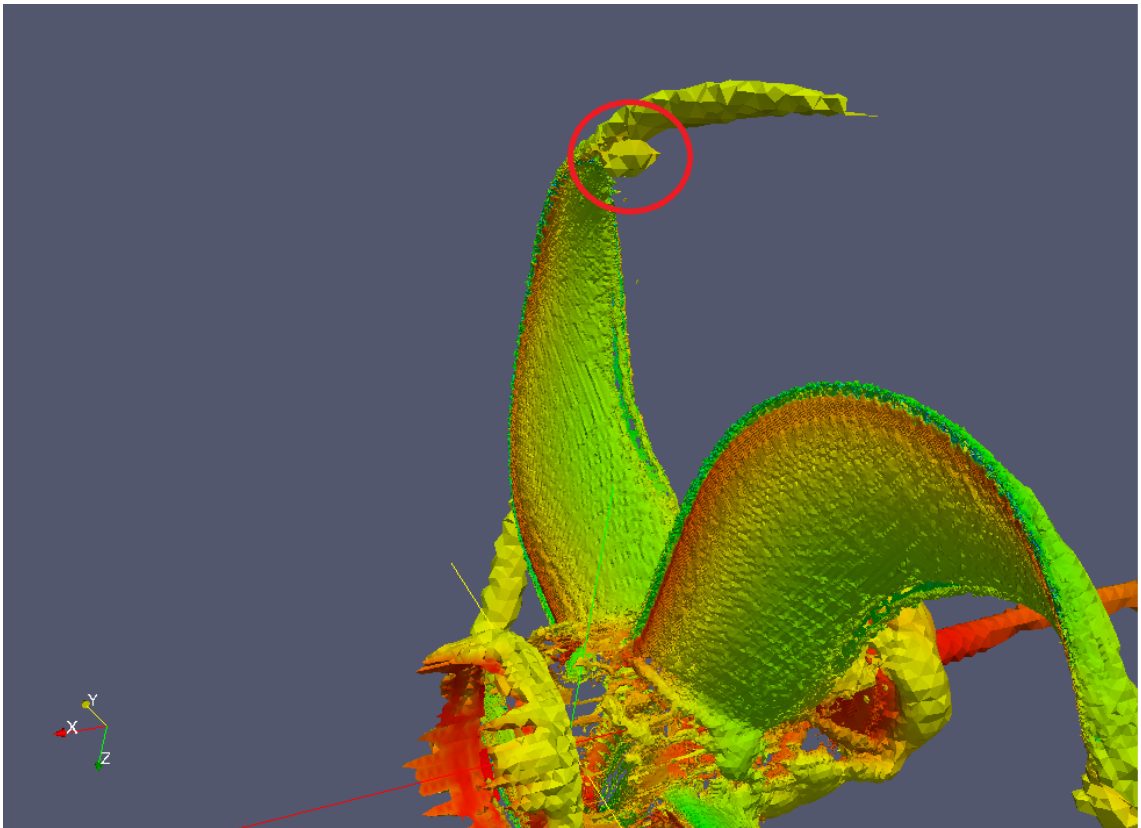
Figure 4.9: Separated leading edge vortex (red circle) of the unconventional propeller at J ≈ 1

4.3   Self-Propulsion Model scale

In this section, the performance of the two propellers operating behind a hull is evaluated using RANS in model scale. The domain is a box shaped domain congruent to the bare hull case. Like the bare hull case, the boundary conditions are assigned to the flow domain in such a way that the Inlet and Outlet have `inletOutlet` and `outletInlet` respectively. The side, upper and bottom faces are assigned with slip/symmetry boundary conditions. For the bare hull `noSlip` boundary condition is used. The propeller geometry has a wall type patch attribute with `movingWallVelocity` as velocity boundary condition and `zeroGradient` for pressure boundary conditions. The turbulent variables (k, ω and ν_t) for the wall have a fixed assigned values. The flow around the propellers is simulated using RANS method with translational grids (AMI) using the pimpleDyMFoam solver.
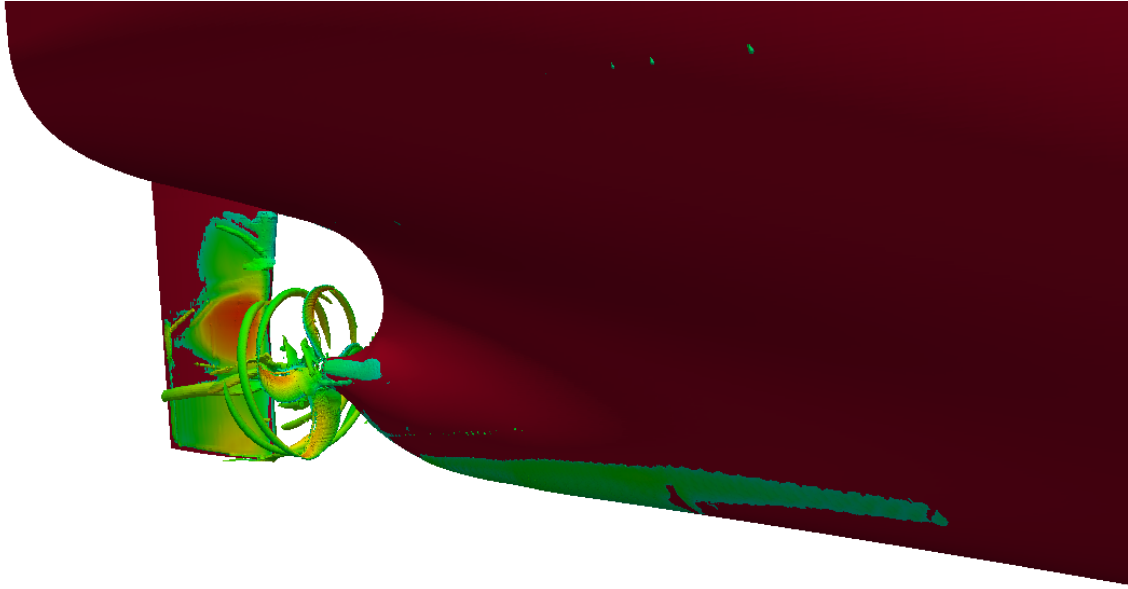
Figure 4.10: Iso-Surface Q coloured by Ux magnitude for unconventional propeller

