

Computational Hydrodynamics of an Autonomous Underwater Vehicle (AUV)

Submitted by

STEVEN HARTA PRAWIRA

A0127702M

Department of Mechanical Engineering

In partial fulfilment of the requirements for the

Degree of Bachelor of Engineering

National University of Singapore

Session 2017/2018

Summary

This project explores the use of computational method to simulate the motion of an AUV that is acted upon by the force generated by its propulsion system. The primary objective is to study the setup of a CFD simulation that is coupled with free body dynamics available in StarCCM+ with the use of overset meshing technique. In this case, the overset meshing technique is used to model the 6 Degree-of-Freedom motion of the AUV with rotating propeller. This meshing technique allows for the visualization of the physical behaviour of the AUV experiencing various fluid forces, on top of the thrust generated by the propeller. Having the ability to visualize the physical behaviour of the AUV and to track the different parameters associated with it, this project can be used further in designing optimal control system for the AUV and many other purposes.

In achieving the above, the project is sub-divided into a few sub-projects to facilitate incremental learning within the CFD environment and the various techniques that comes with it. Also, instead of using complex AUV geometry, this project uses a simple ellipsoid AUV model as a proof-of-concept before moving further from it. Towards the end, the project also discusses how a PID controller can be interfaced with StarCCM+ for further development of the project. All in all, the project has been a fruitful learning journey to find out the possibilities of integrating CFD simulation and rigid body dynamics with StarCCM+.

Acknowledgement

Firstly, I would like to thank Team Bumblebee for allowing me to explore something beyond what has always been done in the mechanical subteam for the AUV. Special mention goes to Grace and Eng Wei who have supported my proposal of this project. Additionally, I would like to thank my mechanical subteam-mates for providing me the much-needed computational resources from our shared computer unit.

Secondly, I would also like to thank Dr. Murali Damodaran for imparting inspirations in the proposal of this project and for pushing me beyond limits to independently learn and pick up the skills and understanding of CFD. Despite being left halfway through my project, I am glad to have experienced such demanding tutelage.

Thirdly, the delivery of this project is only possible with the help of Prof. Teo Chiang Juay and his team, who have gladly lent me StarCCM+ licenses to be used in the duration of my project and answered the various queries that I had with StarCCM+.

Fourthly, I would like to thank Prof. Shu Chang and Dr. Liu Ningyu for taking care of my project to its completion and for providing key insights in making this project a successful one. None of this would have been possible without the constant support and guidance from Prof. Shu and Dr. Liu.

Last but not least, my utmost gratitude goes to my parents, family members and friends who have constantly pushed me and questioned the idea behind my project just to motivate me to produce my very best.

Table of Contents

1. Introduction.....	1
1.1.Aim and Objective.....	1
1.2.Project Approach.....	2
2. Literature Review and Relevant Theories.....	2
2.1.Applied CFD on AUV.....	2
2.2.Different Types of Mesh for CFD Simulation.....	3
2.3.Related Work: Computational Aeromechanics and Control of Quadrotor.....	4
2.4.Virtual Disk Theory.....	4
2.5.Simulated Operating Conditions.....	6
2.6.Y+ Concept and Boundary Layer Theory.....	7
2.7.Turbulence Modeling with Reynolds-Averaged Navier-Stokes (RANS).....	8
2.8.Dynamic Fluid Body Interaction (DFBI) Motion Modeling on StarCCM+.....	10
3. Incremental Sub-Projects.....	11
3.1.Study of Propulsion System using Virtual Disk Method.....	11
3.1.1. Problem Statement.....	11
3.1.2. Problem Setup on StarCCM+.....	12
3.1.3. Results and Discussion.....	13
3.1.4. Results Verification.....	15
3.2.Preliminary Study on Quality of Mesh in Capturing Near-Wall Effect.....	16
3.2.1. Problem Statement.....	16
3.2.2. Problem Setup on StarCCM+.....	17
3.2.3. Results and Discussion.....	18
3.3.Modeling of 6-DOF Free Motion of AUV with Virtual Disk Propulsion.....	19
3.3.1. Problem Statement.....	19
3.3.2. Problem Setup on StarCCM+.....	20
3.3.3. Results and Discussion.....	23
3.4.Modeling of 6-DOF Free Motion of AUV with Actual Spinning Propeller.....	25
3.4.1. AUV Propeller Verification.....	25
3.4.2. Problem Statement for Sub-project 4.....	28
3.4.3. Problem Setup on StarCCM+.....	28
3.4.4. Results and Discussion.....	29
3.4.5. Results Verification.....	32
3.5.Modeling of Controlled Motion of AUV using an Actual Spinning Propeller.....	34
4. Conclusion.....	36
5. Future Work: Implementation of Controlled Motion with Multi-Propeller AUV.....	37
References.....	41
Appendix A: Boundary Layers in Turbulent Flow [17].....	43
Appendix B: Velocity Profiles in Turbulent Flow [17].....	43
Appendix C: Details of Spalart Allmaras Models [18].....	44
Appendix D: Details of Wall Treatment in StarCCM+ [18].....	45
Appendix E: Isometric View of Discretized Fluid Domain from Section 3.1.2.....	46
Appendix F: StarCCM+ Userguide on Setting Direction of Thrust for Virtual Disk [18].....	46
Appendix G: Propeller Performance Data for Virtual Disk Setup [22].....	47
Appendix H: Physics Model Selection for Section 3.2.2.....	47
Appendix I: Specification of Pressure Values at Different Boundaries Details.....	48
Appendix J: Pressure and Vorticity Plot for Section 3.3.....	49

Appendix K: Thrust Monitor from Section 3.3.3.....	50
Appendix L: Wall Y+ Monitor for Section 3.3.3.....	50
Appendix M: Control Parameters Tracked for Sub-project 3.....	51
Appendix N: T200 Thruster Specifications [19].....	55
Appendix O: Experimental Benchmarking of T200 Thruster Results.....	56
Appendix P: Mesh Visualization for CFD Thrust Values Verification.....	56
Appendix Q: Full Results for CFD Thrust Values Verification.....	57
Appendix R: Pressure and Vorticity Contour Plots for Section 3.4.4.....	59
Appendix S: Control Parameters Tracked for Sub-project 4.....	60
Appendix T: Quadrotor Vertical Takeoff Using Overset Meshing Technique [9].....	64

List of Figures

Figure 1. Bumblebee AUV Geometry [1]	1
Figure 2. Approximate Dimension of Simplified AUV [2]	2
Figure 3. Approximate Dimension of Simplified AUV Propeller [2]	2
Figure 4. Control Volume Around Virtual Disk	4
Figure 5. Bluefin-21 AUV [13]	6
Figure 6. HUGIN AUV [15]	7
Figure 7. Dimension of Fluid Domain	12
Figure 8. Mesh Generation (Section View).....	12
Figure 9. Thrust Direction Specification	13
Figure 10. Thrust Monitor	14
Figure 11. Initial Pressure Distribution Data Plot.....	14
Figure 12. Pressure Contour of Virtual Disk	14
Figure 13. Velocity Magnitude Contour of Virtual Disk	15
Figure 14. Velocity (Left) and Pressure (Right) Contours and Streamwise Pressure (Bottom) [3].....	16
Figure 15. Dimension of Fluid Domain	17
Figure 16. Different Views of Mesh Generated	18
Figure 17. Wall y+ Monitor	19
Figure 18. Constant Gauge Pressure Set on Top Boundary of Known Depth	20
Figure 19. Gauge Pressure at Each Depth Level on Side Boundaries	20
Figure 20. Constant Gauge Pressure at Bottom Boundary of Known Distance from Top Boundary	21
Figure 21. New Coordinate System to Find Z-Position from Top Boundary	21
Figure 22. Setup of User-Defined Field Function in StarCCM+	22
Figure 23. Position of Virtual Disk in AUV Body.....	22
Figure 24. Progress of Velocity Magnitude Contour Plots	23
Figure 25. AUV Axes of Motion.....	24
Figure 26. Y-translation of AUV Time Plot.....	24
Figure 27. Z-translation of AUV Time Plot.....	24
Figure 28. X-translation of AUV Time Plot	25
Figure 29. Thrust Measurement Jig	26
Figure 30. CFD and Experimental Thrust Values Comparison	27

Figure 31. Section Plane Breakdown of Mesh.....	28
Figure 32. Progress of Velocity Magnitude Contour Plots.....	30
Figure 33. AUV Axes of Motion.....	31
Figure 34. X-Translation of AUV Time Plot.....	31
Figure 35. Y-translation of AUV Time Plot.....	31
Figure 36. Z-translation of AUV Time Plot.....	32
Figure 37. Thrust Value Monitor for AUV.....	32
Figure 38. Singh's Quadrotor Overset Mesh System [9].....	33
Figure 39. Fluid Domain and AUV Placement Illustration	34
Figure 40. Signal Flow Schematic Illustration.....	36
Figure 41. Forces and Moments on Bumblebee AUV	38
Figure 42. Transformation Matrix R	38
Figure 43. Overset Mesh System for Bumblebee AUV	40

List of Symbols

Symbol	Description
$Q_{additional}$	Increased flow rate downstream of virtual disk
$Q_{downstream}$	Flow rate downstream of virtual disk
$Q_{upstream}$	Flow rate upstream of virtual disk
A_{out}	Cross-sectional area of high velocity flow out of control volume
V_{out}	Desired flow velocity downstream of virtual disk
V_{in}	Initial flow velocity upstream of virtual disk
S	Cross-sectional area of control volume inlet
T	Thrust generated by virtual disk
ρ	Density
A_0	Cross-sectional area of virtual disk
V_1	Flow velocity at virtual disk
p_2	Pressure upstream of virtual disk
p_1	Pressure downstream of virtual disk
p_0	Pressure of fluid distant from the virtual disk
a	Velocity increase factor
Re	Reynolds number
U	Flow speed
L	Characteristic length
ν	Kinematic viscosity of fluid
y^+	Distance from wall measured in terms of viscous length
y	Normal distance from wall to wall-cell centroid
u^*	Reference velocity
$\tau(y)$	Wall shear stress
μ	Dynamic viscosity of fluid
u^+	Wall-parallel velocity non-dimensionalized with u^*

\vec{v}	Fluid velocity vector
f_b	Resultant of body forces (gravity and centrifugal forces)
φ	Arbitrary solution variable (such as velocity, pressure, energy, etc)
$\bar{\varphi}$	Averaged value of arbitrary solution variable
φ'	Fluctuating value of arbitrary solution variable
v_g	Reference frame velocity relative to global laboratory frame
I	Identity tensor
$T_{viscous}$	Viscous stress tensor
T_t	Reynolds stress tensor
μ_t	Turbulent eddy viscosity
S	Main strain rate tensor
\bar{v}	Mean velocity vector
\vec{F}	Total force vector on AUV
m	Mass of AUV
\vec{V}	Linear velocity vector on AUV
t	Time
\vec{T}	Total torque vector on AUV
\tilde{I}_{CG}	Mass moment of inertia of AUV with respect to AUV center of gravity
\vec{W}	Angular velocity vector of AUV
\vec{F}_p	Pressure force vector
\vec{T}_p	Torque vector caused by pressure forces
\vec{F}_τ	Shear force vector
\vec{T}_τ	Torque vector caused by shear forces
P_i	Pressure acting on face i
\vec{A}_i	Area vector of face i
τ_i	Shear stress acting on face i
\vec{r}_i	Vector for distance from the AUV center of gravity to the center of face i
R	Ramping function
$t_{release}$	Release time
t_{ramp}	Ramp time
\vec{V}^{n+1}	Linear velocity vector at timestep n+1
\vec{V}^n	Linear velocity vector at timestep n
Δt	Timestep size
\vec{W}^{n+1}	Angular velocity vector at timestep n+1
\vec{W}^n	Angular velocity vector at timestep n
K_T	Propeller thrust coefficient
K_Q	Propeller torque coefficient
ETA	Propeller efficiency
J	Advance ratio

$P_{reference}$	Reference pressure
$P_{absolute,h_0}$	Absolute pressure at depth h_0
$P_{atmospheric}$	Atmospheric pressure
g	Acceleration due to gravity
h_0	Depth of top boundary
z	Depth level measured from top boundary
z_0	Depth level of bottom boundary measured from top boundary
I_{yy}	y-component of mass moment of inertia
I_{xx}	x-component of mass moment of inertia
I_{zz}	z-component of mass moment of inertia
$X_{desired}$	Desired state in controller
$X_{current}$	Current state in controller
e	Discrepancy between desired and current states
u	Control input
P	Proportional gain for controller
I	Integral gain for controller
D	Derivative gain for controller
$T_{forward}$	Forward thrust
ω	Rotational speed of propeller
$T_{reverse}$	Reverse thrust
$T_{required}$	Required thrust
F_{drag}	Drag force on thruster
Y_d	Desired Y-position
Y	Current Y-position
F_x	Resultant force in x-direction
F_y	Resultant force in y-direction
F_z	Resultant force in z-direction
T_ϕ	Resultant moment in roll axis
T_θ	Resultant moment in pitch axis
T_ψ	Resultant moment in yaw axis
F_1 to F_8	Forces generated by thrusters 1 to 8

1. Introduction

This project looks at hydrodynamics of an AUV by integrating CFD with 6-Degree-of-Freedom (6-DOF) rigid body dynamics in StarCCM+ to study the AUV motion. This can be used to examine scenarios, to estimate drag and to optimize controls of the AUV.

1.1. Aim and Objective

In this project, the author aims to gain exposure to intricacies of CFD and to understand the techniques involved in integrating CFD with 6-DOF rigid body dynamics and control system for an AUV. Ultimately, the objective of this project is to use the techniques learned as a design tool for future generations of a multi-propeller Bumblebee AUV, which the author has an experience designing, previously. Towards the end, techniques learnt will also be verified against relevant work to enhance its credibility.

To begin, instead of focusing directly on a complex system like the Bumblebee AUV, the project will be done on a simplified single-propeller AUV. The results obtained will be used as a proof-of-concept that techniques learnt will work. Hence, although results obtained using the simplified AUV do matter, the details of the results will not be of much significance. Result details will become significant when the proven working techniques are implemented on the Bumblebee AUV.



Figure 1. Bumblebee AUV Geometry [1]¹

The simplified AUV model to be used is obtained from GrabCAD:

¹ Convention used: “[1]” refers to citation number as per the References list in page 41.

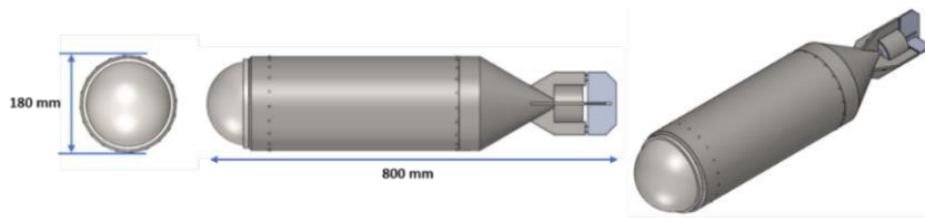


Figure 2. Approximate Dimension of Simplified AUV [2]

The AUV model also comes with a propeller model which can be seen as follows:

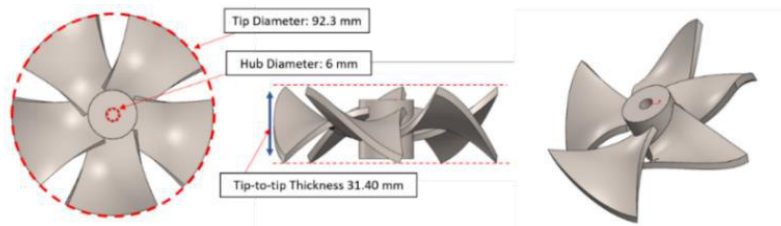


Figure 3. Approximate Dimension of Simplified AUV Propeller [2]

1.2. Project Approach

To facilitate incremental learning of the techniques involved, the project is divided into:

1. Study of Propulsion System using Virtual Disk Method;
2. Preliminary Study of Quality of Mesh in Capturing Near-Wall Effect;
3. Modeling of 6-DOF Free Motion of AUV using the Virtual Disk Propulsion System;
4. Modeling of 6-DOF Free Motion of AUV using an Actual Spinning Propeller;
5. Modeling of Controlled Motion of AUV using an Actual Spinning Propeller.

2. Literature Review and Relevant Theories

2.1. Applied CFD on AUV

Over the years, academic efforts in CFD has centered around the advancement of CFD methods and algorithms. CFD has, thus, evolved from a theoretical research subject into a tool capable of analyzing real engineering problems [3]. This development has charted new paths for applied CFD study in many fields, including hydrodynamics study of marine structures, such as: marine crafts and propellers [4]. This application has proven to provide accurate results with marginal discrepancies from empirical values

[5]. Further development of comprehensive CFD packages (e.g. StarCCM+ and Fluent) and high performance computing have also allowed the study of dynamic fluid body interaction (DFBI), looking not only at fixed objects, but also dynamic systems involving moving objects in a fluid field over a period of time [6]. Relating back to this project, the advancement in CFD will, indeed, be very useful as the author aims to look into the coupling of DFBI feature in StarCCM+ with control laws to model the hydrodynamics of an AUV that is set in motion by its propulsion system.

2.2. Different Types of Mesh for CFD Simulation

Discretization of fluid domain in CFD simulation can, generally, use three types of mesh: hexahedral, tetrahedral and polyhedral meshes [7]. A study was done to compare results in monitoring pressure drop of flow within a duct using the three types of mesh of comparable size [8]. In this study, a few key results were obtained. Firstly, from pressure residual monitors, solution on polyhedral mesh produced lowest absolute residual value, while achieving fastest pressure residual convergence. Secondly, despite minor discrepancies in converged pressure drop value, solution obtained using polyhedral mesh was the fastest to reach steady state, followed by that using hexahedral and tetrahedral meshes. Lastly, runtime to steady state using polyhedral mesh emerged as the shortest, followed by that using hexahedral and tetrahedral meshes. This study showed that polyhedral mesh has potential in yielding equally accurate results with added benefits, such as: faster convergence, robust convergence with lower residuals and shorter solution runtime. Looking at such potential, this project shall consider the use of polyhedral mesh type available in StarCCM+.

2.3. Related Work: Computational Aeromechanics and Control of Quadrotor

In his study of quadrotor flight dynamics, Singh [9] focused on the integration of rigid body dynamics, controls and CFD to solve the 3D incompressible Navier-Stokes with a turbulence model. He used this study to assess the performance of quadrotor in free, near ground and above water flight. This study was done in a CFD environment with flow computed using Spalart-Allmaras turbulence model widely used for external aerodynamics. Singh also coupled the use of DFBI model to solve the rigid body dynamics simultaneously. Additionally, to simulate autonomous flight patterns for the quadrotor, he implemented control laws developed using MATLAB and coupled with the simulation using Javascript. As a whole, Singh managed to model an autonomous vertical hovering and rolling stability maneuvers for the quadrotor with the aforementioned implementation. Having some underlying similarities with the current project, Singh's study shall thus be used as reference to validate some steps within this project that are relevant to his study.

2.4. Virtual Disk Theory

Virtual disk is a simple representation of propeller where its effect is attained through a pressure discontinuity. This simplified model will reduce unnecessary complexity, especially in the early part of the project [3]. Consider the following control volume:

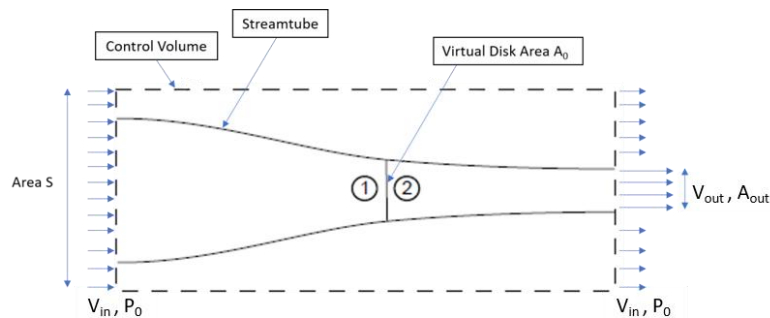


Figure 4. Control Volume Around Virtual Disk

The virtual disk accelerates surrounding fluid such that stream of high velocity fluid, V_{out} , emerges out of the control volume. Regions far away from the disk at the outlet are assumed to maintain fluid velocity of V_{in} , same as the inlet velocity. Specific pressure boundaries are imposed at the disk in order to achieve the desired velocity of V_{out} , given inlet velocity V_{in} and the propeller geometry. Enforcing continuity to control volume, increased flow rate downstream also increases flow rate upstream:

$$Q_{additional} = A_{out}(V_{out} - V_{in}) \quad (1)$$

From Newton 2nd Law and some manipulation, thrust (T) can then be calculated as:

$$T = \rho A_{out} V_{out} (V_{out} - V_{in}) \quad (2)$$

Enforcing continuity, $\rho A_{out} V_{out} = \rho A_0 V_1$, where V_1 is fluid velocity at the virtual disk and A_0 is the cross sectional area of the virtual disk, equation (2) becomes:

$$T = \rho A_0 V_1 (V_{out} - V_{in}) \quad (3)$$

And,

$$T = A_0 (p_2 - p_1) \quad (4)$$

where p_2 is pressure upstream and p_1 is pressure downstream of the disk. Thus,

$$(p_2 - p_1) = \rho V_1 (V_{out} - V_{in}) \quad (5)$$

Using Bernoulli's principle, expressions for downstream and upstream regions are:

$$\text{Upstream: } p_0 + \frac{1}{2} \rho V_{in}^2 = p_1 + \frac{1}{2} \rho V_1^2 \quad (6a)$$

$$\text{Downstream: } p_0 + \frac{1}{2} \rho V_{out}^2 = p_2 + \frac{1}{2} \rho V_1^2 \quad (6b)$$

Equations (6) and (5) are further re-arranged to,

$$V_1 = \frac{1}{2} (V_{out} + V_{in}) \quad (7)$$

Also, if V_{out} can be defined as a linear increase in inlet velocity:

$$V_{out} = V_{in} (1 + x) \quad (8)$$

where x is the velocity increase factor. Hence, velocity V_1 at the virtual disk is:

$$V_1 = V_{in} \left(1 + \frac{x}{2}\right) \quad (9)$$

Substitution of (8) and (9) into (6a) and (6b) and re-arranging will yield:

$$p_1 = P_0 - \rho V_{in}^2 \frac{x}{2} \left(1 + \frac{x}{4}\right) \quad (10)$$

$$p_2 = P_0 + \rho V_{in}^2 \frac{x}{2} \left(1 + \frac{3x}{4}\right) \quad (11)$$

$$T = \rho A_0 V_{in}^2 \left(1 + \frac{x}{2}\right) x \quad (12)$$

2.5. Simulated Operating Conditions

A. Specification of Operating Fluid

This project uses fresh water of density 1000 kg m^{-3} and operating temperature 25°C . At this temperature, kinematic viscosity of water is $0.8927 \times 10^{-6} \text{ m}^2\text{s}^{-1}$ [10]. The operating pressure will be discussed further later. The fresh water properties can be governed by the IAPWS-IF97 (Water) model in Star-CCM+.

B. Estimate of Operating Reynolds Number

The operating Reynolds Number can be calculated using the following equation [11]:

$$Re = \frac{U L}{\nu} \quad (13)$$

where U is speed, L is characteristic length and ν is kinematic viscosity. To estimate operating Reynolds number, the author considers typical average speed for an AUV. The first type is the Bluefin-21 AUV with an average operating speed of 1.5 ms^{-1} [12].



Figure 5. Bluefin-21 AUV [13]

Another common AUV is the HUGIN-1000 with average operating speed of 2 ms^{-1} [14].



Figure 6. HUGIN AUV [15]

Using average speed of $U = 1.5 \text{ ms}^{-1}$ and the length of the simplified AUV model ($L = 800 \text{ mm}$) in Figure 3, the estimated operating Reynolds number is:

$$Re = \frac{1.5 \times 0.8}{0.8927 \times 10^{-6}} = 1.344 \times 10^6$$

As AUVs come in different shapes, a conservative case of external flow over flat plate critical Reynolds number ($Re = 500000$) is used [16]. As operating Reynolds number is more than critical Reynolds number, the AUV will be operating in turbulent regime.

2.6. Y+ Concept and Boundary Layer Theory

In turbulent regime, near-wall flow behaviour is a complicated occurring and to distinguish different near-wall regions, wall y^+ concept is applied [17]. y^+ is a dimensionless quantity that gives distance from wall measured in viscous length:

$$y^+ = \frac{yu^*}{\nu} \quad (14)$$

where y is wall to wall-cell distance, u^* is reference velocity and ν is kinematic viscosity.

Generally, near-wall flow region is separated into 3 layers (Appendix A and B):

A. Linear sub-layer ($y^+ < 5$)

Based on no-slip condition, fluid is stationary at solid surface and turbulent eddy motion will also stop at near-wall region. As such, fluid at the near-wall region will be dominated by viscous shear and that the shear stress in this layer can be assumed equal to the wall shear stress. Hence, this can be written as:

$$\tau(y) = \mu \frac{\partial U}{\partial y} \approx \tau_w \quad (15)$$

Application of boundary conditions and some manipulations will yield:

$$u^+ = y^+ \quad (16)$$

where u^+ is wall-parallel velocity non-dimensionalized with reference velocity u^* :

$$u^+ = \frac{u}{u^*} \quad (17)$$

B. Log-law layer ($30 < y^+ < 500$)

Outside the linear sub-layer, there is a region where viscous and turbulent effects are equally significant. Here, shear stress is assumed to be same as wall shear stress and changing gradually as it goes away from wall. The y^+ to u^+ relationship is:

$$u^+ = \frac{1}{k} \ln y^+ + C \quad (18)$$

where k and C are constants found empirically.

C. Buffer layer ($5 < y^+ < 30$)

In between the two aforementioned layers, neither law holds, with the furthest distinction from both laws occurring approximately at $y^+ = 11$. This means that before 11 wall units, linear approximation of shear stress is more accurate and after 11 wall units, logarithmic approximation is to be applied.

2.7. Turbulence Modeling with Reynolds-Averaged Navier-Stokes (RANS)

A turbulent flow is by nature unsteady[18]. RANS formulation is developed from Navier-Stokes (NS) equations that are averaged; In general, the NS equations for an incompressible fluid flow is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \vec{v} = 0 \quad (19)$$

$$\frac{\partial}{\partial t} (\rho \vec{v}) + \rho \vec{v} \cdot \nabla \vec{v} = -\nabla p + \mu (\nabla \cdot \nabla) \vec{v} + f_b \quad (20)$$

For RANS, each variable φ is separated into averaged value and fluctuating value:

$$\varphi = \bar{\varphi} + \varphi' \quad (21)$$

Inserting the broken-down solution variables into the equations (19) and (20) yields:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot [\rho(\bar{v} - v_g)] = 0 \quad (22)$$

$$\frac{\partial}{\partial t}(\rho \bar{v}) + \nabla \cdot [\rho \bar{v}(\bar{v} - v_g)] = -\nabla \cdot \bar{p}I + \nabla \cdot (T_{viscous} + T_t) + f_b \quad (23)$$

where ρ is density, \bar{v} and \bar{p} are mean velocity and pressure, respectively, v_g is the reference frame velocity relative to global frame, I is identity tensor, $T_{viscous}$ is viscous stress tensor and f_b is resultant of the body forces. The equations are practically identical to the original NS equations, other than an additional term in the momentum transport equation. This term is the Reynolds stress tensor:

$$T_t = -\rho \begin{pmatrix} \overline{u'u'} & \overline{u'v'} & \overline{u'w'} \\ \overline{u'v'} & \overline{v'v'} & \overline{v'w'} \\ \overline{u'w'} & \overline{v'w'} & \overline{w'w'} \end{pmatrix} \quad (24)$$

The next step is to model T_t with respect to mean flow quantities. The approach in StarCCM+ deployed for this simulation is the eddy viscosity models. The existence of turbulent eddy viscosity μ_t makes it simpler for modeling of Reynolds stress tensor in terms of mean flow quantities. The common model is the Boussinesq approximation:

$$T_t = 2\mu_t S - \frac{2}{3}(\mu_t \nabla \cdot \bar{v})I \quad (25)$$

where S is the main strain rate tensor, \bar{v} is the mean velocity and I is identity tensor.

There are various transport equations to derive the turbulent viscosity μ_t . For this project, Spalart-Allmaras (SA) models will be used. The SA models are typically suited for external-flow applications (Appendix C). Additionally, the SA models can be coupled with different wall treatments in StarCCM+ (Appendix D). Throughout the project, the standard SA model combined with all y^+ wall treatment will be used.

2.8. Dynamic Fluid Body Interaction (DFBI) Motion Modeling on StarCCM+

StarCCM+ can solve 6-DOF dynamic body motion equations coupled with flow equations [9]. Here, the two reference frames used are, firstly, the one attached to the moving AUV body (AUV frame) and, secondly, global inertial frame attached to fluid field (inertial frame). The velocity transformation from the AUV frame to the inertial frame will yield relative velocity term v_g in the NS equations (eqn. 22 & 23).

Subsequently, the governing equations for 6-DOF motion in the AUV frame is:

$$\vec{F} = m \frac{\partial \vec{V}}{\partial t} \quad (26)$$

$$\vec{T} = \vec{I}_{CG} \frac{\partial \vec{W}}{\partial t} \quad (27)$$

where \vec{F} is force vector, \vec{T} is torque vector, \vec{I}_{CG} is mass moment of inertia of the AUV relative to rotation axes through the AUV's center of gravity (CG), m is the mass of the AUV, and \vec{V} & \vec{W} are the linear and angular velocity vectors of the AUV, respectively.

Also, fluid forces acting on AUV consists of pressure and shear forces. Total force and torque include pressure force (\vec{F}_p) and torque (\vec{T}_p) and shear force (\vec{F}_τ) and torque (\vec{T}_τ):

$$\begin{aligned} \vec{F} &= \vec{F}_p + \vec{F}_\tau \\ \vec{F} &= \sum_{surface} (P_i \vec{A}_i + \tau_i \vec{A}_i) \end{aligned} \quad (28)$$

$$\begin{aligned} \vec{T} &= \vec{T}_p + \vec{T}_\tau \\ \vec{T} &= \sum_{surface} (\vec{r}_i \times (P_i \vec{A}_i) + \vec{r}_i \times (\tau_i \vec{A}_i)) \end{aligned} \quad (29)$$

where P_i is pressure acting on surface i , \vec{A}_i is area of surface i , τ_i is shear stress acting on surface i and \vec{r}_i is vector for distance from AUV's CG to center of surface i .

In StarCCM+, force and torque applied onto the AUV uses ramping function R to prevents sudden force impact leading to solution instability. This is incorporated as:

$$\vec{F} = R \sum_{surface} (P_i \vec{A}_i + \tau_i \vec{A}_i) \quad (30)$$

$$\vec{T} = R \sum_{surface} (\vec{r}_i \times (P_i \vec{A}_i) + \vec{r}_i \times (\tau_i \vec{A}_i)) \quad (31)$$

where the ramping function can be written as:

$$R = \begin{cases} 0 & t < t_{release} \\ \frac{t-t_{release}}{t_{ramp}} & t_{release} \leq t < t_{release} + t_{ramp} \\ 1 & t \geq t_{release} + t_{ramp} \end{cases} \quad (32)$$

where $t_{release}$ is release time and t_{ramp} is ramp time. It is also suggested that $t_{release}$ used is 10 to 50 timesteps to ensure proper initialization. Here, the timestep used is, in general, 0.001 s. Thus, the release time used is 0.01 s. And, ramp time used is 0.02 s (20 timesteps) in order for faster result progress. Finally, the linear and angular velocities are solved using equations (26) and (27) on a forward difference scheme:

$$\vec{V}^{n+1} = \vec{V}^n + \frac{\vec{F}}{m} \Delta t \quad (33)$$

$$\vec{W}^{n+1} = \vec{W}^n + \frac{\vec{T}}{I_{CG}} \Delta t \quad (34)$$

3. Incremental Sub-Projects

3.1. Study of Propulsion System using Virtual Disk Method

3.1.1. Problem Statement

This sub-project is done to simulate flow field over the AUV propeller (Fig. 3) using virtual disk model. For this problem, the propeller is fixed within a fluid domain and set to produce a constant thrust of 25 N onto an initially stationary fluid field. The author

aims to use this sub-project to learn the setup of virtual disk that will be used in sub-project 3 to create a simplified propulsion system for the AUV.

3.1.2. Problem Setup on StarCCM+

A. Fluid Domain Setup

For this problem, a virtual disk is placed in the center of a 4 m by 0.9 m by 0.9 m fluid domain. The author chooses the dimension rather arbitrarily as focus is on near field around the virtual disk itself. However, he ensures that the flow field is long enough both forward and aft of the disk in order not to disrupt flow upstream and downstream.