Here, the flow is simulated at the same ship speed and propeller revolution rate as the model tests. The performance is ranked between the two propellers both by CFD and experimentation. Figure 4.11 shows the difference in propulsive coefficients of unconventional propeller with the conventional propeller at the model test condition. The ranking showed a similar trend in the comparison between the propellers. Figure 4.12 compares the effective wake, thrust deduction, hull efficiency and propeller efficiency between conventional and unconventional propeller. In model scale the unconventional propeller obtains a higher effective wake compared with the conventional propeller. The propeller efficiency is lower for unconventional propeller and hull efficiency is higher compared with the conventional propeller. Figure 4.12 shows a series of plots of axial velocities at different longitudinal positions at non dimensional distance (X/D) 0.28, 0.18, 0.10 respectively in upstream of the propellers. The axial velocity is integrated on each plane and time averaged for one rotation. From this it was observed that the conventional propeller has more suction than the unconventional one in model scale.
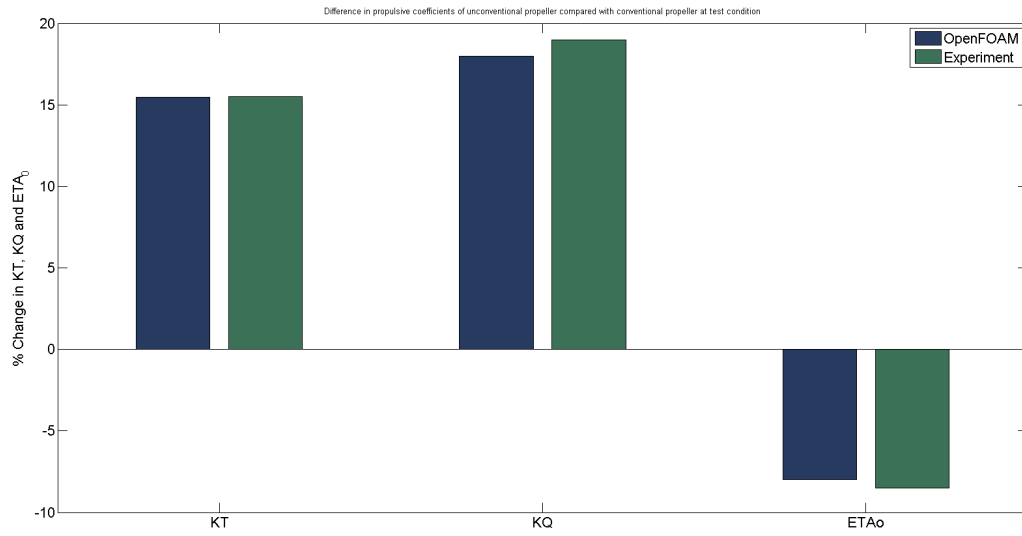
Figure 4.11: Difference in propulsive coefficients of unconventional propeller with conventional propeller at test condition.
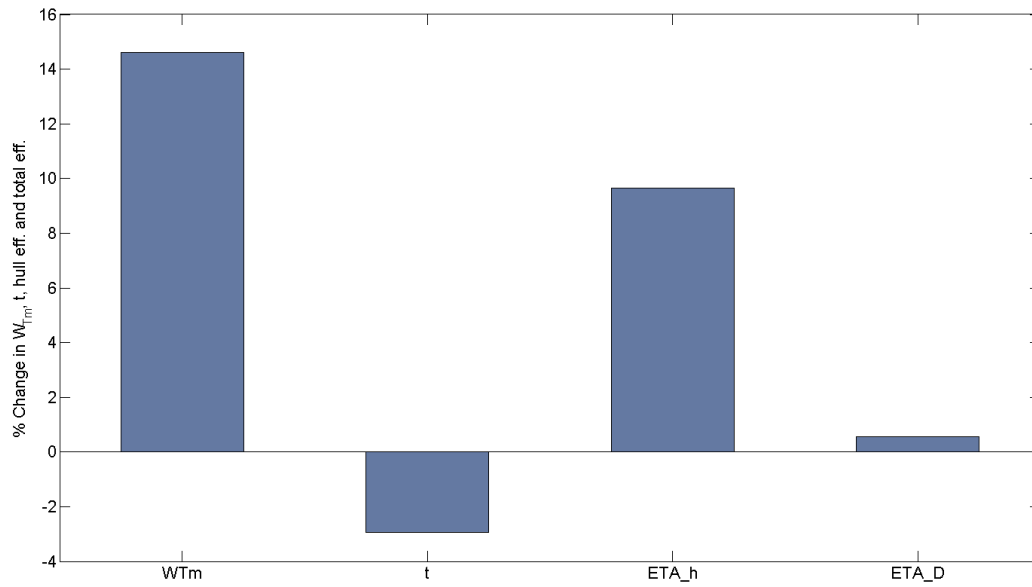
Figure 4.12: Difference in propulsive coefficients of unconventional propeller with conventional propeller at test condition (CFD).
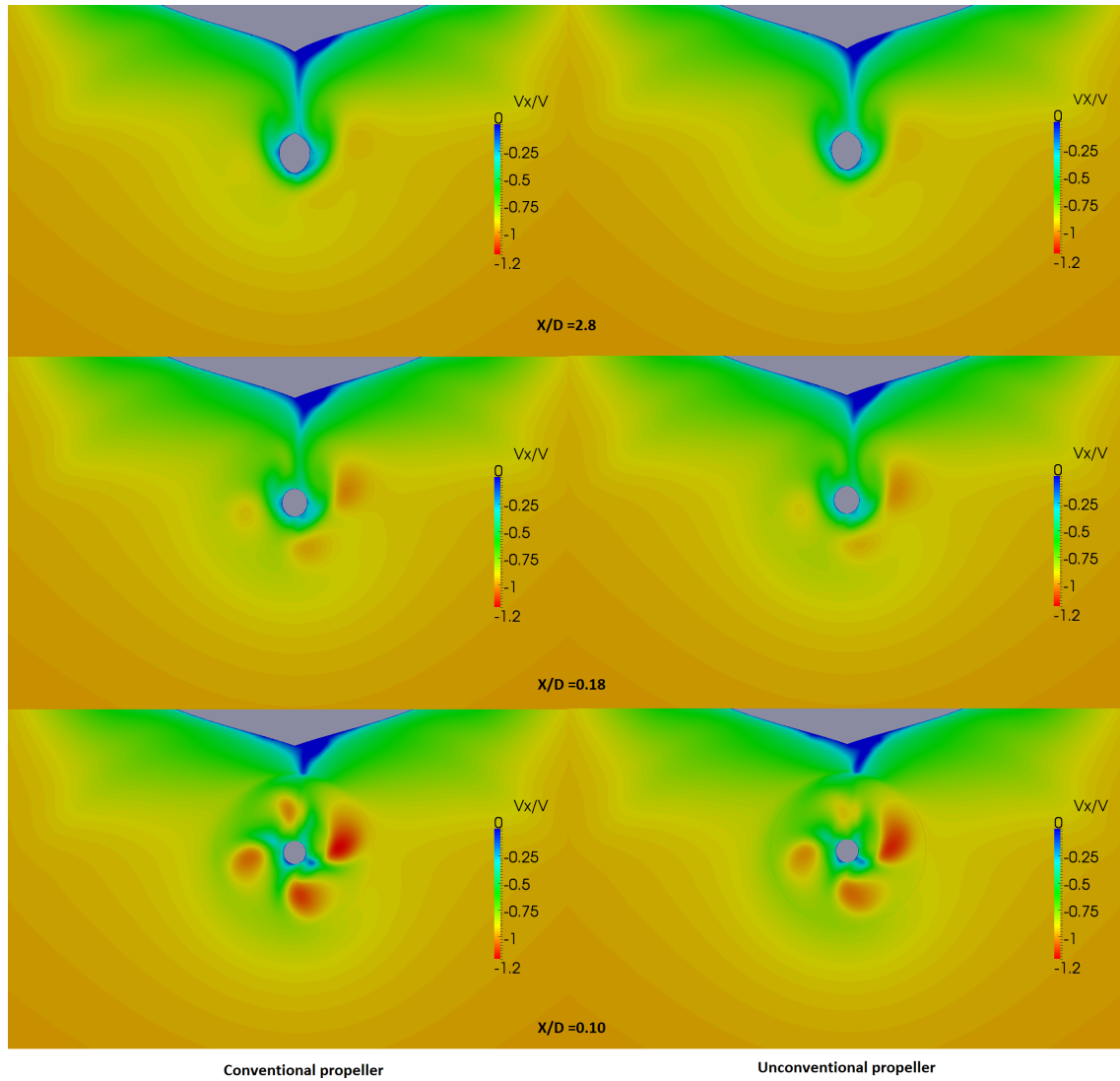
Figure 4.13: Axial velocity distributions at different longitudinal positions starting from upstream of the propellers (conventional-left and unconventional right) in model scale.

4.4   Full-Scale Open-water

In the 1978 ITTC Performance Prediction Method, the rate of revolutions (n_s) and the design advance coefficient (J_{TS}) are obtained from the full scale open water characteristics. Therefore it is necessary to study the Reynolds scaling effects for a full scale propeller operating in open water condition. The full scale effective wake (W_{TS}) is calculated using formula 1.1 and then the load of the full scale propeller is obtained by estimating the $\frac{K_T}{J^2}$ ratio which is given by:

$$\frac{K_T}{J^2} = \frac{S_s}{2D^2} \frac{C_{Ts}}{(1-t)(1-W_{Ts})^2 n_p} \tag{4.4}$$

where S_s is the wetted surface the hull, D is the propeller diameter, n_p is the propeller revolution rate in model scale and C_{Ts} is the total resistance coefficient. With $\frac{K_T}{J^2}$ as input value, the full scale advance ratio (J_{Ts}) and the torque coefficient (K_Q) are determined from the full scale propeller open water characteristics. The full scale open water characteristics are determined using the ITTC-78 open water scaling method. The rate of revolutions for the full scale propeller is obtained by:

$$n_s = \frac{(1-W_{Ts})V_s}{J_{Ts}D} \tag{4.5}$$

The ITTC 78 method is applied to predict the full scale performance through model scale simulations. The design advance coefficient (J_{Ts}) and propeller revolution rate (n_s) are estimated for both conventional and unconventional propellers.

To study the Reynolds scaling effects at the ITTC predicted J_{Ts}, the propeller test cases for full scale are set up. From the experience from the model scale simulations and through the literature surveys, the Multiple Reference Frame (MRF) method has a quite acceptable accuracy in predicting the open-water performance. Thus MRF technique is chosen for full scale open water simulations. The simulation set up in full scale is quite consistant to that of model scale setup. The wall functions are used to simulate the boundary layer as the y+ value for both the propeller test cases (conventional and unconventional) is around 90. The simulations carried out with a design advance coefficient J_{Ts} and propeller revolution rate (n_s), estimated by ITTC 78 method for both propellers. The open water thrust, torque and efficiencies are calculated as in model scale simulations. The percentage change in the scaling corrections of K_T, K_Q and ETA_0 is estimated using both ITTC 78 method and using OpenFOAM RANS simulations.

For both the propellers the ITTC scaled propulsive coefficients didn't show good correlation with the RANS results. The OpenFOAM calculated K_T, K_Q and ETA_0 values are approximately 5-12% higher than the ITTC 78 scaled values. Figures 4.15 and 4.17 show the scaling corrections as a percentage change of K_T, K_Q and ETA_0 at a design advance coefficient (J_{Ts}) estimated by the ITTC 78 method (Equation 4.5). The conventional propeller has a J_{Ts} of 0.486 and unconventional propeller has a J_{Ts} of 0.483. To better understand the differences between model scale and full scale simulations, the integrated pressure and shear stress for thrust and torque are investigated separately. If only viscous shear stresses are considered the model to full scale corrections predicted by RANS simulation reduce greatly (Figures 4.15 and 4.17). This clearly implies that there must be a Reynolds number effect on the blade pressure distributions not captured by the friction coefficient based ITTC 78 method. The viscous-only scaling correction from the RANS simulation is similar to the friction coefficient bases ITTC 78 method for both the cases.

However, the viscous scaling predicted by RANS is greater than ITTC 78 method. This could be due to the fully turbulent boundary layer assumption in the OpenFOAM solver.

Figure 4.18 show the percentage change of various properties of present unconventional propeller with conventional one at a different advance ratios. The full scale simulations of both propellers at the ITTC scaled design advance coefficient(J_{Ts}) show that the unconventional propeller has 2.5% lesser efficiency that the conventional one. The $\frac{K_T}{J^2}$ value is compared between ITTC 78 method and from the RANS simulation. The $\frac{K_T}{J^2}$ value predicted by RANS is 0.2% under-predicted compared with the ITTC predicted $\frac{K_T}{J^2}$ value for a conventional propeller and 7.3% over-predicted by RANS when compared to the ITTC predicted value for the unconventional propeller. Further the amount of thrust produced by the unconventional propeller is 6.8% higher than the conventional one. This comparison questions the design advance coefficient (J_{Ts}) predicted by ITTC 78 method for the unconventional propeller. Since the $\frac{K_T}{J^2}$ value predicted by ITTC 78 and RANS methods are much coherent, it is deduced that the ITTC 78 method works quite well for the conventional propeller. Therefore, it is assumed that the thrust produced by the conventional propeller is the required thrust (T_{req}) for the operating condition. Now for the unconventional propeller, the advance coefficient is adjusted and flow is simulated till the thrust of the propeller reaches T_{req}. From this simulations, a new design advance coefficient (J_{TsRANS}) is predicted for the unconventional propeller at the operating conditions. The RANS predicted design advance coefficient (J_{TsRANS}) has a value of 0.5156 which is 6.6% higher than the design advance coefficient (J_{Ts}) predicted by ITTC 78 method. Figure 4.19 show the scaling corrections as a percentage change of K_T, K_Q and ETA_0 at a design advance coefficient (J_{TsRANS}). At J_{TsRANS}, the unconventional propeller has 7.2% higher efficiencies than conventional one. If the viscous-only scaling corrections applied the unconventional propeller has 3.3% higher efficiency than the conventional propeller. (Note: For conventional propeller $J_{TsRANS} = J_{Ts}$ and for unconventional propeller $J_{Ts} < J_{TsRANS}$.)