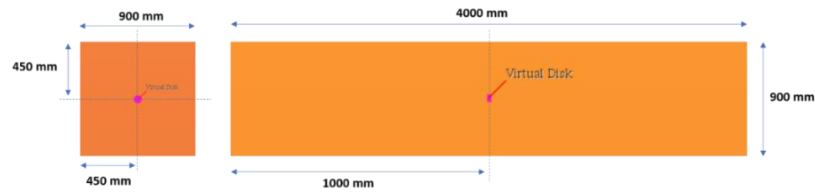


Figure 7. Dimension of Fluid Domain

Polyhedral cells with base size of 0.1 m are used to discretize the fluid domain. The mesh is further refined within a region enclosed by a cylinder concentric to the virtual disk, extending upstream and downstream of the virtual disk to capture near-disk flow details. The refined region is also made three times the diameter of the disk to better capture the effects of the disk in radial direction. The refinement uses cells with size of 0.01 m. Altogether, the fluid domain is broken down into 0.3 million cells. A section of the volume mesh is shown as follows (full view of discretized domain in Appendix E).

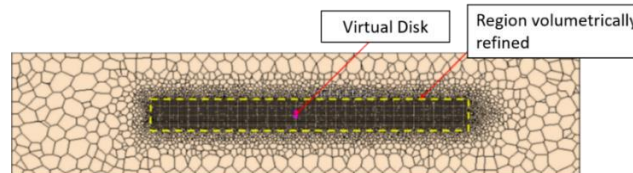


Figure 8. Mesh Generation (Section View)

For simplicity, the author assumes that the virtual disk is dipped into very shallow stationary field of water. Hence, all outer boundaries of the domain are set as pressure

outlet boundaries with approx. atmospheric pressure. For initial condition, the fluid velocity in the domain is set as 0 m/s everywhere.

B. Virtual Disk Setup in StarCCM+

Firstly, the author sets the virtual disk to follow the propeller dimensions in Fig. 3. Secondly, the virtual disk is placed in the center of fluid domain. Subsequently, the thrust direction requires an input of the normal of the disk. This relates closely to the handedness of the propeller blade, which is set to be right-hand. Hence, the direction of thrust is specified as follows (guideline for thrust direction setup in Appendix F):

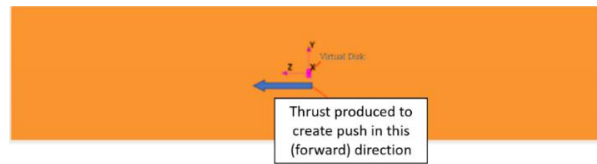


Figure 9. Thrust Direction Specification

Next, the propeller performance data is input in a table with data obtained from a typical marine propeller with specific blade angle (Appendix G). To specify operating point, StarCCM+ offers three options: Rotation rate n , Thrust T or Torque Q . Here, the author specifies the operating point to be constant thrust T of 25 N.

The last input is specification of inflow velocity plane. The two parameters to specify are the velocity plane radius and the velocity plane offset. Here, velocity plane radius is 50.77 mm. As for the velocity plane offset, it is set to be 9.23 mm.

3.1.3. Results and Discussion

For this case, the author mainly monitors the thrust curve, pressure distribution and velocity magnitude of the flow field around the virtual disk to check the results. The following thrust curve shows the constant 25 N thrust generated out of the virtual disk. The thrust generated is set to ramp up from 0 N to 25 N in 4 timesteps (0.004 s).

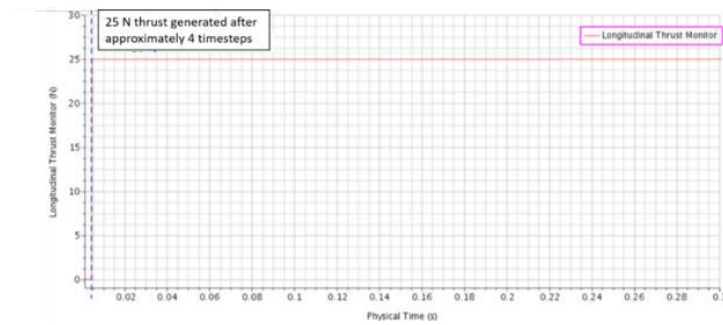


Figure 10. Thrust Monitor

Secondly, the pressure distribution is also monitored, especially in the initial stage of the thrust build-up, to capture the pressure discontinuity between the regions upstream and downstream of the disk. This is done by plotting the pressure distribution data points along the streamwise direction of the virtual disk as follows:

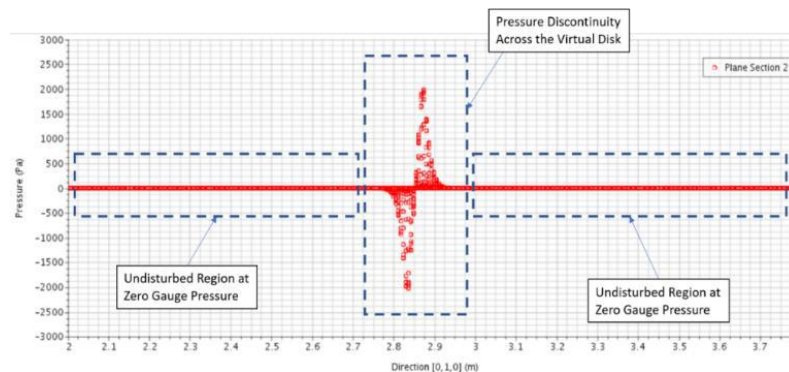


Figure 11. Initial Pressure Distribution Data Plot

The pressure discontinuity as explained from the virtual disk theory can be clearly observed. Additionally, fluid field far away from the virtual disk remains undisturbed maintaining atmospheric pressure. The same observation can also be picked up through the pressure contour plot, especially in its very initial stage of thrust generation. The pressure contour plot is as follows.

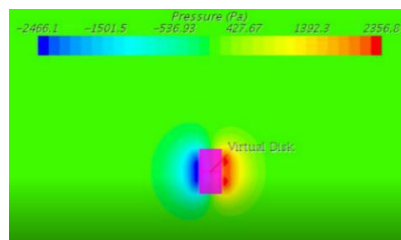


Figure 12. Pressure Contour of Virtual Disk

The region coloured dark blue refers to region of highly negative gauge pressure and the region coloured red is region with highly positive gauge pressure. Animation of how the pressure contour plot evolves can be seen here: [Animation 1](#).

Lastly, for the velocity magnitude monitor, the expected outcome is a stream of high velocity fluid out of the virtual disk. At the end of the simulation run, the velocity magnitude contour plot on the section plane can be observed as follows:

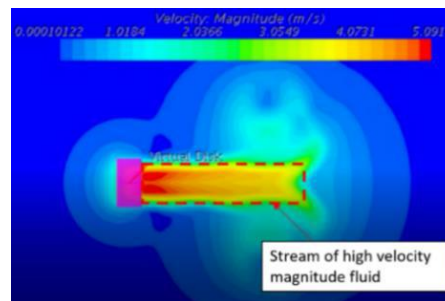


Figure 13. Velocity Magnitude Contour of Virtual Disk

Animation of how the velocity magnitude evolves can be seen here: [Animation 2](#).

3.1.4. Results Verification

To verify the above results and to ensure that the setup learnt is valid, the author compares the above results to an existing research work. In this case, the author consults a study by Coe [3]. In his study, Coe creates a simplified model of propeller for a general purpose AUV using the virtual disk method using an existing propeller geometry. The virtual disk is then simulated to produce constant thrust, very much similar to the problem statement of the above sub-project. Coe's results are as follows:

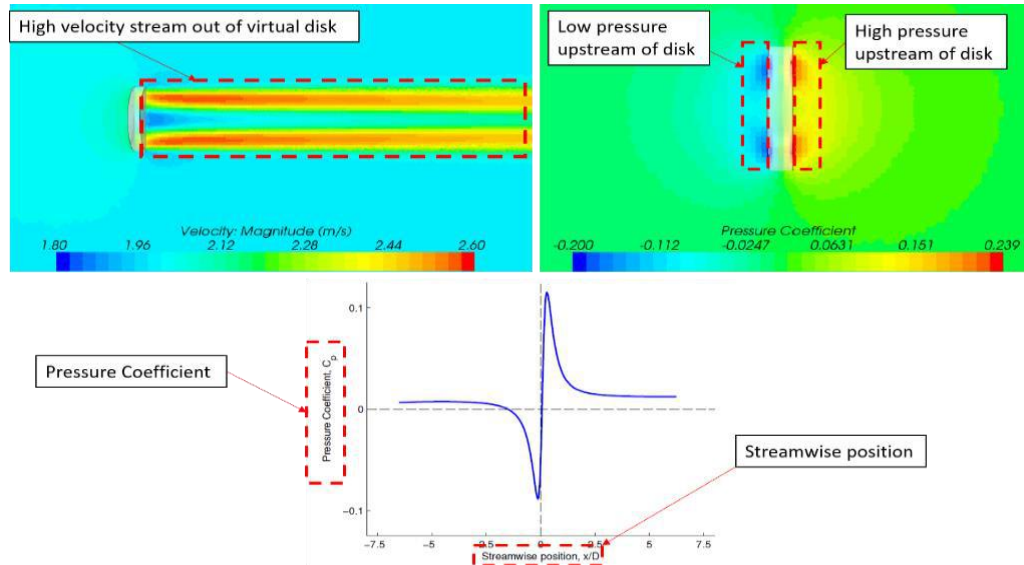


Figure 14. Velocity (Left) and Pressure (Right) Contours and Streamwise Pressure (Bottom) [3]

Comparing results in Fig. 14 to results obtained in this sub-project, it can be seen that there are, indeed, close similarities in result trends, although absolute values may differ depending on thrust value and boundary conditions specified by Coe. However, since this sub-project only looks into learning the setup of virtual disk, the similarity in trends will suffice for validation. Firstly, in both cases, the virtual disk produces high velocity flow downstream. Secondly, from the pressure contour, distinct low and high pressure regions can be observed upstream and downstream of the virtual disk, respectively. Lastly, the pressure discontinuity in streamwise direction across the virtual disk in Coe's result is also very similar to the results obtained in the seen in Fig. 11. The pressure discontinuity trends for both the sub-project result and Coe's result also tie in closely with virtual disk theory discussed in section 2.5.

3.2. Preliminary Study on Quality of Mesh in Capturing Near-Wall Effect

3.2.1. Problem Statement

Subsequently, the author aims to determine whether near-wall mesh around AUV is sufficient to capture near-wall effect and is suitable for turbulence model applied. For

this, the AUV is fixed in fluid domain against incoming flow of 0.1 m/s. Result of this sub-project will decide whether mesh created is sufficient for subsequent sub-projects.

3.2.2. Problem Setup on StarCCM+

A. Fluid Domain Setup

The AUV is placed in the center of the fluid domain. The domain has a length of approx. 8 times length of the AUV and width of approx. 15 times diameter of the AUV.

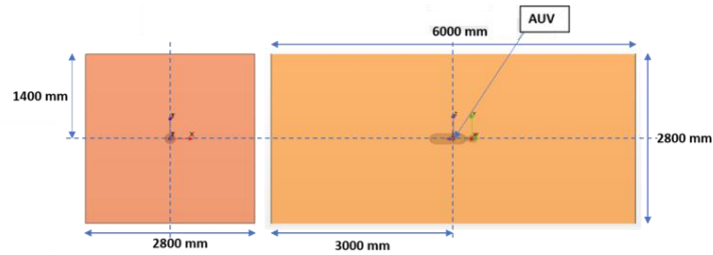


Figure 15. Dimension of Fluid Domain

The author uses overset meshing technique to separate the fluid domain into two regions: background and overset regions. The background region is the cuboid fluid domain region, while the overset region is a spherical region of diameter approx. 1.5 times length of the AUV enclosing the entire body of the AUV. This allows for motion simulation without having to perform re-meshing as the AUV moves around.

To discretize the fluid domain, the author chooses to use polyhedral cells with 0.5 m base size based on a study in Section 2.2 which shows the advantages of using polyhedral cells. Additionally, the mesh is refined on a region overlapping background and overset mesh, extending upstream and downstream of the AUV. The refinement uses cells of size 0.025m. On top of that, further refinement is done in the region enclosing the body of the AUV itself with cells of dimension 0.005 m. Finally, to capture near-wall effect, the author applies the prism layer mesher; 15 prism layers near the AUV body are used [18]. Altogether, the fluid domain is discretized into 4 million cells.

Although the exact dimensions of fluid domain and cells size used are picked rather arbitrarily, they are actually chosen with close reference to a published work by Singh [9]. This will be discussed further in sub-project 4, which makes use of the fluid domain and mesh setup done here together with other aspects picked up along the way.

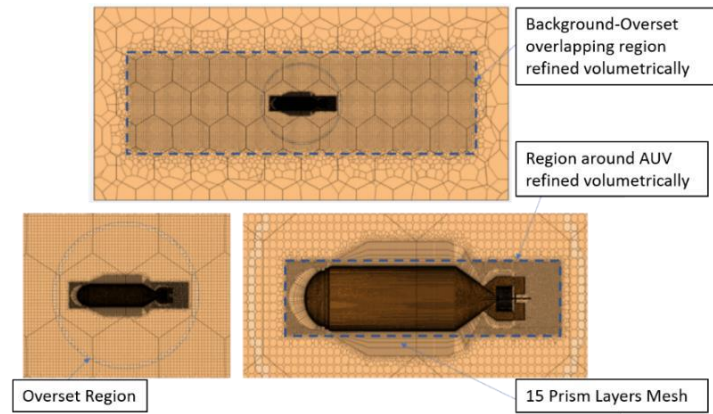


Figure 16. Different Views of Mesh Generated

Based on the problem statement, the left boundary is set as a velocity inlet with y -velocity of 0.1 m/s while zero for other components; here, y -axis is the axis parallel to the length of domain and the AUV. Subsequently, all other boundaries are set as pressure outlet with zero gauge pressure to simulate undisturbed flow condition elsewhere for simplicity similar to the previous sub-project. For initial condition, the fluid velocity in the fluid domain is set as 0.1 m/s along the y -axis.

B. Turbulence Model Selection

For this sub-project, the simulation will make use of RANS model and standard Spalart-Allmaras model coupled with all y^+ wall treatment. The turbulence model selection is set as part of the physics model can be seen in Appendix H.

3.2.3. Results and Discussion

For this study, the author mainly monitors the wall y^+ contours on the AUV. This can be a good indicator of whether near-wall mesh is sufficient to capture near-wall effect.

Moreover, there is a need to see the variation in wall y^+ values to gauge whether the turbulence model combination is suitable for the mesh created.

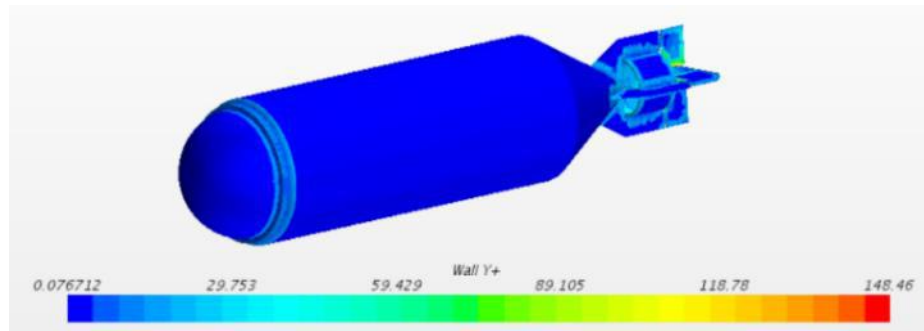


Figure 17. Wall y^+ Monitor

As seen above, almost the entire AUV body has y^+ less than 1. This means that the near-wall mesh is sufficient to capture the viscous sublayer. Also, it can be observed that on other regions, the y^+ values are generally larger than 5 even though these regions are smaller versus those regions capturing the viscous sublayer. Thus, with the presence of both low y^+ regions ($y^+ < 1$) and high y^+ regions ($y^+ > 5$), it can be concluded that the use of the standard Spalart-Allmaras turbulence model combined with all y^+ wall treatment is suitable. Altogether, it can be concluded from this study that the mesh created is sufficient to capture the near wall effect and the turbulence model combination used is suitable for this case.