The ITTC 78 procedure arrive at a full scale propeller characteristics by correcting the model test results for model to full scale friction differences at a representative radius r/R =0.75. This assumption has been questioned for the unconventional propeller due to unique tip geometry and load distribution [1]. Comparing Figures 4.15 and 4.19, it can be observed that at J_{TsRANS} the OpenFOAM viscous scaling effect on propulsive coefficients for the conventional propeller is less than that of the unconventional propeller. The viscous scaling effect on efficiency is about 2% for conventional propeller and 3.5% for unconventional propeller at J_{TsRANS}. While the ITTC 78 efficiency scaling is less than 2% for both the propellers. This study explains the increase in viscous scaling effects due to the additional surface at the tip of the unconventional propeller.

Further the ITTC 78 procedure over-predict the $\frac{K_T}{J^2}$ value which questions the load calculation formula (Equation 4.4) and the ITTC 78 wake scaling formula (equation 1.1). To analyse this discrepancy there is a necessity to compare the self propulsion tests in model and full scale.
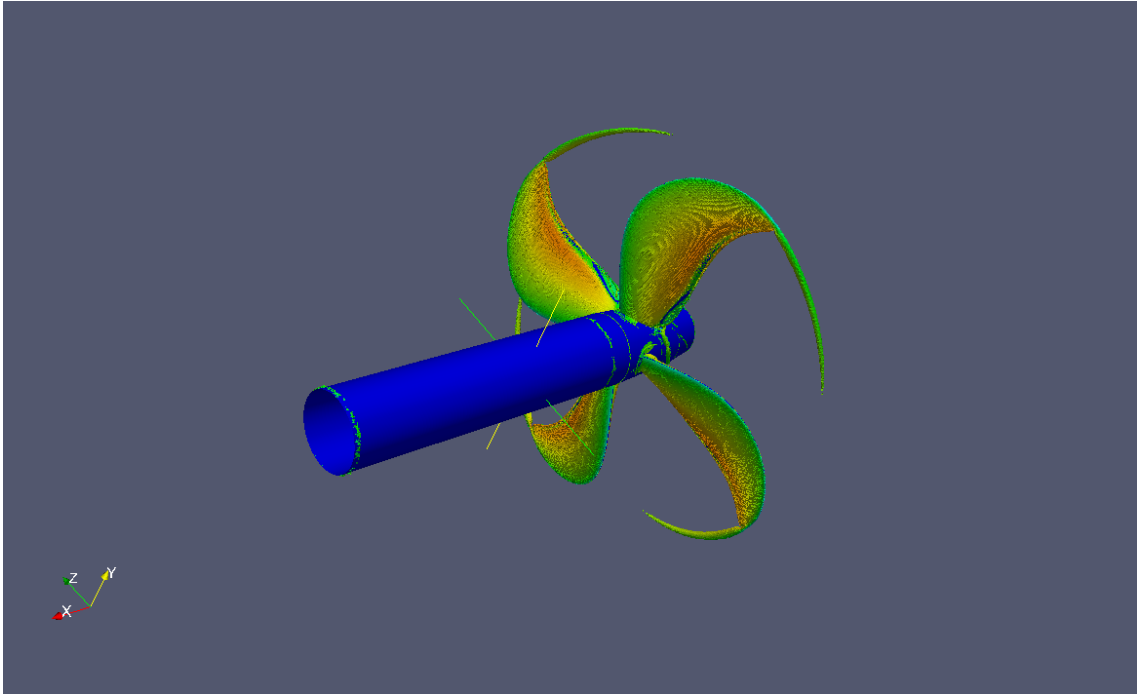
Figure 4.14: Iso-Surface Q coloured by Ux magnitude for convectional propeller (Full-scale)
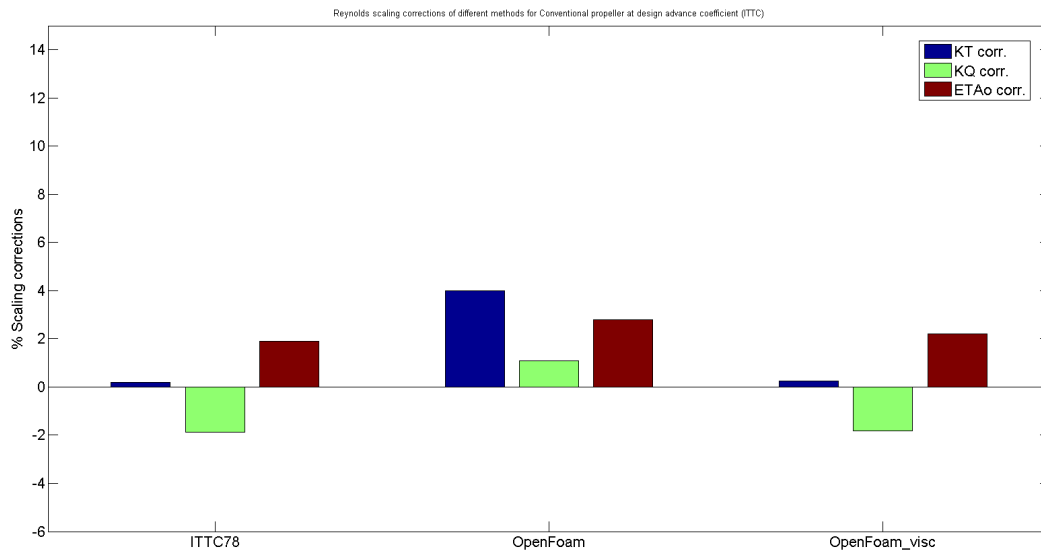
Figure 4.15: Reynolds scaling corrections of different methods for conventional propeller at design advance coefficient (ITTC)
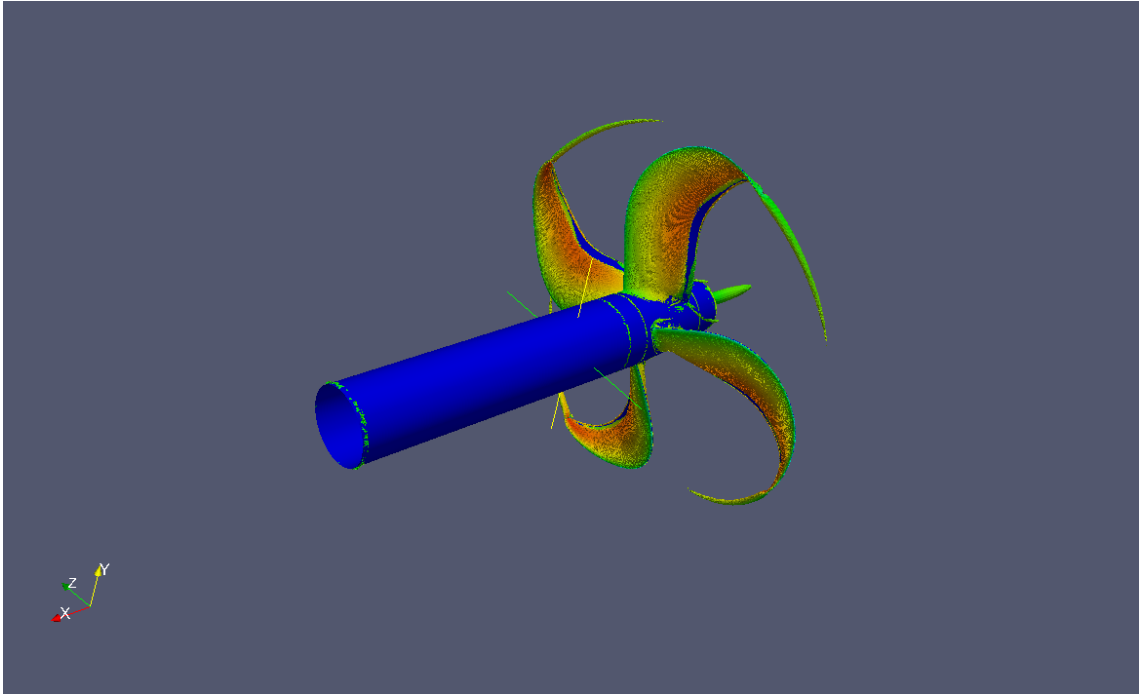
Figure 4.16: Iso-Surface Q coloured by Ux magnitude for unconventional propeller
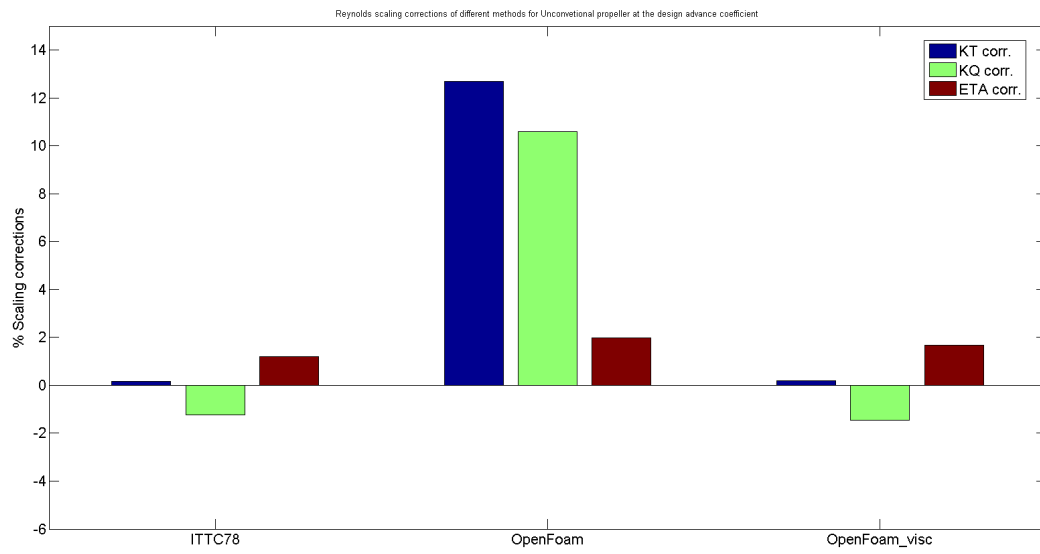
Figure 4.17: Reynolds scaling corrections of different methods for unconventional propeller at design advance coefficient (ITTC)
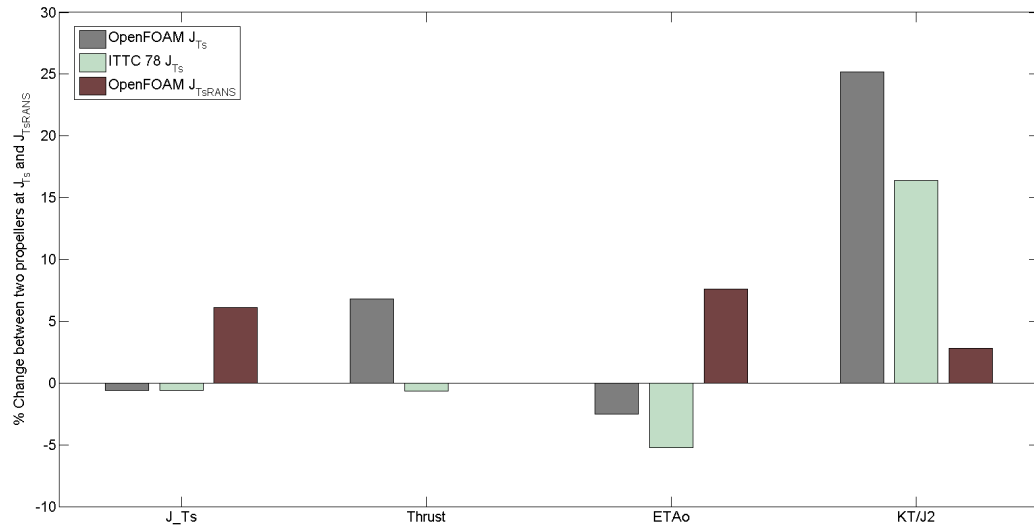
Figure 4.18: Difference in properties of present unconventional propeller with conventional propeller at J_{Ts} and J_{TsRANS}
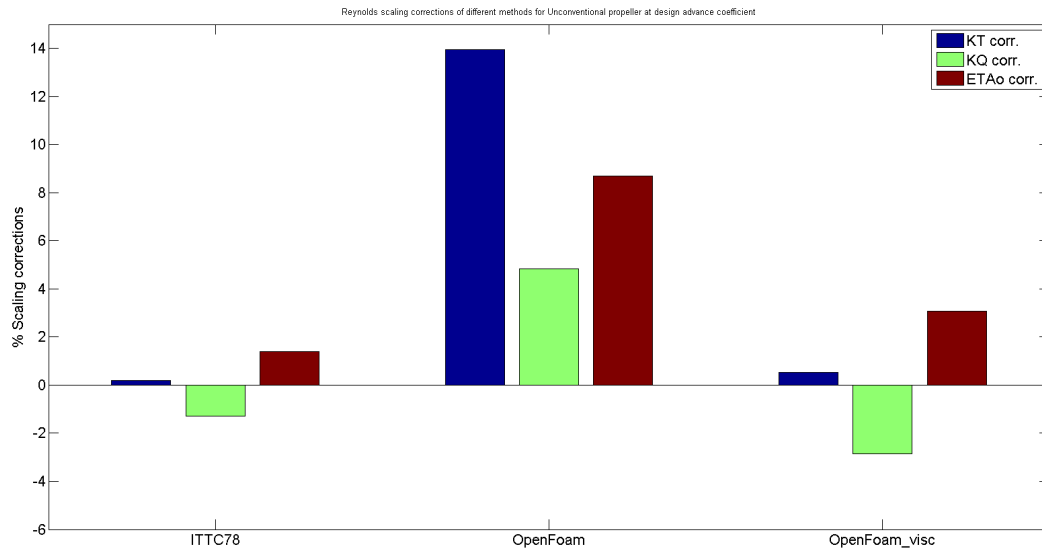
Figure 4.19: Reynolds scaling corrections of different methods for unconventional propeller at design advance coefficient J_{TsRANS}