3.3. Modeling of 6-DOF Free Motion of AUV with Virtual Disk Propulsion

3.3.1. Problem Statement

This sub-project simulates the motion of a single-propeller AUV in an initially stationary flow field with a virtual disk model propeller acting on it. The study looks at how thrust generated by the virtual disk can lead to full 6-DOF motion of the AUV. This sub-project will be the author's first attempt to set the AUV in motion using the simplified virtual disk propulsion system technique learnt and the overset mesh system set up earlier.

3.3.2. Problem Setup on StarCCM+

A. Fluid Domain Setup

For this sub-project, the author chooses to set all boundaries as pressure outlet that replicates hydrostatic pressure at undisturbed far field. The author tries to perform this using the following idea:

The pressure at all 6 boundaries are set as the hydrostatic gauge pressure at that depth, with pressure at top boundary as reference pressure. Setting depth of the top boundary (i.e. h_0) from surface of water, reference pressure at the top boundary:

$$P_{reference} = P_{absolute, h_0} = P_{atmospheric} + \rho g h_0 \quad (35)$$

Taking $h_0 = 20 \text{ m depth}$,

$$P_{reference} = 101325 + (1000 \times 9.81 \times 20) = 297525 \text{ Pa}$$

Subsequently, gauge pressure at each boundary surface can be set as:

a. Top Boundary

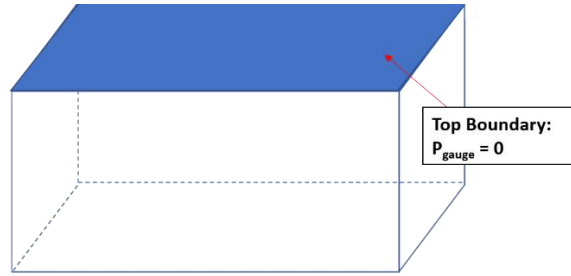


Figure 18. Constant Gauge Pressure Set on Top Boundary of Known Depth

b. Side Boundaries (Forward, Aft, Left and Right of AUV)

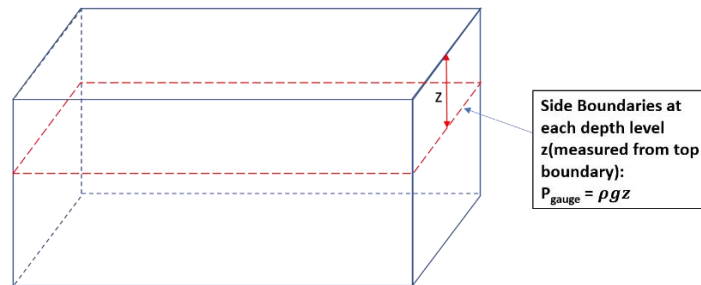


Figure 19. Gauge Pressure at Each Depth Level on Side Boundaries

c. Bottom Boundary

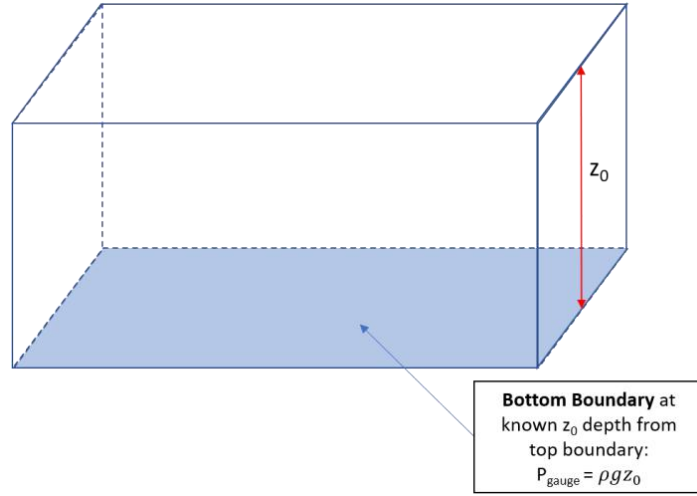


Figure 20. Constant Gauge Pressure at Bottom Boundary of Known Distance from Top Boundary

The author then implemented the above idea in StarCCM+ as follows. Firstly, “Reference Pressure” can be set as part of the physics models selected. Using the set reference pressure, gauge pressure at each of the boundaries will then be specified. For top and bottom boundaries, gauge pressure method can be set as constant with values specified according to each boundary (Appendix I).

For the side boundaries (forward, aft, left and right of AUV), the author sets up a new coordinate system will be used to obtain the z-position which represents the depth level at each point away from the top boundary.

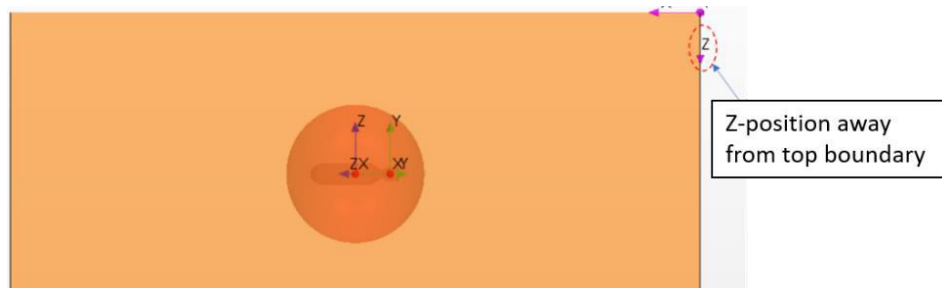


Figure 21. New Coordinate System to Find Z-Position from Top Boundary

Thus, hydrostatic gauge pressure at each point along the sides can be specified via a User-Defined Function that uses z-coordinate of the points along the side boundaries:

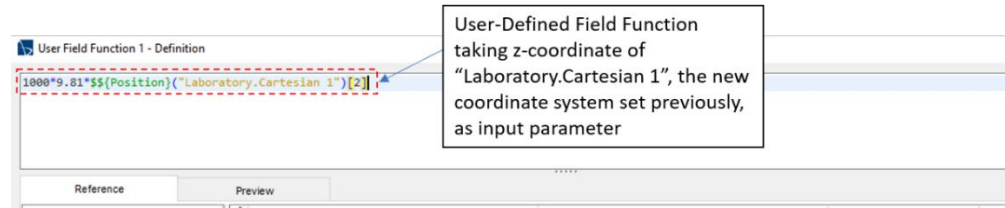


Figure 22. Setup of User-Defined Field Function in StarCCM+

Finally, hydrostatic gauge pressure boundary condition at all side boundaries is set using the above User-Defined Field Function. Refer to Appendix I for more details. This pressure outlet boundary conditions will be applicable to all subsequent sub-projects. For initial condition, fluid velocity in the domain is set as 0 m/s everywhere and initial pressure is set based on the hydrostatic pressure in the domain.

B. DFBI Motion Setup

Here, the author activates the DFBI Translation and Rotation Motion and attaches it to the overset region. The DFBI body is specified as the AUV. Furthermore, the mass of the AUV is specified to be 2 kg, while diagonal terms of mass moment of inertia is specified as $I_{yy} = 0.439 \text{ kgm}^2$; $I_{xx} = 3.210 \text{ kgm}^2$ and $I_{zz} = 3.210 \text{ kgm}^2$. For the AUV motion specification, free motion option in all axes (X, Y, Z, Roll, Pitch, Yaw) is activated. The AUV is also subjected to fluid forces, gravitational forces and thrust by the virtual disk.

C. Virtual Disk Setup

The setup of virtual disk propeller model is potted over from the sub-project 3.1. For this case, the author places the virtual disk at the aft of the AUV.

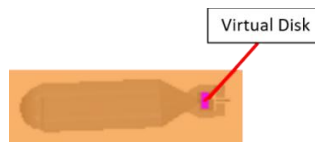


Figure 23. Position of Virtual Disk in AUV Body

D. Turbulence Model Selection

The turbulence models for this study are all potted over from sub-project 3.2.

3.3.3. Results and Discussion

In general, the author has successfully made use of thrust generated by the virtual disk to set the AUV in motion. Firstly, from velocity magnitude plots ([Animation 3](#)), it is seen how virtual disk has created a stream of high velocity fluid aft of the AUV, pushing the AUV forward. In contrast, velocity far away from the AUV is minimal showing that the undisturbed far-field. Cut-out progress velocity magnitude plot is as follows:

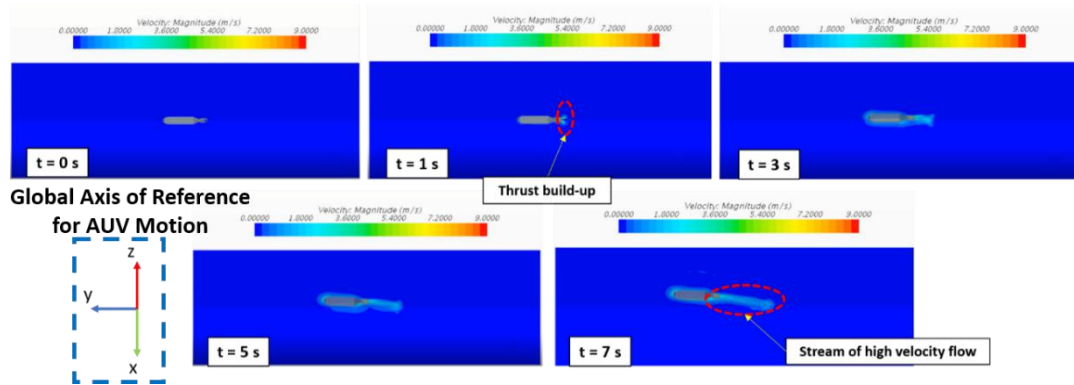
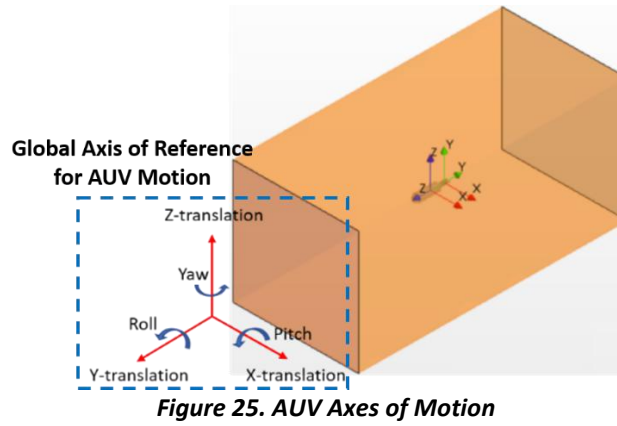


Figure 24. Progress of Velocity Magnitude Contour Plots

Similar progress plots of the pressure ([Animation 4](#)) and vorticity magnitude ([Animation 5](#)) can be seen in Appendix J. Also, from the thrust monitor plot in Appendix K, it can be seen that the virtual disk provides a constant amount of 100 N thrust throughout the run. Furthermore, it can be observed that the wall y^+ monitor shows values much smaller than 1 on most of the AUV body used for this sub-project (Refer to Appendix L), which further substantiates that mesh generated is sufficient to capture the near-wall effect, in tandem with turbulence model and wall treatment applied.

Other than the above results, the author also attempts to track other parameters which are important to control the AUV. This includes: movement of AUV in X, Y, Z, Roll (Y-axis rotation), Pitch (X-axis rotation) and Yaw (Z-axis rotation) axes, linear velocity in

X, Y and Z axes as well as angular velocity in the Roll, Pitch and Yaw axes. Recall that the placement of the AUV and the coordinate system is as follows:



For the free 6-DOF simulation run, the timeplots of translation of AUV in X, Y and Z axes can be found below (Refer to Appendix M for other control parameters tracked)

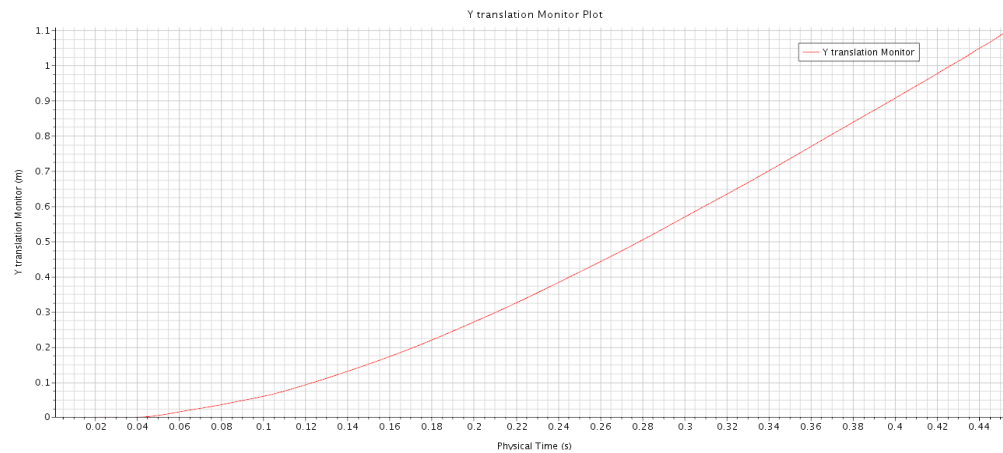


Figure 26. Y-translation of AUV Time Plot

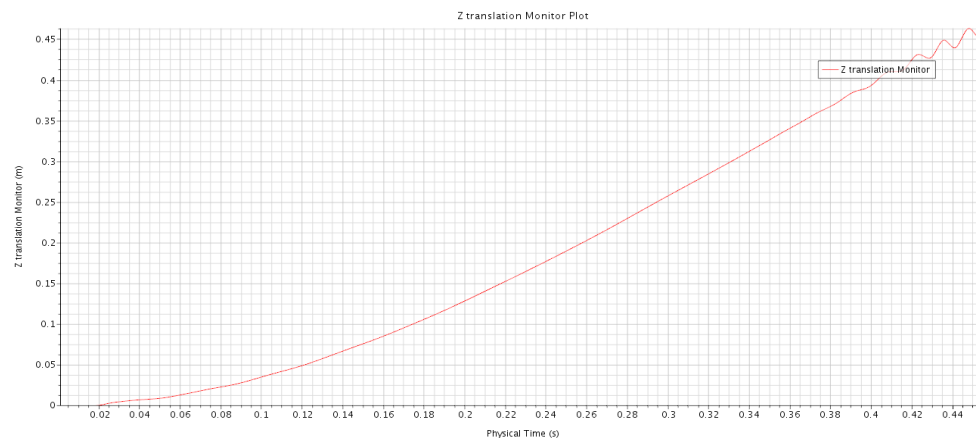


Figure 27. Z-translation of AUV Time Plot

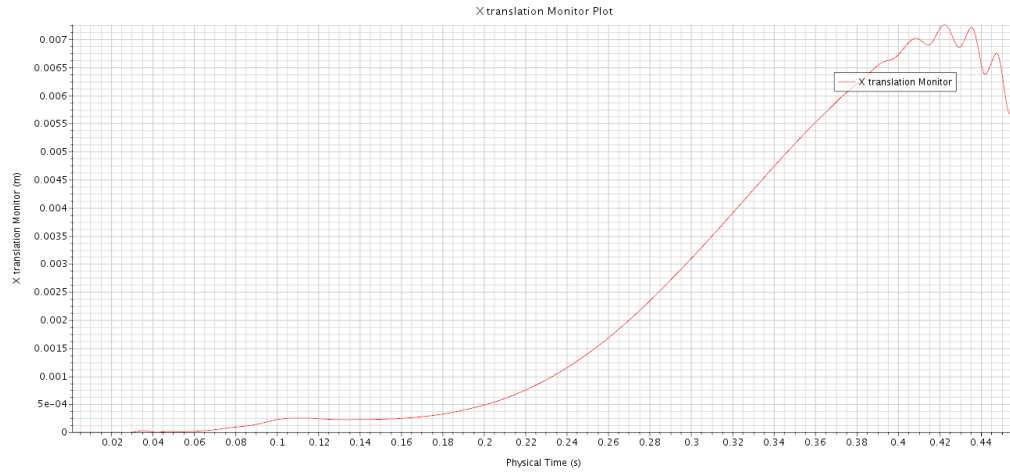


Figure 28. X-translation of AUV Time Plot

3.4. Modeling of 6-DOF Free Motion of AUV with Actual Spinning Propeller

3.4.1. AUV Propeller Verification

Before moving into this sub-project, the author attempts to validate his method of simulating a spinning propeller. The propeller geometry used for this sub-project comes from a commercial underwater thruster, T200, by Blue Robotics [19]. The author chooses this propeller because Blue Robotics provides comprehensive performance specifications of its thruster. Moreover, the author also has access to a T200 thruster set to perform experimental benchmarking to obtain the actual thrust vs rpm values for the T200 thruster. He will then compare these experimental values with simulated values to verify that methods done to simulate the propeller motion is good enough to be adapted to the AUV simulation using the spinning propeller.

Based on Blue Robotics performance specifications (Appendix N), T200 thrusters can be operated at rotational speed ranging from 300-3800 rpm. The benchmarking is done on a custom-designed thrust measurement jig (Fig. 29), using a load cell that outputs pulling force to obtain thrust values. For this experiment, the thruster is run at different

speeds and the thrust values are recorded accordingly. The tabulated result from the experimental benchmarking can be found in Appendix O.

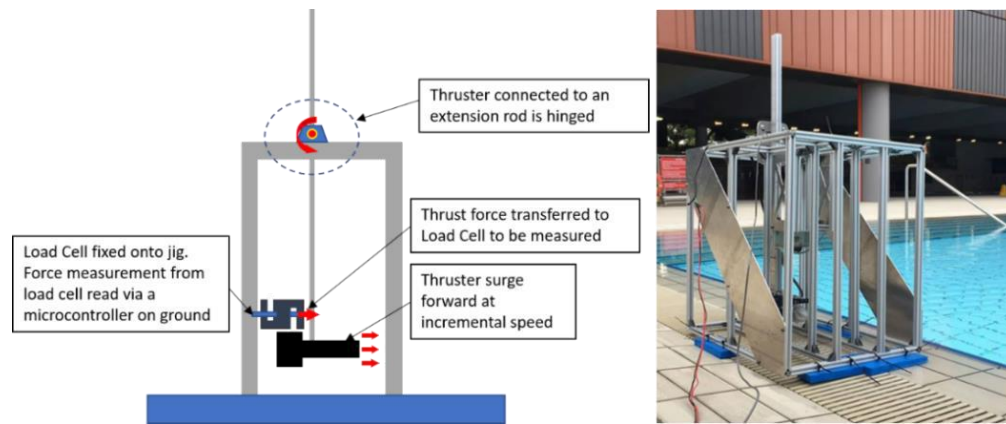


Figure 29. Thrust Measurement Jig

Separately, the author performs CFD simulations to the simplified T200 thruster geometry to measure its thrust values at specified speeds; the geometry includes the actual propeller and the thruster shroud simplified into having constant diameter. While it may affect accuracy of results, the simplified geometry reduces complexity of simulation, especially knowing there are a string of simulation to run through.

For the simulation runs, the thruster geometry is placed in the center of an elongated fluid domain. To discretize the fluid domain, polyhedral cells are used with volumetric refinement done along the region close to the thruster geometry. Also, the author uses the overset meshing technique where the fluid domain is separated into two regions: background region comprising the fluid domain with the shroud geometry and overset region enclosing only the propeller geometry. This allows for the actual rotational motion of the propeller to generate the thrust values to be recorded. Since the range of speed at which thrust values are recorded is known (300 to 3800 rpm), the minimum mesh size is set such that the distance moved by the propeller tip spinning at 3800 rpm at every timestep is less than the size of the minimum mesh to ensure accurate

measurement of the values. Other than mesh size, timestep selection has also been factored in for this consideration. Refer to Appendix P for the mesh illustration. For boundary condition, the propeller is assumed to be dipped in initially stationary shallow water. Hence, all boundaries are set to be pressure outlet with approx. atmospheric pressure. For initial condition, velocity is set as 0 m/s everywhere. Subsequently, the propeller blade overset region is set to spin at different speeds ranging from 300 to 3800 rpm. Approximately steady state thrust value is then recorded for simulation run at each speed. Then, the thrust values are plotted against rotational speed (rpm) and juxtaposed with the thrust values obtained from the aforementioned experimental benchmarking. Refer to Appendix Q for full simulated results. In summary, the thrust values against rotational speed plot is as follows:

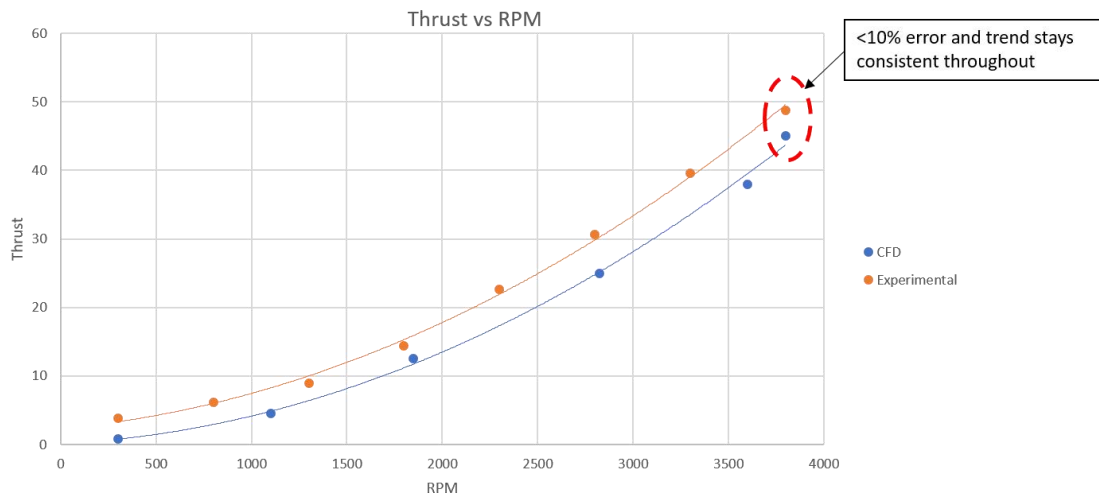


Figure 30. CFD and Experimental Thrust Values Comparison

There is a consistent less than of 10% of error between the actual experimentally benchmarked thrust values and the CFD results. While it is unclear to the author what may have caused the slight difference in values, this may have well been accounted for by the difference in the shape of the shrouds used in the simulation and the actual shroud. The author finds out that shape of shroud does affect the total thrust

generated by a thruster [20]. Hence, with this, it is deemed that method done to simulate the spinning propeller is sufficient to be adapted to the actual sub-project 3.4.

3.4.2. Problem Statement for Sub-project 4

Now that the author is satisfied with his method of simulating the spinning propeller, he will simulate the motion of AUV in an initially stationary flow field with a spinning propeller, which closely resembles an actual AUV propulsion system.

3.4.3. Problem Setup on StarCCM+

A. Fluid Domain Setup

The fluid domain setup follows the preceding two sub-projects closely where the same AUV is placed in the center of the elongated cuboid fluid domain. Likewise, the author uses overset mesh to separate the domain into three regions: background region, overset region enclosing the overall AUV geometry and overset region enclosing the propeller. Polyhedral cells with base size 0.5 m are used to discretize the fluid domain. Also, the mesh is refined within sections overlapping the background and the two overset regions. Together with the selection of timestep, the refinement is done with careful consideration such that at each timestep, propeller tip spinning at maximum possible speed does not move more than the size of one cell. The refinement has cell size of 0.0075 m. Finally, the fluid domain is broken down into 3.8 million cells.

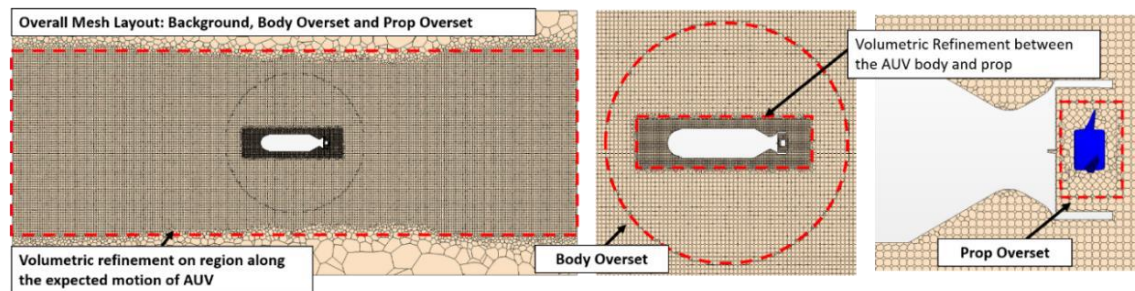


Figure 31. Section Plane Breakdown of Mesh

The boundary conditions for this sub-project is potted over from the previous sub-project, where hydrostatic gauge pressure at each boundary is considered; the reference pressure is set as absolute pressure at 20 m of depth. For initial conditions, initial velocity is set as 0 m/s everywhere to simulate initially stationary fluid field and initial pressure is set using the hydrostatic pressure of each point in the fluid domain.

B. DFBI Motion Setup

Similarly, the author activates DFBI Translation and Rotation Motion and attaches it to the overset region enclosing the AUV and the propeller. Additionally, DFBI Superposed Motion is attached to the overset region enclosing the propeller. The author understands that this model is typically used for rotating parts that will generate force onto the DFBI Body (the AUV). For this case, the propeller region is spun at a constant 3800 rpm, corresponding to the maximum speed of the propeller. The direction of spin is set such that it provides forward thrust for the AUV. Furthermore, details of the AUV (mass, moment of inertia, etc.) are set identical to the previous sub-project.

C. Turbulence Model Selection

Turbulence models for this sub-project are potted over from previous sub-project.

3.4.4. Results and Discussion

The author manages to simulate constantly spinning propeller ([Animation 6](#)) that provides the thrust to move the AUV, resembling an actual AUV propulsion system with spinning propeller. From the velocity magnitude contour plot ([Animation 7](#)) of the simulation run below, it is seen that the spinning propeller generates high velocity stream which propels the AUV forward. In contrary, fluid field far away from the AUV seems undisturbed.

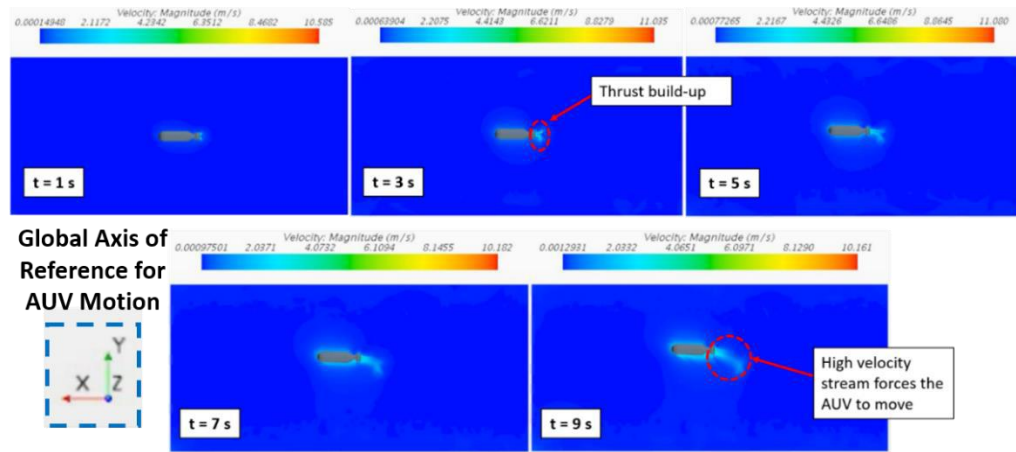


Figure 32. Progress of Velocity Magnitude Contour Plots

Figure 32 shows that, other than moving forward, the AUV also moves to the positive y-direction. The author believes that this is caused by the mass of the AUV that is purposely set to be low, relative to the buoyancy force that it is experiencing. This is done so that the resultant force in x-direction will lead to a large acceleration. Hence, in a short simulation time, substantial amount of motion can be captured. In fact, the author is not particularly interested in getting the AUV to follow certain motion path. Instead, this sub-project is set up to study the techniques used to simulate motion of an AUV acted by propulsive force generated by spinning propeller and whether it is able to capture different control parameters pertaining to the motion (linear/angular displacements and velocities) that are in sync to what is captured by the animated contour plot.

Other than only looking at velocity magnitude contour plot, the author also looks at pressure and vorticity contour plots ([Animation 8](#) and [Animation 9](#)) that are available in Appendix R to observe that there is no serious anomaly occurring the fluid field near and faraway from the AUV. Lastly, as mentioned earlier, the simulation is also made to track the various AUV control parameters (i.e. linear/angular displacements and

velocities). The time plot for the x, y and z displacements can be seen below. For full results of the control parameters, please refer to Appendix S.

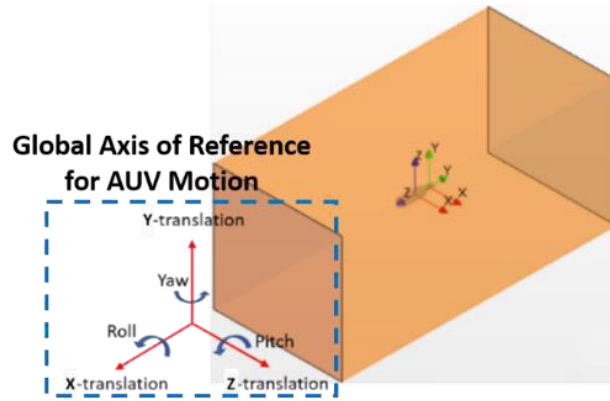


Figure 33. AUV Axes of Motion

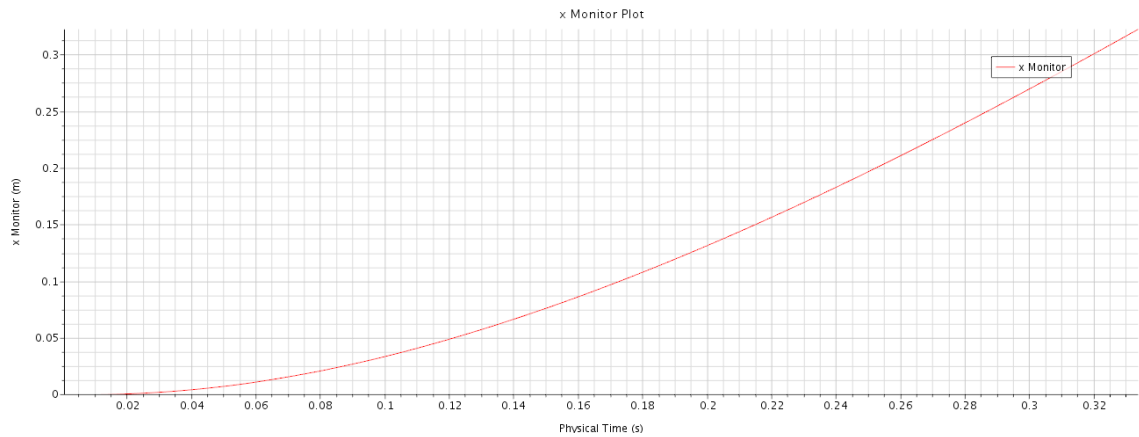


Figure 34. X-Translation of AUV Time Plot

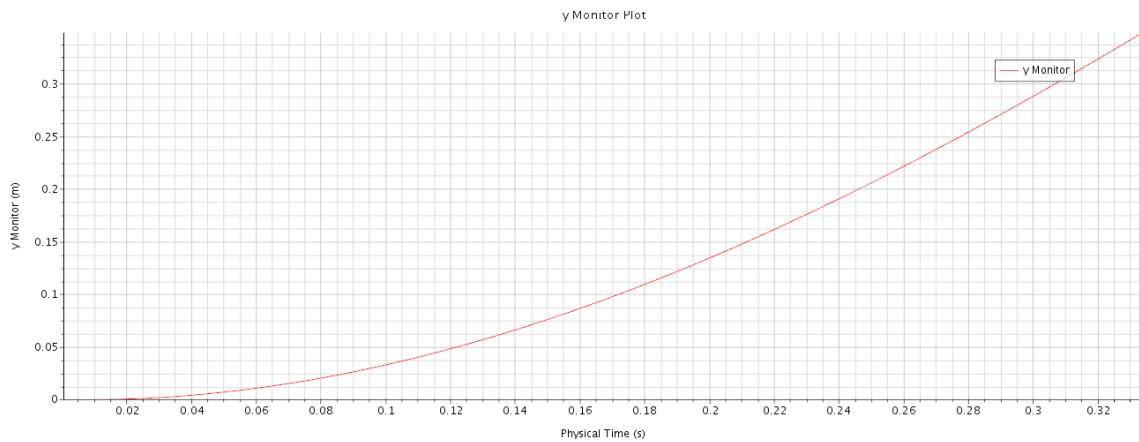


Figure 35. Y-translation of AUV Time Plot

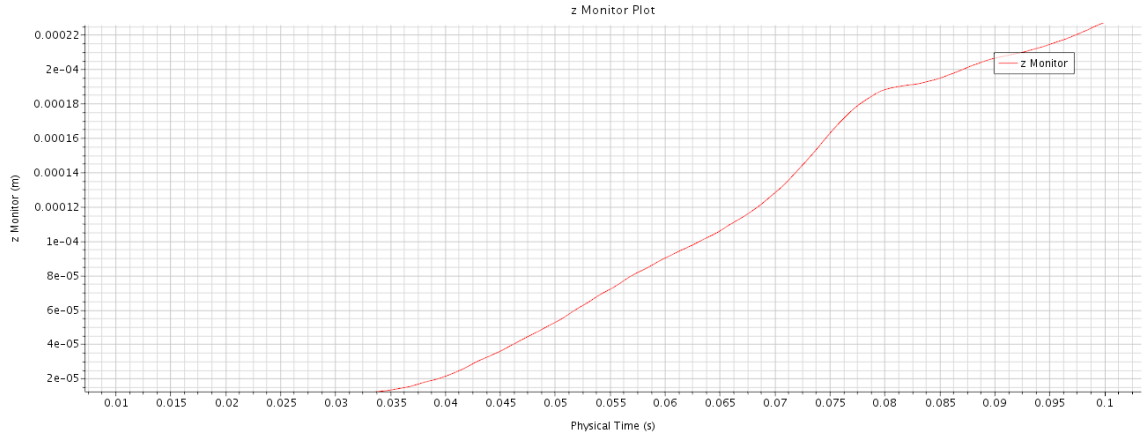


Figure 36. Z-translation of AUV Time Plot

3.4.5. Results Verification

There are two aspects of verification available for this sub-project. Firstly, the author looks at thrust value generated by the spinning propeller attached to the AUV. The thrust output for the simulation run is recorded in the form of thrust vs time graph below. The thrust recorded here, 37 N, seems slightly lower as compared to the thrust value of the propeller spinning at the same speed simulated in the verification step (i.e. 45 N). This may be accounted for by the difference in the size of shroud surrounding the propeller on AUV and the propeller used in section 3.4.1. Nonetheless, the slightly more than 15% difference in thrust values can still be deemed largely acceptable.