4.5 Full-Scale Self Propulsion

In this section, the performance of the two propellers operating behind a hull is evaluated in full scale. The flow around the propellers is simulated using RANS method with translational grids (AMI) using the pimpleDyMFoam solver. The simulation set up in full scale is quite similar to that of model scale setup. The wall functions are used to simulate the boundary layer as the y+ value for both the test cases (hull with conventional and unconventional propeller) is around 150. The flow is simulated at the same Froude number as the model scale ($F_n = 0.153$) and with the propeller revolution rate (n_s) estimated by the ITTC 78 performance prediction method. Again, there is no consideration of free surface in the simulation. The ranking between the two propellers is compared with the predictions of the ITTC 78 method. For estimating the effective power of the hull without free surface, the flow is simulated around the bare hull without the propeller.
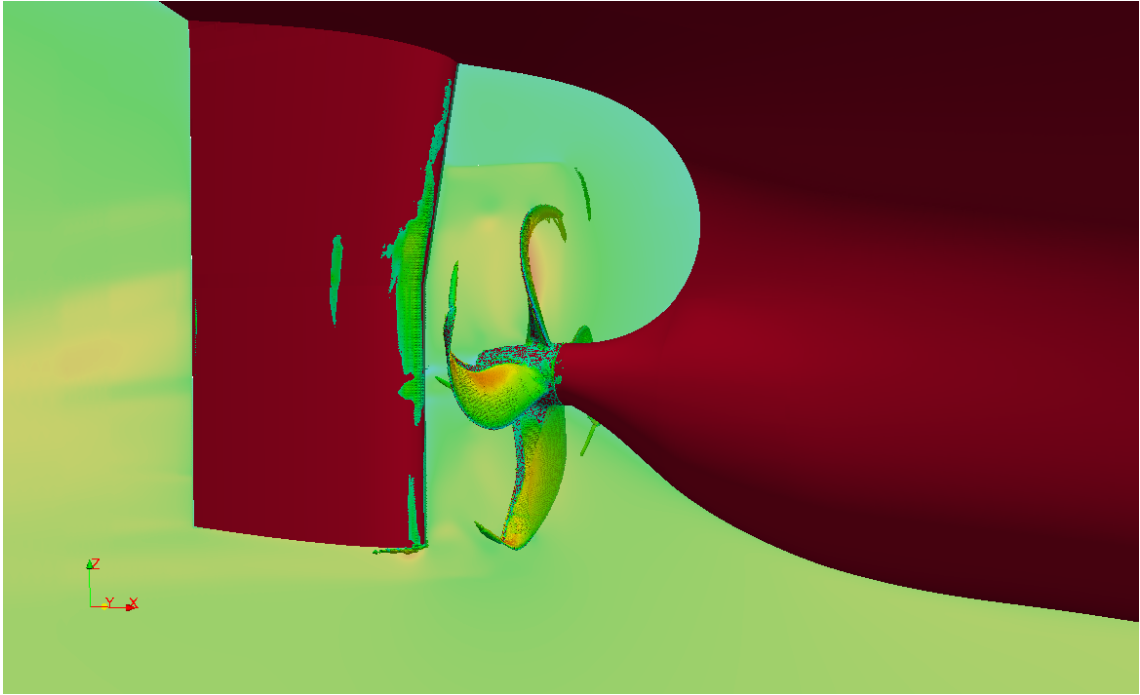
Figure 4.20: Iso-Q plot and axial velocity variation coloured by U_x magnitude for a full scale unconventional propeller behind the hull

In Figure 4.21 the percentage change of various propulsive factors (K_T, K_Q, ETA_0 and K_T/J^2) between both the propellers is compared with the ITTC 78 full scale design condition (i.e same ship speed and rps of the propeller). Figure 4.22 shows the difference in the effective wake fraction (W_{Ts}), thrust deduction (t), hull efficiency (ETA_h) and total efficiency (ETA_D) respectively. The prediction of the RANS forecasted the condition of two test cases to be under propelled. At this under propelled condition the full scale advance ratios (J_{Ts}) of conventional and unconventional propeller are 0.4931 and 0.5335 respectively. However, the ITTC 78 estimates J_{Ts} for conventional propeller to be 0.486 and 0.482 for unconventional propeller.

The RANS assessment at the full scale design condition showed that the present unconventional propeller obtains lower effective wake and hull efficiency than the conventional one. It also estimates that the present unconventional propeller has a higher propeller and

propulsive efficiency than the conventional propeller. However, the ITTC 78 performance prediction method estimates that the present conventional propeller has a lower effective wake and hull efficiency than the unconventional propeller. It also estimates that the propeller and propulsive efficiencies are higher for the present conventional compared to the unconventional propeller.
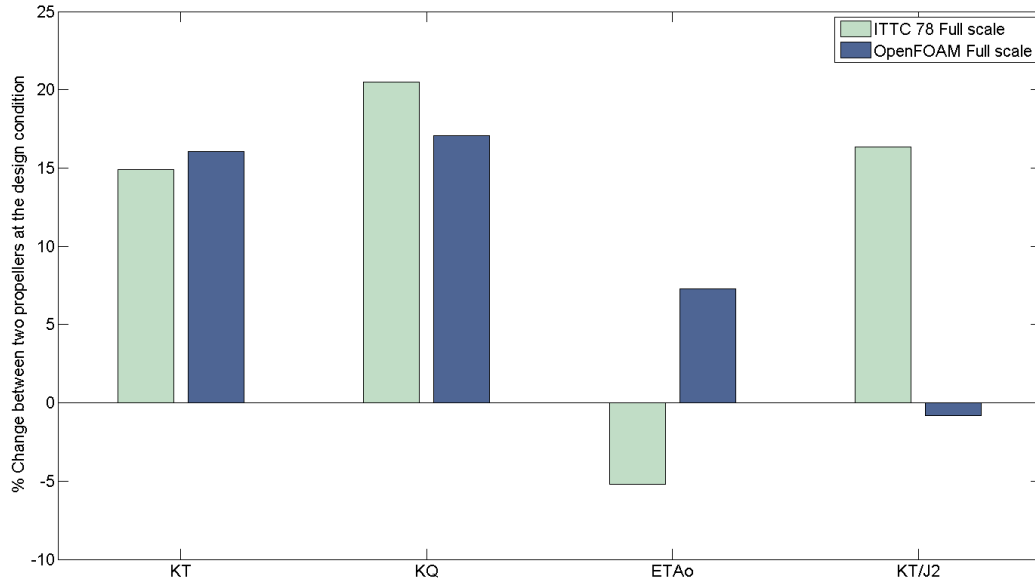
Figure 4.21: Difference in propulsive coefficients of unconventional propeller with conventional propeller using ITTC 78 and OpenFOAM
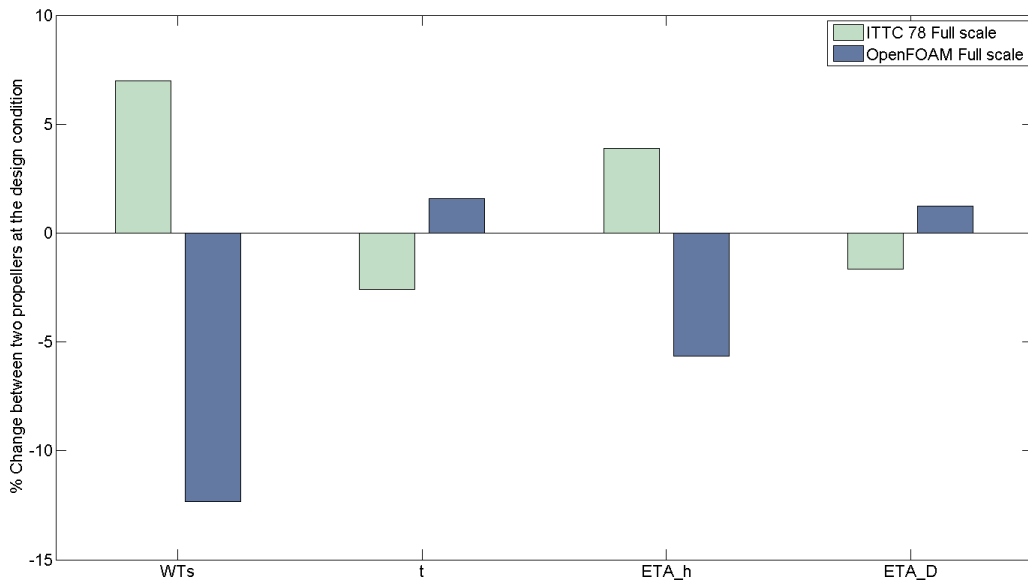
Figure 4.22: Difference in propulsive coefficients of unconventional propeller with conventional propeller using ITTC 78 and OpenFOAM