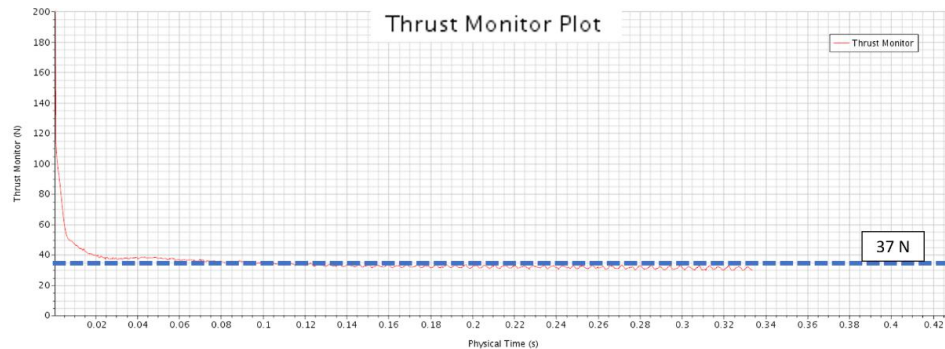


Figure 37. Thrust Value Monitor for AUV

Secondly, setup for this sub-project also closely resembles Singh's quadrotor aeromechanics and flight control project [9] discussed in Section 2.3. Firstly, Singh

divides his entire fluid domain into one background and a few overset regions; each propeller is enclosed with one cylindrical overset region and the quadrotor body with the four propellers are also enclosed by one spherical overset region:

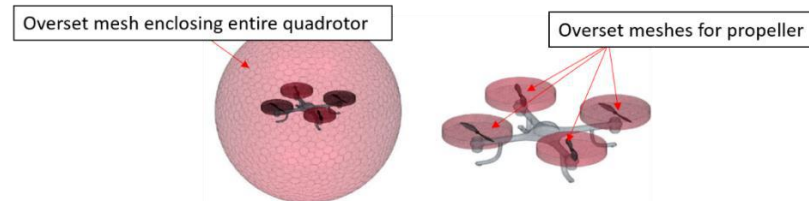


Figure 38. Singh's Quadrotor Overset Mesh System [9]

This is similar to the mesh system applied by the author in this sub-project to allow independent rotation of the propeller overset mesh. The propeller rotation will then apply forces onto the DFBI body – AUV in this sub-project and quadrotor in Singh's work – to generate motion. Refer to Appendix T, [Animation 10](#) or [Animation 11](#) to observe how Singh's quadrotor perform a vertical takeoff using the above mesh setup, which appears very similar to what is performed in this sub-project. Additionally, the average size of cells for domain discretization in both cases are also of similar order of magnitude² at dimensions approx. 0.02 m. Hence, although the author does not perform grid convergence study to find optimal mesh size due to time constraint, the comparison with a related past work suggests that the mesh size used for this sub-project, sub-projects 2 and 3 (sections 3.2 and 3.3, respectively) is of certain level of validity. Altogether, the close comparison of a few aspects of this sub-project with the propeller thrust verification section and Singh's published work help to verify the validity of results obtained as well as the methods used to derive the results in this sub-project.

² Cell size is volume - averaged with the number of cells in the fluid domain and volume of the domain.

3.5. Modeling of Controlled Motion of AUV using an Actual Spinning Propeller

In this sub-project, the author shall discuss his attempt to model single-axis controlled motion of AUV using a PID-controller interfaced with StarCCM+. However, due to time constraints, this sub-project shall only look at the implementation of the simulation without the actual results. For this part, the author assumes the same problem setup (fluid domain, meshing technique, boundary conditions, etc.) as the preceding sub-project 4. Hence, the AUV is placed in the centre of an elongated cuboid fluid domain of dimension 6 m by 2.8 m by 2.8 m. And, the AUV shall have a controlled single-axis motion in the Y-axis (i.e. free translation along Y-axis only, while no motion for other axes). In this case, the AUV will be set to move a fixed distance of 2 m forward controlled by the amount thrust acting on the AUV by the spinning propeller.

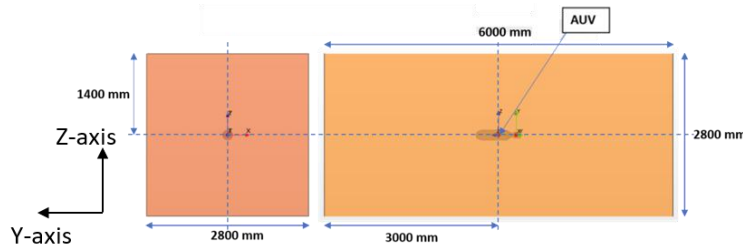


Figure 39. Fluid Domain and AUV Placement Illustration

In general, a PID controller can be written in the form of:

$$e(t) = X_{desired}(t) - X_{current}(t) \quad (36)$$

$$u(t) = Pe(t) + I \int_0^t e(t)dt + D \frac{de(t)}{dt} \quad (37)$$

where $e(t)$ is the discrepancy between desired state $X_{desired}(t)$ and current state $X_{current}(t)$, $u(t)$ is the control input and P , I and D are the proportional, integral and derivative gains of PID controller, respectively. The advantage of implementing such controller with StarCCM+ is that, StarCCM+ is able to help track $X_{current}(t)$ real-time

by solving the NS and rigid body dynamics equations concurrently and to constantly feed the $X_{current}(t)$ values back into the PID control loops.

Currently, the only parameter to track is Y-translation and the control input is thrust, which is affected by the speed of the propeller. Recall that speed-thrust relations of the propeller in forward spin configuration is given in Fig. 31. Using Excel, the author obtains the simplified speed-thrust relations for forward spin configuration as:

$$T_{forward} = 0.000003\omega^2 + 0.0014\omega + 0.1525 \quad (38)$$

where $T_{forward}$ is the thrust generated and ω is the rotational speed the propeller is to be set at. While yet to be done, the same speed-thrust relations for reverse spin configuration of the propeller can be obtained in an identical fashion, such that:

$$T_{reverse} = a\omega^2 + b\omega + c \quad (39)$$

These equations will be useful to translate the thrust required control input into rotational speed, which is the parameter that can be directly adjusted during the simulation run. From there, the author attempts to write the PID controller used to stabilize the Y-translation for the AUV as:

$$T_{required} = F_{drag}(t) + P(Y_d - Y(t)) + I \int_0^t (Y_d - Y(t))dt + D \frac{d(Y_d - Y(t))}{dt} \quad (40)$$

where $T_{required}$ is required thrust as the control input, $F_{drag}(t)$ is drag at that point in time, Y_d is desired Y-position that the AUV is supposed to travel to (i.e. 2 m forward), $Y(t)$ is current Y-position of the AUV and P, I and D are proportional, integral and derivative gains of the controller, respectively. Knowing $T_{required}$, the thruster rotational speed can be solved using either equation (37) or (38) depending on the sign of $T_{required}$ and be updated every timestep. The above method can be written using

Java code interfaced with StarCCM+ or using the User Field Functions available in StarCCM+. A schematic illustration of the above process is seen below:

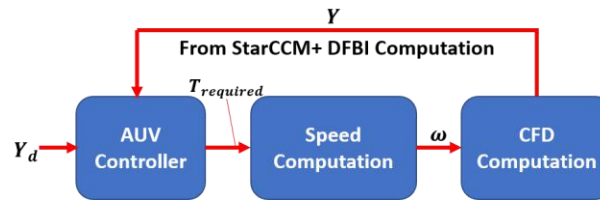


Figure 40. Signal Flow Schematic Illustration

4. Conclusion

In conclusion, the project looks at hydrodynamics of AUV by integrating CFD with 6-DOF dynamics available in StarCCM+ to study motion of an AUV exerted with thrust by spinning propeller. To reiterate, in this project, the author aims to gain exposure to the intricacies of CFD platform and to understand the techniques involved in integrating CFD with 6-DOF rigid body dynamics, and, eventually, control system for an AUV. The author is motivated to apply techniques learnt as a design tool for future generations of Bumblebee AUV (Fig. 1) that he has an experience building, previously.

Overall, the author has achieved his aims and objectives mentioned above. This is done through the five incremental sub-projects which the author has planned for himself. Nevertheless, the author also faces two major problems along the way.

Firstly, the author starts with a lack of basic understanding of CFD. This has made it difficult to grasp some of the underlying concepts of CFD, especially early in the project. Hence, progress turns out a little slow in the beginning. This is worsened by the fact that the author has to switch supervisor halfway through the project. This results in a lot of independent research and learning as well as trial-and-error to find out the best methods to solve the problems in the incremental sub-projects. Fortunately, things

turn out fairly well in the end as the author manages to achieve what he wants for the sub-projects, except sub-project 5 due to time constraints.

Secondly, the author has a lack in computational power to run the CFD simulations for the project. Initially, the author is given access to StarCCM+ in High Performance Computer resources in NUS, that allows parallel computing using up to 24 CPU cores. However, the StarCCM+ licenses expire two months into the project and do not get renewed. Hence, the author has to resort to borrowing licences from another professor to run the CFD simulations in the author's personal workstation using up to only 4 CPU cores. This slows down the project substantially because some CFD simulation runs can take up to 1 week to obtain comprehensive results. Nevertheless, the author perseveres and plans his time to efficiently run all the different CFD simulations required for each sub-project in order to stay on par with his peers.

That said, having grasped the basic understanding of CFD and the techniques involved to integrate CFD with rigid body dynamics and control laws from this project, the author is excited to apply his understanding to the multi-propeller Bumblebee AUV which he has designed and built, previously. This will be further discussed in the future work section following this, which the author has done some prior research on.

5. Future Work: Implementation of Controlled Motion with Multi-Propeller AUV

Moving forward, the ultimate scheme of things for this project is the actual implementation of a controlled full 6-DOF motion of a multi-propeller AUV that is representative of the Bumblebee AUV (Figure 1). While some framework has been established in the attempt to control the single-propeller moving AUV in a single-axis, more work needs to be implemented for multi-propeller AUV moving in all 6-DOF.

Firstly, the implementation shall involve establishing the flight dynamics of the AUV, which depends closely on how the propellers are placed in the AUV itself. The Bumblebee AUV has an 8-thruster configuration, such that forces and moments in each axis of motion can be obtained roughly as follows:

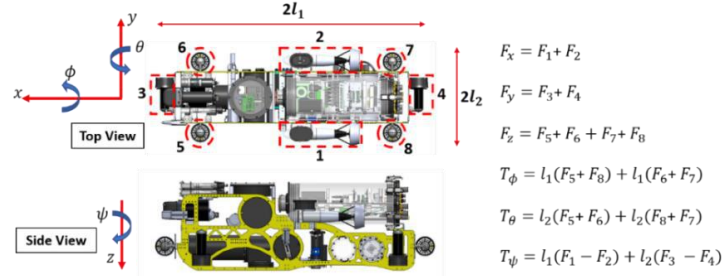


Figure 41. Forces and Moments on Bumblebee AUV

where F_x, F_y, F_z are accumulated forces in x, y and z directions, respectively, T_ϕ, T_θ, T_ψ are accumulated moments in roll (ϕ), pitch (θ) and yaw (ψ) axes, respectively, and F_1 to F_8 are forces generated by thrusters 1 to 8. The linear motion of the AUV can then be obtained by finding component of thrust along each inertial reference frame, done by transforming the linear forces attached to the body fixed frame to the inertial frame of reference, using the transformation matrix R_{x-y-z} [9].

$$R_{x-y-z} = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ \sin \theta \sin \phi \cos \psi - \cos \phi \sin \psi & \sin \theta \sin \phi \sin \psi + \cos \phi \cos \psi & \cos \theta \sin \phi \\ \sin \theta \cos \phi \cos \psi + \sin \phi \cos \psi & \sin \theta \cos \phi \sin \psi - \sin \phi \cos \psi & \cos \theta \cos \phi \end{bmatrix}$$

Figure 42. Transformation Matrix R

Comparatively, the angular motion can be obtained by interfacing Rolling, Pitching and Yawing torque with the mass moment of inertia tensor, available in Solidworks. Thus, after obtaining the angular motion equations, complete flight dynamics of the AUV is the combination of the linear motion and the angular motion equations.

Secondly, knowing the flight dynamics equation of the AUV, the implementation proceeds with the PID controller code in Java to be interfaced with StarCCM+.

However, notice that the Bumblebee AUV is redundantly-actuated having 8 thrusters and only 6-DOF. Essentially, this means that there are 8 unknowns (thrust force for each thruster) to be solved using 6 equations. One way to solve this is to separate the motion equations into two independent sets of equations: one set to control specific movement and one set to control the correction for AUV dynamics [21]. These sets of equations will help to solve for the thrust force required from each thruster for the AUV to move in a certain controlled manner; similar to the preceding sub-project, for this case, StarCCM+ will constantly provide the current states of the AUV to be used in the PID controllers. The thrust force can then be translated into propeller speed for each thruster that is used back as control input into StarCCM+. However, more research needs to be done before things can be further implemented here.

Thirdly, since not all of the thrusters is identical, careful profiling of the performance of the different sets of propellers is also required. Onboard the Bumblebee AUV, the two sets of thrusters include 2 units of Videoray Thrusters and 6 units of Seabotix BTD150 Thrusters. For each set of thruster, thrust versus propeller speed curves will need to be obtained to translate the thrust force calculated from the controller into actual control input to the spinning propeller in StarCCM+. This, essentially, will be very similar with what is done in the early phase of the 4th sub-project where the BlueRobotics T200 thruster propeller's performance is profiled.