Figure 4.23 shows a series of plots of axial velocities at different longitudinal positions at non dimensional distance (X/D) 0.28, 0.18, 0.10 respectively in upstream of the propellers. The axial velocity is integrated on each plane and time averaged for one rotation. From this it was observed that the unconventional propeller has more suction than the conventional one in full scale.
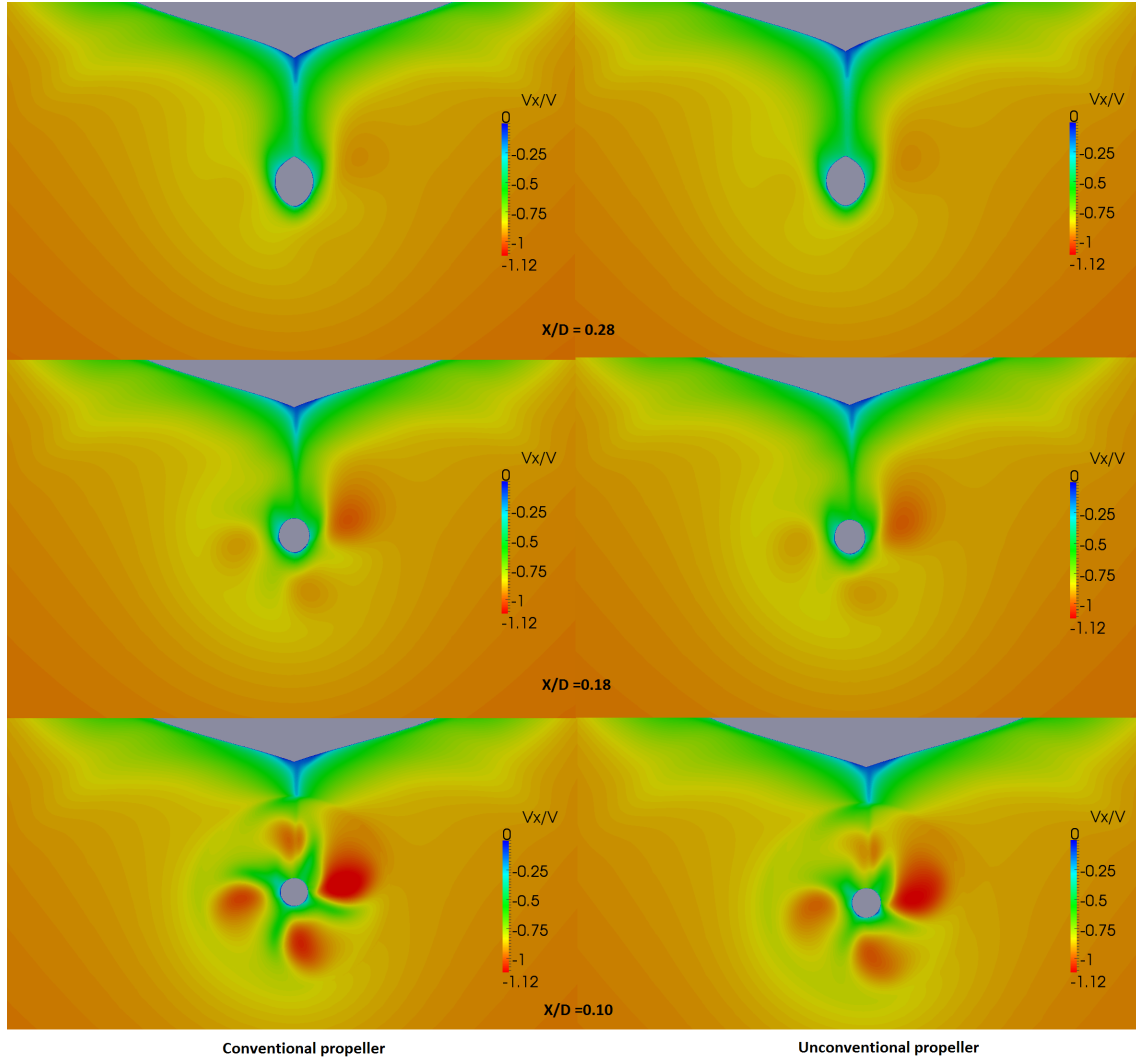
Figure 4.23: Axial velocity distributions at different longitudinal positions starting from upstream of the propellers (conventional-left and unconventional-right) in full scale.

Chapter 5

Summary and Conclusions

The ITTC 1978 performance prediction method has been a universal tool for assessing the performance of the full scale vessels with conventional propellers. In general, this method works well for conventional propeller designs and has been in implementation for more than 30 years. However, due to the introduction of new unconventional propeller designs like the Kappel propeller, question this method particularly with wake scaling. The goal of this thesis is to establish a model and full scale prognosis on the unconventional and conventional propellers using CFD and to compare it with the ITTC 78 method. Further, the areas of further research are identified which can justify the discrepancies between the two methods.

The results presented in this report show how the OpenFOAM RANS libraries are applied to simulate the main features of flow around the hull and the propeller. The dicretization mesh is developed using the open source `snappyHexMesh` utility. The open source tools could generate meshes with a decent quality but not in an efficient way and it also lacked many important features. However, there is a definite potential for developing an efficient open-source mesh generation approach in order to reduce the computational costs. The results from the open-source CFD correlated well with the available experimental data for the test cases and the differences between the results have been discussed. The present CFD setup using open-source CFD also showed a good agreement when compared with commercial CFD tools like Fluent (Bare hull case).

The scale effects have been investigated for present conventional and the unconventional propeller using both ITTC 78 method and OpenFOAM RANS. The pressure and viscous effects are studied separately to better understand the scaling effects. If only the changes in viscous stresses are considered the model to full scale corrections correlate well with the ITTC 78 corrections. This justifies the friction coefficient based ITTC 78 method. Further this study show that there must be a Reynolds number effect on blade pressure distributions which is not taken into account by the ITTC 78 method.

The ITTC 78 procedure estimates the full scale propeller characteristics by correcting the model test results for model to full scale friction differences at a representative radius r/R =0.75. This assumption has been questioned for the unconventional propeller due to unique tip geometry and load distribution [1]. In support of this, the RANS scaling also showed an addition scaling effect on the unconventional propeller compared with the conventional propeller. The unconventional propeller showed higher propeller efficiency and lower delivered power at RANS predicted design advance coefficient (J_{TsRANS}).

The performance of the present conventional and unconventional propeller behind a hull is evaluated using RANS and the ITTC 78 method at same ship speed (V_s) and propeller revolution rate (n_s). The results from the RANS didn't correlate well with the ITTC 78 method for unconventional propeller. The RANS analysis predict that the present unconventional propeller obtains lower effective wake that the conventional one (about 12%) at this particular design condition in contradiction to the ITTC 78 method. The full scale RANS investigation on various efficiencies of the propeller showed that the present unconventional propeller has higher propeller efficiency (about 7.3%) and total efficiency (about 1.23%) compared to the conventional propeller contradicting the ITTC 78 method. However, the hull efficiency is higher for the present conventional propeller (about 5%).

The Reynolds number effect on blade pressure distributions of unconventional propeller necessitated further investigation on full scale open-water performance. From the RANS assessment of the propellers in open-water and behind the hull conditions showed that the ITTC 78 over predicts the load on the propeller (K_T/J^2) which could be due to the discrepancies in wake scaling. Further it was observed that at the design condition the unconventional propeller obtains much higher drag (about 14%) compared to conventional propeller in model scale. In full scale, this difference was smaller (about 6%) and this additional drag in model scale could affect the scaling of the propeller operating in a wake field. However, all these constraints needed a further investigation. The wave making wake is yet an another important factor which affects the effective wake which have not been investigated in this thesis.

Chapter 6

Future work

The RANS study contradict the predictions of the ITTC 78 method for the unconventional propeller. The results presented in this report are however not sufficient to make clear and precise conclusions from the differences and findings identified in the ITTC 78 method. As mentioned in the previous chapter, there are additional influences which are not investigated in this project. These influences must be studied in the future work. However, just focusing on RANS analysis using one particular tool and its investigation on a single unconventional propeller test case, do not complete the story. Although the scaling effects assessed on different kind of unconventional propeller designs operating in different flow regimes would have a substantial effect on the present research. The areas of future work that are to be focused are:

- Implementing other modelling techniques: The turbulence models implemented in the test cases assume the boundary layer to be fully turbulent which may sometimes over predict the viscous stresses. However, the implementation of turbulence models such as $kkl - \omega$ could improve the laminar-turbulent transition predictions. Implementing non linear turbulence models like Algebraic Reynolds Stress model could improve the nominal wake predictions. In most of the test cases, the wall functions are used in order to calculate the velocity profiles near the wall. However, most reliable way is to resolve the viscous sub layer which is expensive. Therefore, the effect of wall functions have to be studied specifically for full scale test cases.

- Comparison of ITTC 78 open-water scaling with other methods: The ITTC 78 procedure uses a similarity assumption that reduces the span wise quantities to a representative radius of 0.75 r/R. For Kappel and other unconventional designs a significant part is situated after 0.75 r/R. The methods such as sistemar [4] and Anderson strip wise blade friction calculation method [2] consider this effect in to account. Therefore, it is advised that a pedagogical comparison of these methods along with the ITTC 78 methods and CFD results will be of a great importance in order to study the differences with ITTC 78 method.

- Effect of free surface: The free surface will contribute to the wave making wake which will have a potential impact on effective wake. This effect is not taken in to account in this thesis due to complexities in open-source meshing and the time frame of the project. However, the free surface will have a minor effect in the comparative study.

- Ranking between the propellers at self-propulsion point in model and full scale: The comparison study at the self propulsion point would be a supreme way to investigate

the discrepancies as it can predict the ideal propeller revolution rates. This can also suggest an ideal towing force in order to counter the additional drag on the Kappel propeller in model scale.

- Exploring the science of ITTC 78 wake scaling formula: Better understanding of ITTC formulation may identify various constraints affecting the scaling procedure.

- Mesh studies: Mesh studies must be performed in full scale in order to validate the full scale results in a much better way and the effect of wall functions in the solution has to be studied. Further, an efficient mesh generation strategy must be implemented in snappyHexMesh in order to reduce the computational time. There are other alternatives for the open-source meshing such as CFMesh, Salome etc. which can be implemented for constructing better quality meshes.

- Comparison of CFD evaluation with the sea trails: Comparing the performance of the propeller evaluated in full scale using CFD with the service performance of the ships could validate and justify the solution methods used in the simulation. It also helps in identifying various influencing factors in the performance.

- Effect of leading edge separation on Kappel propller at higher advance ratios: It is observed that (Figure 4.8) the RANS analysis predict a small area of separation. The effect of this separation propulsive coefficients is to be further investigated using high fidelity methods if the design advance coefficient exceeds the value 1.

References

[1] Nielsen, J.R., Kappel, J.J., Bering, R.M. (2014) Scaling of conventional and unconventional propellers. ITTC 2014 conference. Copenhagen.

[2] Andersen, P. (1996) A Comparative Study of Conventional and Tip-Fin Propeller performance. 21st Symposium on Naval Hydrodynamics, Trondheim, Norway.

[3] Andersen, P., Friesch, J., Kappel, J.J. (2005) Development of a Marine Propeller with Nonplanar Lifting Surfaces. Marine Technology 42(3) pp.144-158.

[4] Brown, M. , Sanchez-Caja, A., Adalid, JG., Black, S., Perez Sobrino, M., Duerr, P., Schroeder, S., Saisto, S. (2014) Improving Propeller Efficiency Through Tip Loading. 30th Symposium on Naval Hydrodynamics, Hobart, Tasmania, Australia.

[5] Sanchez-Caja, A. Sipila, T.P., Pylkkanen, J.V. (2006) Simulation of the Incompressible Viscous Flow around and Endplate Propeller using a RANSE Solver. 26th Symposium on Naval Hydrodynamics, Rome, Italy.

[6] Cheng, H., Chien, Y., Hsin, C., Chang, K., Chen, P. (2010) A numerical comparison of end-plate effect propellers and conventional propellers. 9th International Conference on Hydrodynamics, Shanghai, China.

[7] Bertetta, D., Brizzolara, S. (2012) EFD and CFD Characterization of a CLT Propeller. International Journal of Rotating Machinery, Volume 2012, Article ID 348939.

[8] Gaggero, S. and Brizzolara, S. (2012) Endplate effect propellers: A numerical overview. Sustainable Maritime Transportation and Exploitation of Sea Resources–Rizzuto & Guedes Soares (eds), Taylor & Francis Group, London, ISBN 978-0-415-62081-9.

[9] Andersen, P., Kappel, J.J., (2009) Aspects of Propeller Developments for a Submarine. First International Symposium on Marine Propulsors (smp'09), Trondheim, Norway.

[10] Järpener, C., (2011) Projection of a mesh on a .stl surface. "http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2011/ChristofferJarpner /ReportInOpenFOAMbyChristofferJarpner.pdf2 "

[11] Gonzalez-Adalid, J., Perez Sobrino, M., et-al (2014) Comparison of different scaling methods for model tests with CLT propellers. 11th International conference on hydrodynamics (ICHD 2014), Singapore.

[12] Streckwall, H., Greitch, L., Muller, J., Scharf, M., Bugalski, T. (2013) Development of a strip method proposed to serve as a new standard for propeller performance scaling. Ship Technology Research.

[13] Dyne, G., (2005) On the principles of propellers with end-plates. International Journal of Maritime Engineering.

[14] Andersen, P., Borrod, AS., Blanchot, H. (2004) Evaluation of the service performance of ships. Marine Technology 42(4) pp.177-183.

[15] Lindgren, H., Dyne, G., (1979) Ship performance prediction. International symposium on advances in Marine Technology, Trondheim, Norway.

[16] OpenFOAM Foundation (2014), OpenFOAM user guide, downloaded: (19-Jan-2015) "http://foam.sourceforge.net/docs/Guides-a4/UserGuide.pdf"

[17] Turnen, T., Siikonen, T., Lundberg, J., Bensow, R. (2014) Open-Water Computations of a Marine Propeller using OpenFOAM, Barcelona, Spain.

[18] Dyne, G. and Bark, G., (2005) Ship Propulsion, course compendium of the Msc course, Division of Hydrodynamics, Dept. of Shipping and marine technology, Chalmers University of Technology, Göteborg, Sweden.

[19] Davidson, L., (2014) Fluid mechanics and Turbulence modelling, course material in Msc course, Division of Fluid Dynamics, Dept. of Applied Mechanics, Chalmers University of Technology, Göteborg, Sweden.

[20] Nilsson, H., (2014) Course material from "MSc/PhD course in CFD with OpenSource software", http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2014/

Appendix A

ITTC 78 Methods

A.1 Resistance tests

During the resistance test the model is towed at speeds giving the same Froude numbers (F_n) as for the full scale ship. The total model resistance (R_{Tm}) is measured.

The conversion from model (m) to ship (s) is made according to the 1978 ITTC Performance Prediction Method. This implies that the frictional resistance coefficient (C_F) is calculated from the 1957 model-ship correlation line, giving the relation between C_F and Reynolds number (R_n):

$$C_F = \frac{0.075}{(log_{10}R_n - 2)^2}$$

$$R_n = \frac{VL}{\nu}$$

$$F_n = \frac{V}{\sqrt{gL}}$$

where,
V= speed, in m/s
L = length of the waterline for ship and model respectively, in m
ν = kinematic viscosity. in m^2/s (ITTC 1960)
g = acceleration due to gravity, in m/s^2

It is further assumed that the form factor (k) based upon ITTC 1957 and the residuary resistance coefficient (C_R) are identical for the model and the ship at the same F_n. If ΔC_F is the roughness allowance coefficient and C_{AA} is the air resistance coefficient, the total ship resistance (R_{Ts}) can be calculated from:

$$R_{Ts} = C_{Ts}\frac{1}{2}V_s^2 S_s 10^{-3}$$

$$C_{Ts} = \frac{S_s + S_{Bk}}{S_s}\Big[(1+k)C_{Fs} + \Delta C_F\Big] + C_R + C_{AA}$$

$$C_{Rm} = C_{Tm} - (1+k)C_{Fm} = C_{Rs}$$

$$C_{Tm} = \frac{R_{Tm}}{\frac{1}{2}V_s^2 S_m}$$

where,

C_{Tm} and C_{Ts} = the total resistance coefficient in model and ship scale respectively.

R_{Tm} and R_{Ts} = total resistance in model and ship scale respectively.

ρ_m and ρ_s = mass density of the tank water and sea water respectively, in Ns^2/m^4

V_m and V_s = ship speed in model and ship scale respectively.

S_m and S_s = wetted surface in m^2 in model and ship scale respectively.

S_{Bk} = wetted surface in m^2 of ship bilge keels

The ship model is not in general fitted with bilge keels.

The form factor (k) showing how large the viscous resistance of model and ship is compared to the ITTC 1957 correlation line is determined from the resistance tests in the low speed range, where the wave resistance is small.

The roughness allowance coefficient (ΔC_F) is assumed to be:

$$\Delta C_F = \left[105 \left(\frac{k_s}{L} \right)^{\frac{1}{3}} - 0.64 \right] 10^{-3}$$

with hull roughness $k_s = 150 \mu m$

The air resistance coefficient (C_{AA}) is assumed to be:

$$C_{AA} = 0.001 \frac{A_T}{S_s}$$

where A_T =transverse projected area above the water of the ship including superstructures in m^2

Finally, the ship's effective power (P_E) is calculated from:

$$P_E = V_s R_{Ts} 10^{-3}, \text{ in MW}$$

A.2   Self-propulsion tests

The ship model is equipped with propelling machinery and working propeller(-s). In the self propulsion test the model is towed at speeds giving the same Froude numbers (Fn) as for the full scale ship. The propeller rate of revolutions is in general adjusted so that the towing force (Ra) will reach the value:

$$R_a = \frac{1}{2}\rho_m S_m V_m^2 [C_{Fm} - (C_{Fs} + \Delta C_F)]$$

where,
R_a = towing force, in N
ρ_m = mass density of tank water, in Ns^2/m^4
S_m = wetted surface, in m^2
V_m = speed, in m/s
C_{Fm} and C_{Fs} = frictional resistance coefficients (ITTC 1957)
ΔC_F = friction correction dependent on surface condition of the ship hull (Normally ΔC_F = 0.0004)

During the tests propeller thrust (T), torque (Q) and rate of revolutions (n) are measured. The thrust (T) and torque (Q) are expressed in non-dimensional form as:

$$K_{Tm} = \frac{\sum T}{\rho_m D_m^4 n^2 n_p},$$

$$K_{Qm} = \frac{\sum Q}{\rho_m D_m^5 n^2 n_p},$$

where D = diameter of the propeller and n_p = number of propellers.
with K_{Tm} as input data the advance ratio (J_{Tm}) the torque coefficient (K_{QTm}) are derieved from the open-water characteristics of the model propeller. The wake fraction ($W_T m$) and the relative rotative efficiency (η_R) are calculated from:

$$W_{Tm} = 1 - \frac{J_{Tm} Dn}{V},$$

and

$$\eta_R = \frac{K_{QTm}}{K_{Qm}}.$$

The thrust deduction fraction (t) is obtained from:

$$t = \frac{\sum T + R_a - R}{\sum T},$$

where R is the measured model resistance R_{Tm} (see A.1) corrected for the difference in temperature between resistance and self-propulsion tests.

The 1978 ITTC Performance Prediction Method gives the rate of revolutions (n_s) and delivered power (P_{Ds}) for the ship (s) obtained from the full scale propeller open water characteristics. These characteristics are determined by correcting the model (m) values for drag scale effects (see A.3).

The full scale wake (W_{Ts}) is calculated from the model wake (W_{Tm}) and the thrust deduction factor according to:

$$W_{Ts} = (t + 0.04) + (W_{Tm} - t - 0.04)\frac{(1+k)C_{Fs} + \Delta C_F}{(1+k)C_{Fm}}$$

where
k = form factor (see A.1)
ΔC_F = roughness allowance coefficient (see A.1)
if $W_{Ts} > W_{Tm}$, W_{Ts} will be set equal to W_{Tm}

The load of the full scale propeller is obtained from:

$$\frac{K_T}{J^2} = \frac{S_s}{2D^2}\frac{C_{Ts}}{(1-t)(1-W_{Ts})^2 n_p}$$

where,
S_s = wetted surface of the hull, in m^2
D = propeller diameter, in m
C_{Ts} = total resistance coefficient

With K_T/J^2 as the input value, the full scale advance ratio (J_{Ts}) and the torque coefficient (K_{QTs}) are determined from the full scale propeller open water characteristics and the following quantities are calculated.
The rate of revolutions:

$$n_s = \frac{(1-W_{Ts})V_s}{J_{Ts}D}$$

The delivered power:

$$P_{Ds} = 2\pi\rho D^5 n_s^3 n_p \frac{K_{QTs}}{\eta_R}10^{-6}MW$$

where ρ is the mass density of the sea water, Ns^2/m^4

The thrust of the propeller:

$$T_s = \frac{K_T}{J^2}J_{Ts}^2\rho D^4 n_s^2 10^{-3}kN$$

The torque of the propeller:

$$Q_s = \frac{K_{QTs}}{\eta_R}\rho D^5 n_s^2 10^{-3}kNm$$

The total efficiency:

$$\eta_D = \frac{P_E}{P_{Ds}}$$

P_E is the ship's effective power, in MW (see A.1)

The propeller efficiency:

$$\eta_0 = \frac{K_T}{K_Q}\frac{J_{Ts}}{2\pi}$$

K_T as well as K_Q are valid for J = J_{Ts}

The hull efficiency:

$$\eta_H = \frac{1-t}{1-W_{Ts}}$$

A.3   Propeller open water tests

The propeller model is mounted on a horizontal shaft and is moved through the water at an immersion of the shaft centre equal to the propeller diameter, if nothing else is mentioned.

Thrust (T), torque (Q) and rate of revolutions (n) are measured on the shaft behind the propeller model. The normal test method is to keep the rate of revolutions constant whilst the speed of advance (V_A) is varied so that a loading range of the propeller is examined. The thrust (T) and torque (Q) measured during the tests are converted into non-dimensional coefficients (K_{Tm} & K_{Qm} respectively) which are defined as:

$$K_{Tm} = \frac{T}{\rho_m D^4 n^2}$$

$$K_{Qm} = \frac{Q}{\rho_m D^5 n^2}$$

where D = diameter of the propeller, n = propeller revolution rate and ρ is the mass density of tank water.

The propeller efficiency η_0 and the advance ratio (J) are calculated according to:

$$\eta_0 = \frac{J K_T}{2\pi K_Q}$$

where

$$J = \frac{V_A}{nD}$$

and V_A is the speed of advance.

Notice that all the quantities and coefficients mentioned above refer to the propeller model. These propeller open water characteristics are used to calculate the mean wake fraction (W_{Tm}) and the relative rotative efficiency (η_{Rm}) from the self propulsion tests.

The 1978 ITTC Performance Prediction Method gives rate of revolutions and delivered power for the ship (s) obtained from the full scale propeller open water characteristics. These characteristics are determined by correcting the model (m) values for drag scale effects according to the following:

$$K_{Ts} = K_{Tm} + \Delta K_T$$

$$K_{Qs} = K_{Qm} + \Delta K_Q$$

where

$$\Delta K_T = \Delta C_D 0.3 \frac{P}{D} \frac{cZ}{D}$$

$$\Delta K_Q = -\Delta C_D 0.25 \frac{cZ}{D}$$

The difference in blade drag coefficient is $\Delta C_D = C_{Dm} - C_{Ds}$, where

$$C_{Dm} = 2(1 + \frac{t}{c})\left[\frac{0.044}{(R_{nco})} - \frac{5}{(R_{nco})^{\frac{2}{3}}}\right]$$

$$C_{Ds} = 2(1 + \frac{t}{c})\left[1.89 + 1.62log_{10}\frac{c}{k_p}\right]^{-2.5}$$

In these formulas c is the cord length of the blades, t is the maximum blade thickness, P/D is the pitch ratio and R_{nco} is the local Reynolds number at radius $0.75\frac{D}{2}$.

$$R_{nco} = \frac{cV}{\nu}$$

where

$$V = V_A\sqrt{1 + \left(\frac{\pi 0.75}{J}\right)^2}$$

and ν = kinematic viscosity, in m^2/s

Z = number of propeller blades

k_p = full scale blade roughness, which is set to 30 μm

Appendix B

Useful snappyHexMesh settings

```
*******************************************\
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      snappyHexMeshDict;

}
*******************************************\
castellatedMeshControls
{

    // Refinement parameters
    // ~~~~~~~~~~~~~~~~~~~~


    minRefinementCells 10;


    maxLoadUnbalance 0;


    nCellsBetweenLevels 2;

  // Resolve sharp angles
    resolveFeatureAngle 30;

    allowFreeStandingZoneFaces false;

}

snapControls
{

    nSmoothPatch 2;
```

```
    tolerance 1;

    //- Number of mesh displacement relaxation iterations.
    nSolveIter 300; //100;

    //- Maximum number of snapping relaxation iterations.
    //Should stop before upon reaching a correct mesh.
    nRelaxIter 5;

    // Feature snapping

        //- Number of feature edge snapping iterations.
        //  Leave out altogether to disable.
        nFeatureSnapIter 30;


        implicitFeatureSnap false;


        explicitFeatureSnap true;


        multiRegionFeatureSnap false;
}

// Settings for the layer addition.
addLayersControls
{
nGrow 0;

    // Advanced settings


    featureAngle 150;


    slipFeatureAngle 30;


    nRelaxIter 10;

    // Number of smoothing iterations of surface normals
    nSmoothSurfaceNormals 1;


    nSmoothNormals 3;
```

```
        // Smooth layer thickness over surface patches
        nSmoothThickness 1;  //2;  //5;

        // Stop layer growth on highly warped cells
        maxFaceThicknessRatio 2;  //0.8;  //0.5;

        // Reduce layer growth where ratio thickness to medial
        // distance is large
        maxThicknessToMedialRatio 2;  //0.9;  //0.3;

        // Angle used to pick up medial axis points

        minMedianAxisAngle 90;


        // Create buffer region for new layer terminations
        nBufferCellsNoExtrude 0;



        nLayerIter 50;

        nRelaxedIter 20;
}

meshQualityControls
{

 maxNonOrtho 65;


maxBoundarySkewness 10;  //20;
maxInternalSkewness 2;  //4;


maxConcave 150;


minVol 1e-30;  //1e-13;


minTetQuality 1e-30;

//- Minimum face area. Set to <0 to disable.
minArea -1;


minTwist 0.01;
```

```
minDeterminant  0.001;


//   mesh)  (0 –> 0.5)
minFaceWeight  0.05;

//– Volume  ratio  of  neighbouring  cells  (0 –> 1)
minVolRatio  0.01;

/
minTriangleTwist  −1;



.
//minVolCollapseRatio  0.1;


//–  minFaceWeight  (0 –> 0.5)
minFaceWeight  0.02;

relaxed
{
 maxNonOrtho  65;
// maxBoundaryScewness  10;
// maxInternalScewness  2;
 minTetQuality  1e−30;
 minVol  1e−13;
}

// Advanced

    //– Number  of  error  distribution  iterations
    nSmoothScale  4;
    //– amount  to  scale  back  displacement  at  error  points
    errorReduction  0.75;
}

// Write  flags
writeFlags
(
    layerFields      // write  volScalarField  for  layer  coverage
);

mergeTolerance  1e−6;
```

Appendix C

Open-water simulations (model and full scale)

Table C.1: Open-water model scale Conventional propeller

J	K_T	K_Q	ETA
0.504	0.14562	0.01832	0.6362
0.4032	0.185	0.02183	0.5487
0.3025	0.2218	0.02475	0.427

Table C.2: Open-water model scale Kappel propeller

J	K_T	K_Q	ETA
0.555	0.1474	0.01978	0.6562
0.444	0.1907	0.02351	0.5712
0.333	0.2292	0.02660	0.4567

Table C.3: Open-water Full scale

Propeller	J	K_T	K_Q	ETA
conv. J_{Ts}	0.486	0.1608	0.019368	0.642347
kappel. J_{Ts}	0.483	0.1988	0.024407	0.626281
kappel. J_{TsRANS}	0.5156	0.1860	0.02209	0.691254

Table C.4: Open-water Full scale (viscous only corrections)

Propeller	J	K_T	K_Q	ETA
conv. J_{Ts}	0.486	0.1551	0.018816	0.6376
kappel. J_{Ts}	0.483	0.1766	0.02174	0.6242
kappel. J_{TsRANS}	0.5156	0.1639	0.02046	0.6571

*Note: For conventional propeller $J_{TsRANS}=J_{Ts}$

Appendix D

Propeller behind a hull (model and full scale)

Table D.1: Propeller behind the hull (model scale OpenFOAM)

Propeller	**J**	K_T	K_Q	**ETA**	W_{Tm}	**t**	**ETA**$_h$	**ETA**$_D$
conventional	0.439	0.17459	0.021256	0.5738	0.357	0.195	1.253	1.059
kappel	0.414	0.2014	0.025220	0.5263	0.41	0.189	1.374	1.065

Table D.2: Propeller behind the hull (Full scale OpenFOAM)

Propeller	**J**	K_T	K_Q	**ETA**	W_{Ts}	**t**	**ETA**$_h$	**ETA**$_D$
conventional	0.4931	0.1547	0.01816	0.6686	0.312	0.177	1.196	0.76
kappel	0.5335	0.1797	0.02126	0.7177	0.274	0.181	1.128	0.769

Table D.3: Propeller behind the hull (Full scale ITTC)

Propeller	**J**	K_T	K_Q	**ETA**	W_{Ts}	**t**	**ETA**$_h$	**ETA**$_D$
conventional	0.486	0.1611	0.01976	0.6305	0.315	0.2008	1.1665	0.7350
kappel	0.483	0.1852	0.02382	0.5977	0.337	0.1956	1.2132	0.7228

Table D.4: Open-water Full scale (viscous only corrections)

Bare hull	**F**roude no. (Fn)	**P**ower effective (MW)
full scale (ITTC)	0.153	7.193
full scale (openFoam)	0.153	6.833

*Note: No cosideration of free surface in OpenFOAM simulations.

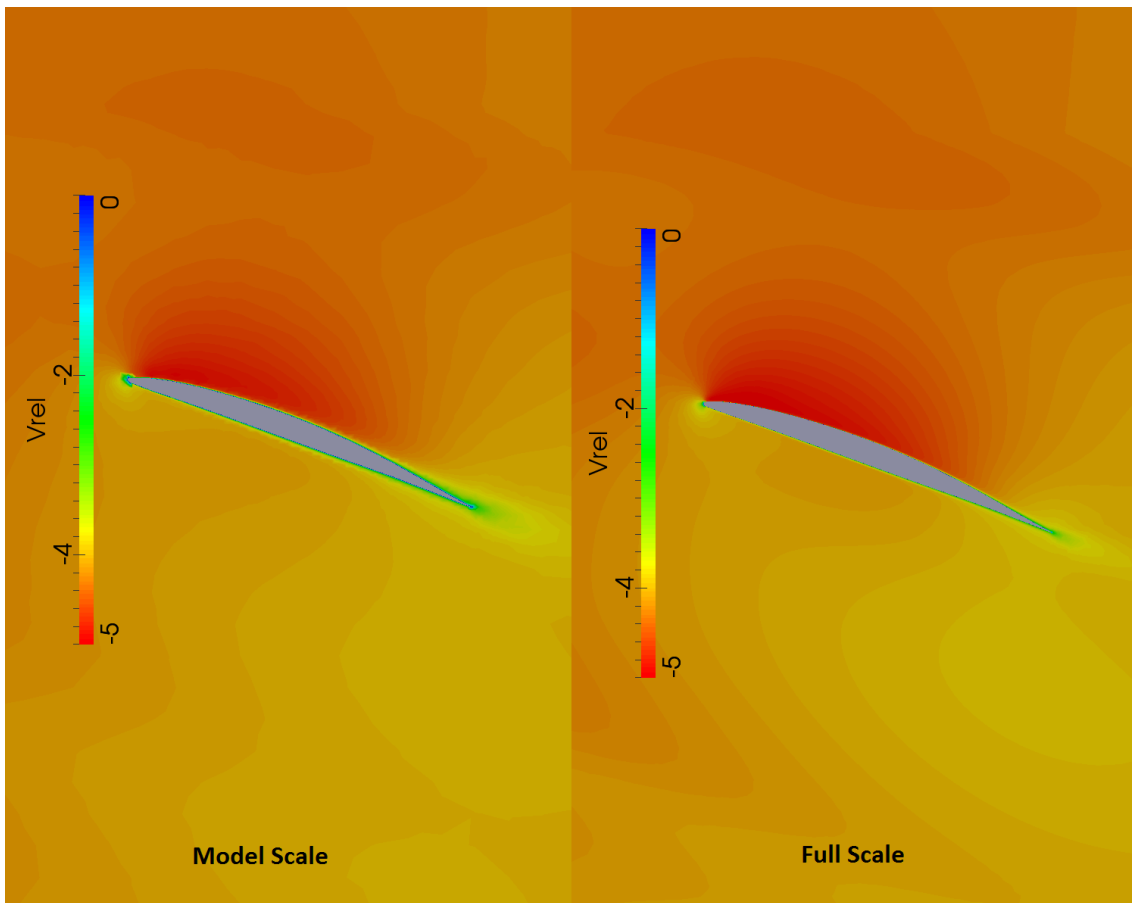# Appendix E

Miscellaneous

Figure E.1: Constant radius slice of the OpenFOAM solution for Unconventional propeller at mode-scale Reynolds number (left) and full scale Reynolds number (right).
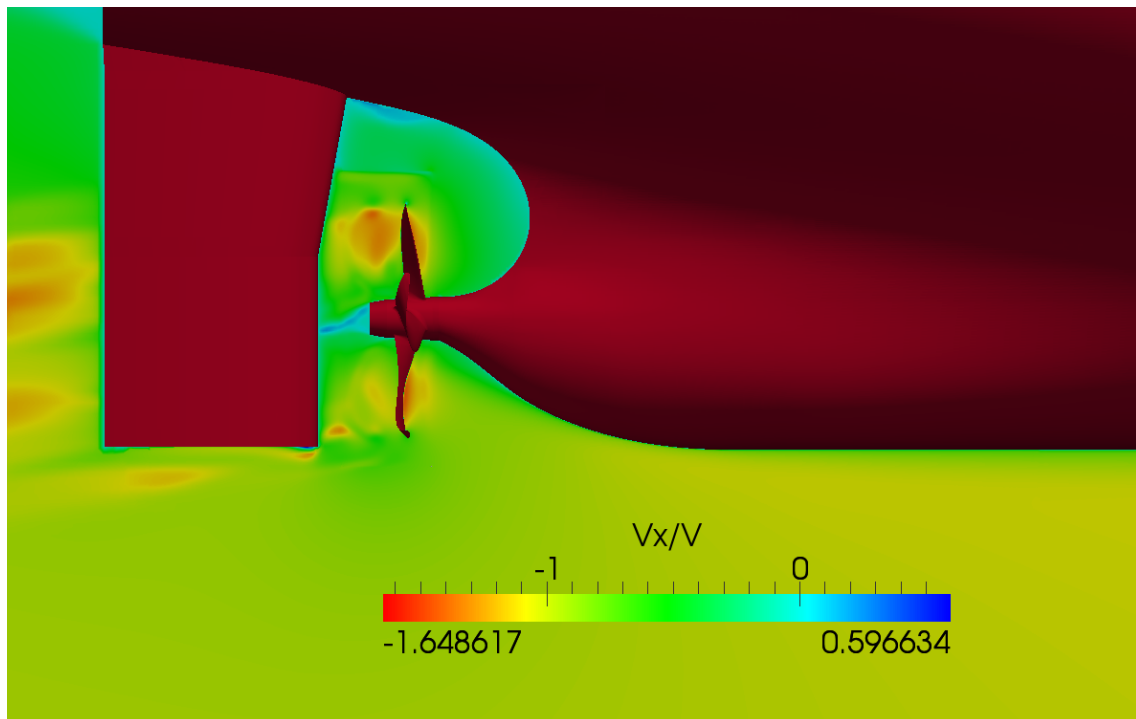
Figure E.2: Axial velocity contour for unconventional propeller behind hull (model scale)
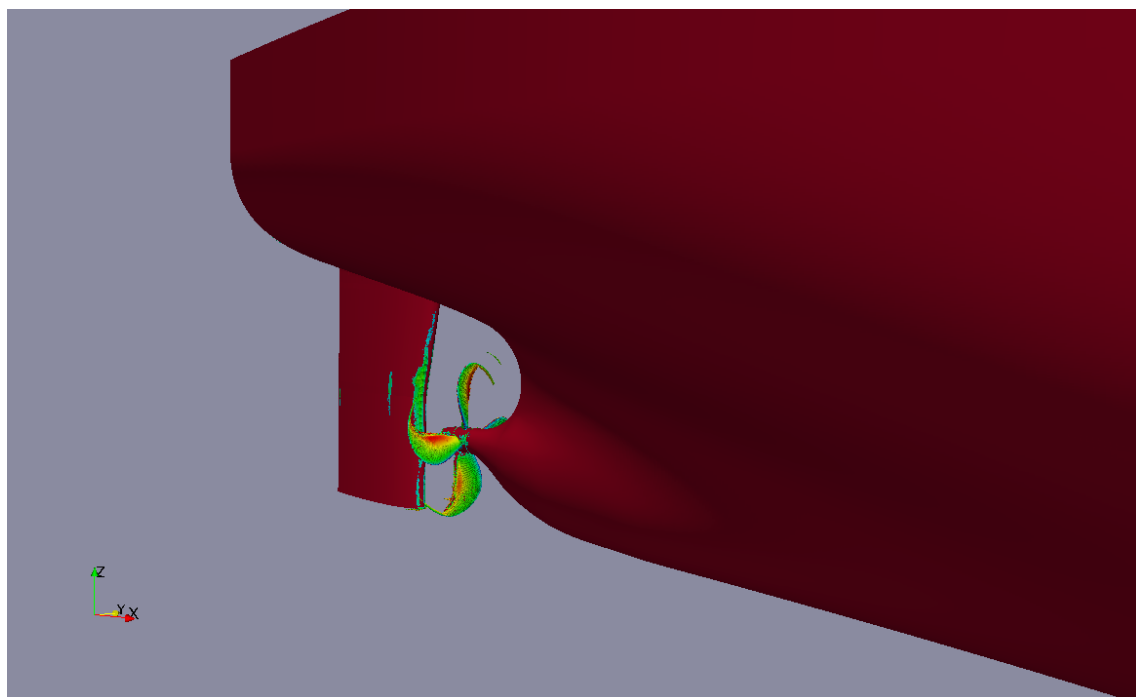

Figure E.3: Iso-Q plot of full scale conventional propeller behind a hull