Lastly, from the CFD simulation aspect, this implementation involves more complex overset mesh system while maintaining the concept adopted in the 4th sub-project. Since there are altogether eight independent propellers, each propeller will require its own cylindrical overset mesh. Additionally, the entire AUV body together with the eight

propellers will be enclosed by a spherical overset mesh. This spherical overset mesh will then have to interface with the background mesh, which represents the fluid domain. This overset mesh system can be illustrated as follows:

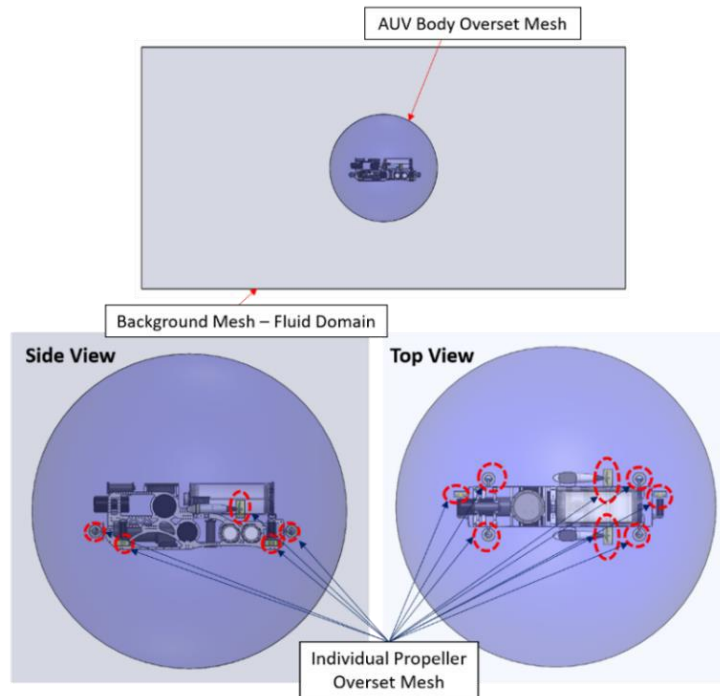


Figure 43. Overset Mesh System for Bumblebee AUV

Altogether, the above implementation can be a good platform to study the design of control system of the Bumblebee AUV where the designed control system can be simulated with the coupling of the approximately actual fluid domain and all its hydrodynamic forces. Combined with the CFD overset meshing method used in this project, the study of the control system will be quite comprehensive with behaviour of the AUV able to be tracked well, especially with the possibility of obtaining animated solutions on top of the graphical reports of motion of the AUV. Nevertheless, one possible compromise for this method is the computational power and time required to run the simulation. However, this should not be a problem given the availability of high performance computing resources in NUS.

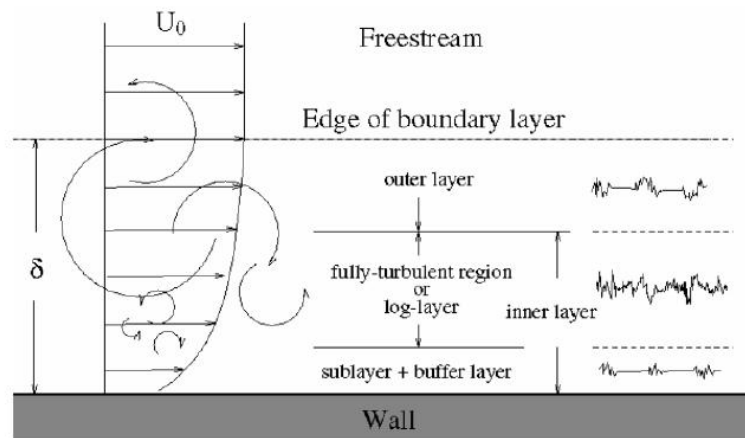
References

- [1] Bumblebee AS, "Bumblebee 3.0 Autonomous Underwater Vehicle," 2016. [Online]. Available: <http://bumblebee.sg/bbauv3/>.
- [2] D. Wojnar, *Underwater Drone*, GrabCAD, 2016.
- [3] R. G. Coe, "Improved Underwater Vehicle Control and Maneuvering Analysis with Computational Fluid Dynamics Simulations," Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2013.
- [4] B. Y. Kiam, H. C. Wai and Y. H. Wen , "Wageningen-B Marine Propeller Performance Characterization Through CFD," *Journal of Applied Science*, pp. 1215-1219, 2014.
- [5] S. Subhas, V. F. Saji, S. Ramakrishna and H. Das, "CFD Analysis of a Propeller Flow and Cavitation," *International Journal of Computer Applications*, pp. 27-33, 2012.
- [6] A. Mueller, *Fluid-Structure Interaction in Star-CCM+*, CD-adapco, 2013.
- [7] M. Peric, "Flow Simulation Using Control Volumes of Arbitrary Polyhedral Shape," *ERCOFTAC Bulletin*, vol. 62, 2004.
- [8] Symscape, "Polyhedral, Tetrahedral, and Hexahedral Mesh Comparison," 25 February 2013. [Online]. Available: <http://www.symscape.com/polyhedral-tetrahedral-hexahedral-mesh-comparison>.
- [9] S. S. Singh, "Computational Modelling of the Aeromechanics and Flight Control of a Quadrotor Unmanned Aerial Vehicle," Indian Institute of Technology Gandhinagar, 2016.
- [10] Engineering ToolBox, "The Engineering ToolBox," 2004. [Online]. Available: https://www.engineeringtoolbox.com/water-dynamic-kinematic-viscosity-d_596.html.
- [11] NASA, "Reynolds Number," 12 June 2014. [Online]. Available: <https://www.grc.nasa.gov/www/BGH/reynolds.html>.
- [12] General Dynamics Mission Systems, Inc., "Bluefin-21 Autonomous Underwater Vehicle (AUV)," 2017. [Online]. Available:

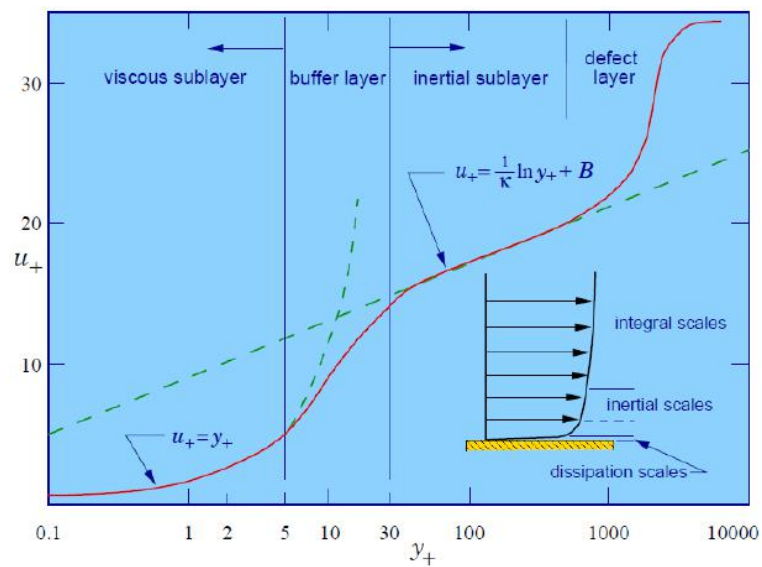
<https://gdmissionsystems.com/products/underwater-vehicles/bluefin-21-autonomous-underwater-vehicle>.

- [13] Kable Intel Ltd., "Bluefin-21 Autonomous Underwater Vehicle (AUV)," 2017. [Online]. Available: <http://www.naval-technology.com/projects/bluefin-21-autonomous-underwater-vehicle-auv/>.
- [14] Kongsberg, "Autonomous Underwater Vehicle (AUV) - The HUGIN Family," 2007.
- [15] Kongsberg, "Autonomous Underwater Vehicle, HUGIN," [Online]. Available: <https://www.km.kongsberg.com/ks/web/nokbg0240.nsf/AllWeb/B3F87A63D8E419E5C1256A68004E946C?OpenDocument>.
- [16] M. L. Weber-Shirk, *External Flows*, School of Civil and Environmental Engineering, Cornell University, 2018.
- [17] LearnCAx CCTech, "Basics of Y Plus, Boundary Layer and Wall Function in Turbulent Flows," 2016. [Online]. Available: <https://www.learncax.com/knowledge-base/blog/by-category/cfd/basics-of-y-plus-boundary-layer-and-wall-function-in-turbulent-flows>
- [18] CD-Adapco, User Guide Star-CCM+ Version 10.02, 2015.
- [19] BlueRobotics, "T200 Thruster," 2018. [Online]. Available: <https://www.bluerobotics.com/store/thrusters/t200-thruster/>.
- [20] J.-M. Laurens, S. Moyne and F. Deniset, "A BEM method for the hydrodynamic analysis of fishing boats propulsive systems," p. 7, May 2012.
- [21] Team Bumblebee, "Design and Implementation of Bumblebee AUV," 2015.
- [22] Q. H. Nagpurwala, *Propellers and Ducted Fans*, Bengaluru: M.S. Ramaiah School of Advanced Studies, Bengaluru, 2008.

Appendix A: Boundary Layers in Turbulent Flow [17]



Appendix B: Velocity Profiles in Turbulent Flow [17]



Appendix C: Details of Spalart Allmaras Models [18]

1. *Standard Spalart-Allmaras*

The standard form of Spalart Allmaras model is a low-Reynolds number model, which is applied without wall functions. This means that the entire boundary layer, including viscous sublayer, can be accurately determined and the model is best applied on fine meshes (small values of y^+).

2. *High-Reynolds number Spalart-Allmaras*

In contrast, this form of the Spalart-Allmaras model is only suited to coarse, wall-function-type meshes where y^+ values are more than 30.

3. *Spalart-Allmaras detached eddy model*

This Spalart-Allmaras model is used in unsteady simulation on top of the two aforementioned approaches. For this approach, the region closer to wall is dominated by RANS-based approach, while other high Reynolds number core turbulent region will be dominated by the LES approach.

Appendix D: Details of Wall Treatment in StarCCM+ [18]

1. *Low y^+ wall treatment*

This wall treatment implements the Spalart-Allmaras model and all the boundary conditions in low-Reynolds number form. This wall treatment is only recommended if mesh is known to be fine enough to resolve the viscous sublayer. For $y^+ > 1$, results will be inaccurate.

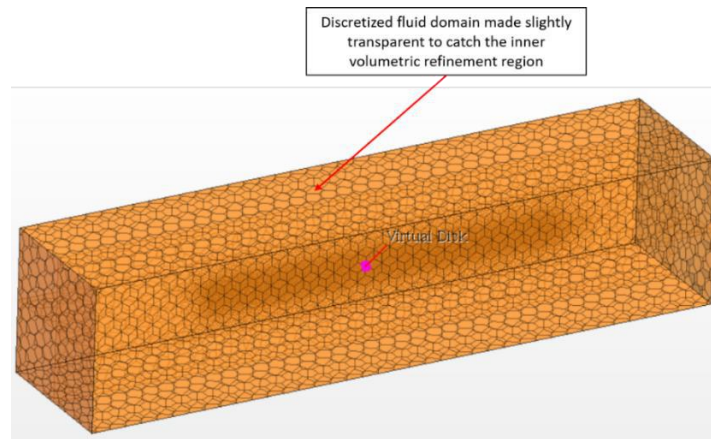
2. *High y^+ wall treatment*

This wall treatment is suitable for near-wall cell centroid that falls within the logarithmic region of the boundary layer and where wall-function approach is more applicable. This wall treatment is only applicable with High-Reynolds number Spalart-Allmaras model.

3. *All y^+ wall treatment*

This treatment is essentially a more practical hybrid model, especially when the near-wall cell centroid is in the buffer region. As a whole, this wall treatment will give results similar to the low y^+ treatment as y^+ approaches 0 and to the high y^+ treatment for y^+ goes beyond 30.

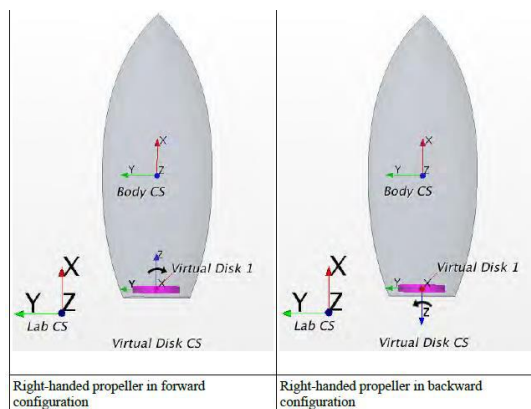
Appendix E: Isometric View of Discretized Fluid Domain from Section 3.1.2



Appendix F: StarCCM+ Userguide on Setting Direction of Thrust for Virtual Disk [18]

The following table and figures from StarCCM+ userguide are used as the main guidelines for the thrust direction specification:

Handedness	Normal	Propulsion Direction	Rotation Direction
Right	(1,0,0)	Forward	CCW
Right	(-1,0,0)	Backward	CCW
Left	(1,0,0)	Forward	CW
Left	(-1,0,0)	Backward	CW



Appendix G: Propeller Performance Data for Virtual Disk Setup [22]

J	KT	KQ	ETA
0	0.34	0.054	0
0.08	0.33	0.052	0.1
0.25	0.3	0.05	0.25
0.4	0.28	0.045	0.4
0.53	0.24	0.038	0.5
0.7	0.175	0.033	0.55
0.8	0.15	0.028	0.6
0.95	0.1	0.022	0.6
1.05	0.05	0.015	0.45
1.2	-0.025	0.01	0.1

Appendix H: Physics Model Selection for Section 3.2.2

Physics 1 Model Selection

Optional Models

- ☐ Co-Simulation
- ☐ Circumferential Heat Flux Averaging
- ☐ Electromagnetism
- ☐ Passive Scalar
- ☐ Turbulence Suppression
- ☐ Dispersed Multiphase
- ☐ Multiphase Interaction
- ☐ Energy-Based Time-Step Control
- ☐ Cell Quality Remediation
- ☐ Radiation
- ☐ Thermal Comfort
- ☐ Convective CFL Time-Step Control
- ☐ Thin Film
- ☐ Lagrangian Multiphase
- ☐ Aeroacoustics
- ☐ Vorticity Confinement Model
- ☐ Boiling

<Optional>

Enabled Models

- ☒ All y+ Wall Treatment
- ☒ Standard Spalart-Allmaras
- ☒ Spalart-Allmaras Turbulence
- ☒ Virtual Disk
- ☒ Gravity
- ☒ Coupled Energy
- ☒ IAPWS-IF97 (Water)
- ☒ Exact Wall Distance
- ☒ Reynolds-Averaged Navier-Stokes
- ☒ Turbulent
- ☒ Gradients
- ☒ Coupled Flow
- ☒ Liquid
- ☒ Implicit Unsteady
- ☒ Three Dimensional

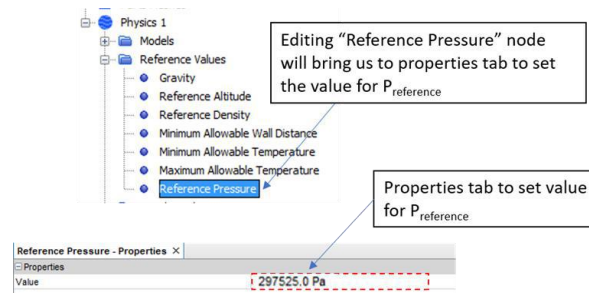
<Not required by other models>

<Not required by other models>

Appendix I: Specification of Pressure Values at Different Boundaries Details

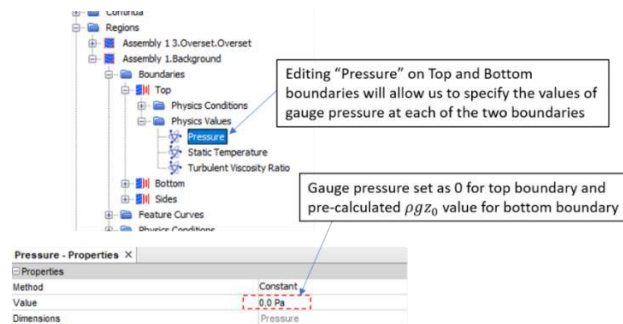
1. Setting of Reference Pressure

This can be done under “Continua > Physics 1 > Reference Pressure” node as seen below:

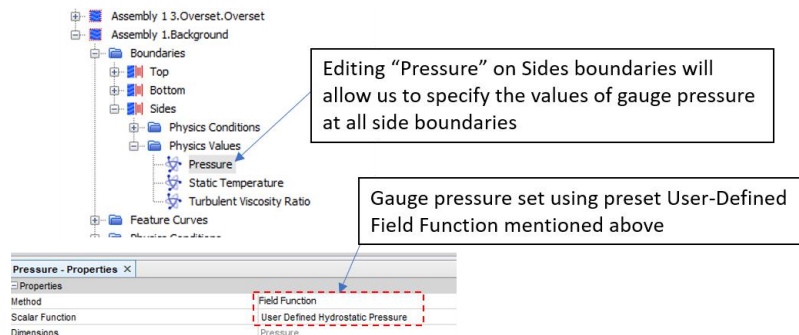


2. Specification of Constant Gauge Pressure

This can be done as follows:

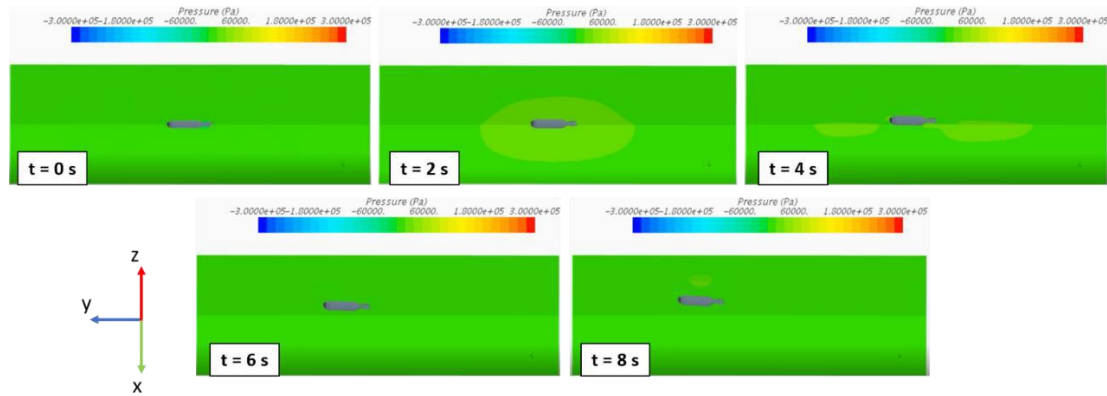


3. Specification of Gauge Pressure using User-Defined Field Function

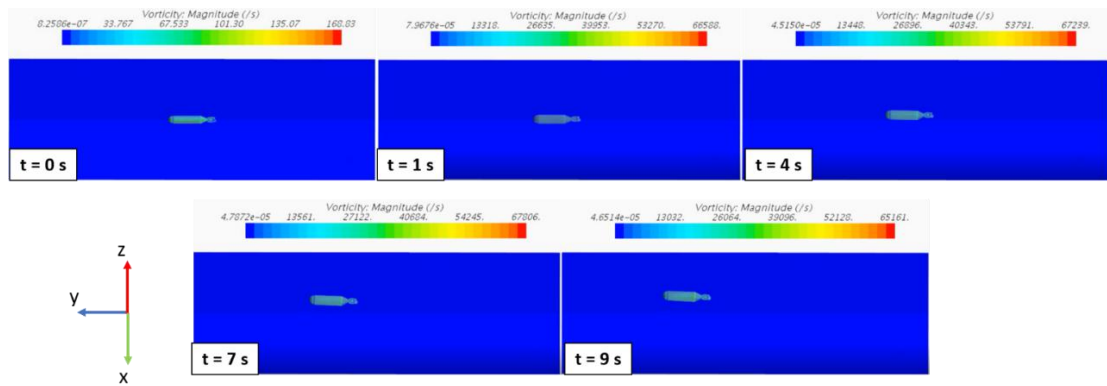


Appendix J: Pressure and Vorticity Plot for Section 3.3

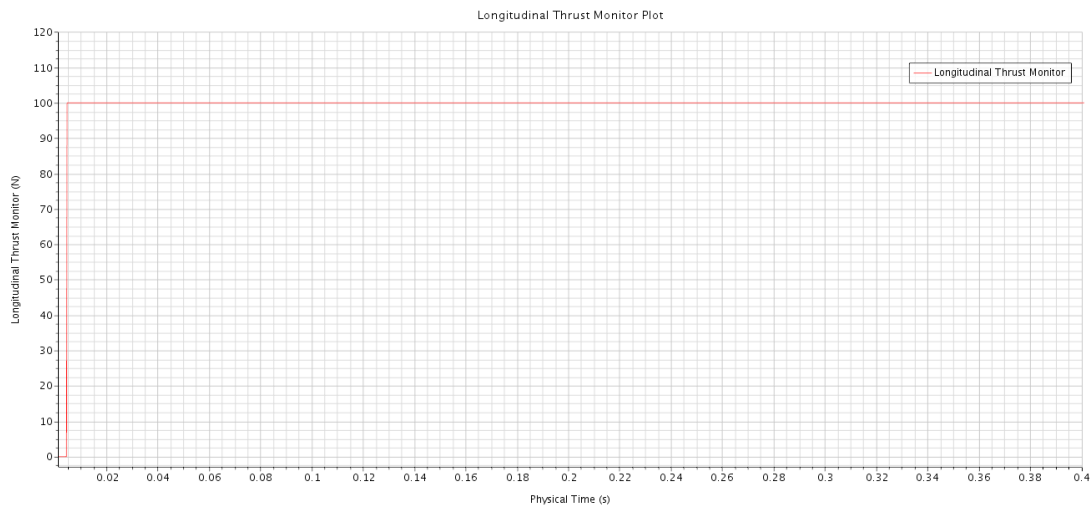
1. Pressure Progress Plot



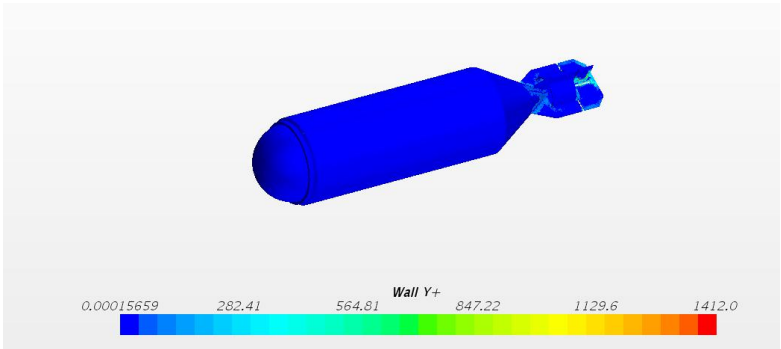
2. Vorticity Progress Plot



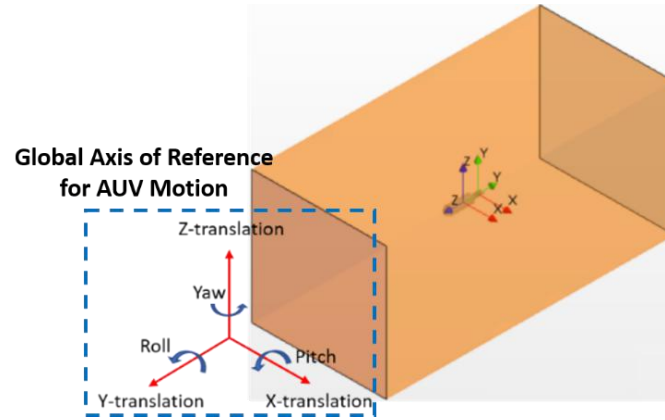
Appendix K: Thrust Monitor from Section 3.3.3



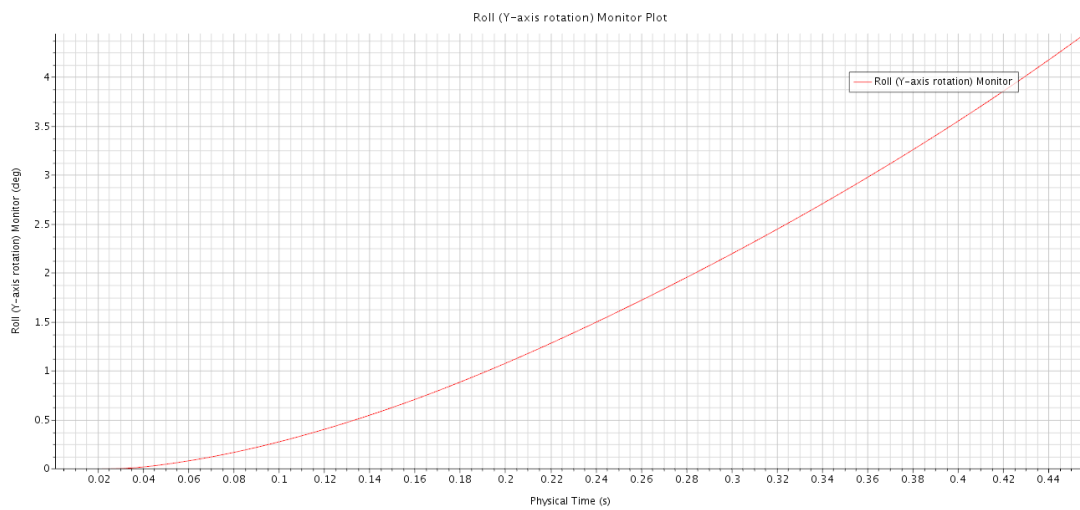
Appendix L: Wall Y+ Monitor for Section 3.3.3



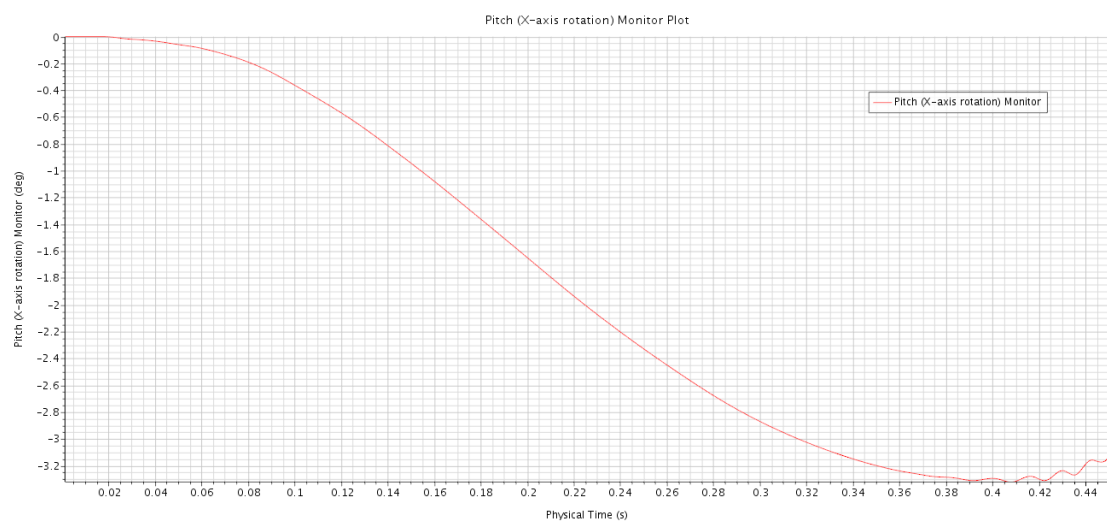
Appendix M: Control Parameters Tracked for Sub-project 3



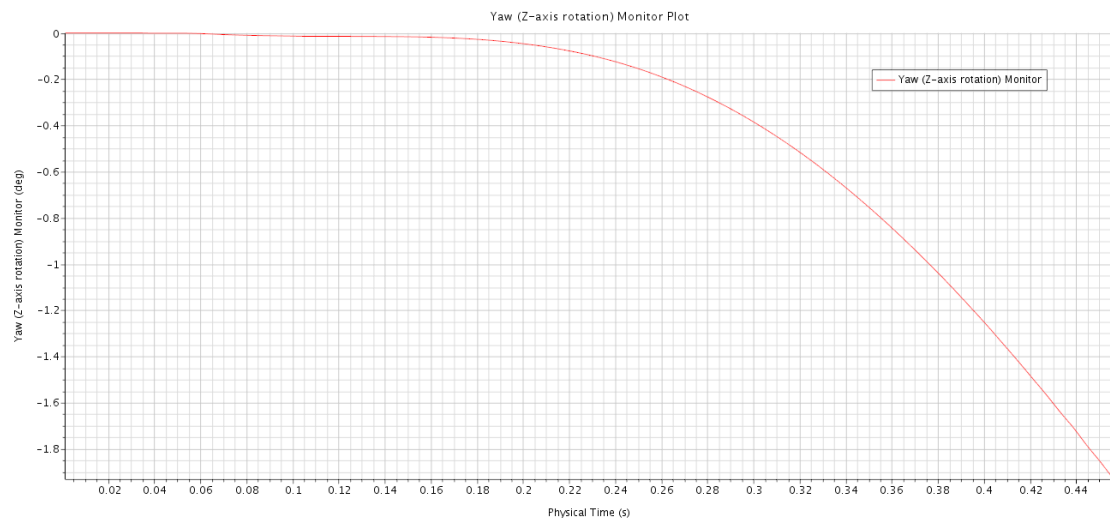
1. Roll (Y-axis rotation) of AUV Time Plot:



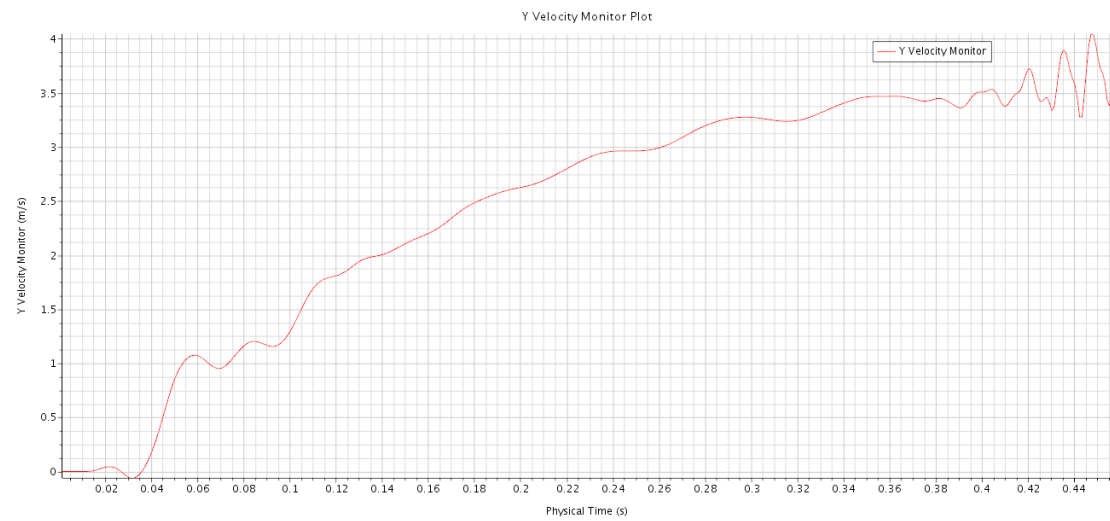
2. Pitch (X-axis rotation) of AUV Time Plot:



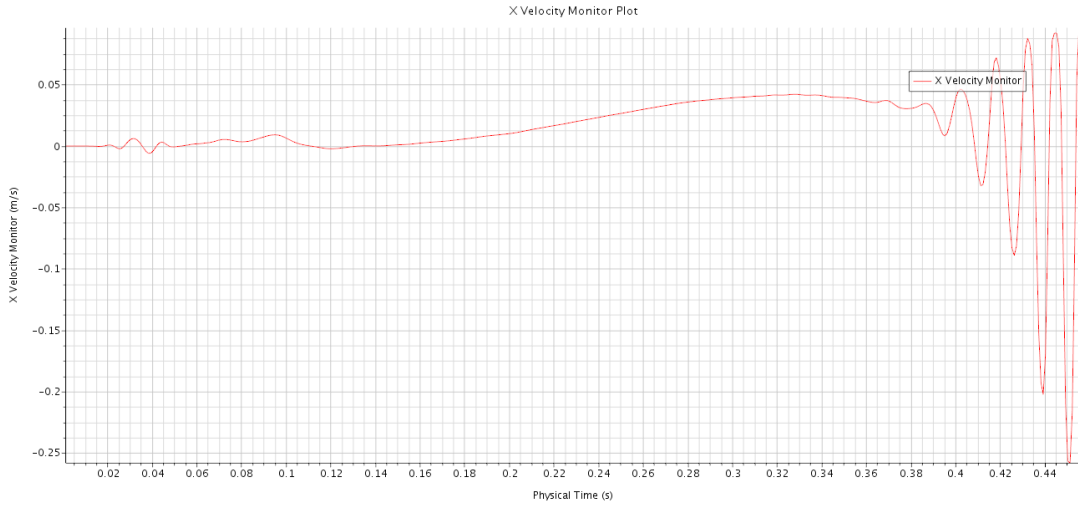
3. Yaw (Z-axis rotation) of AUV Time Plot



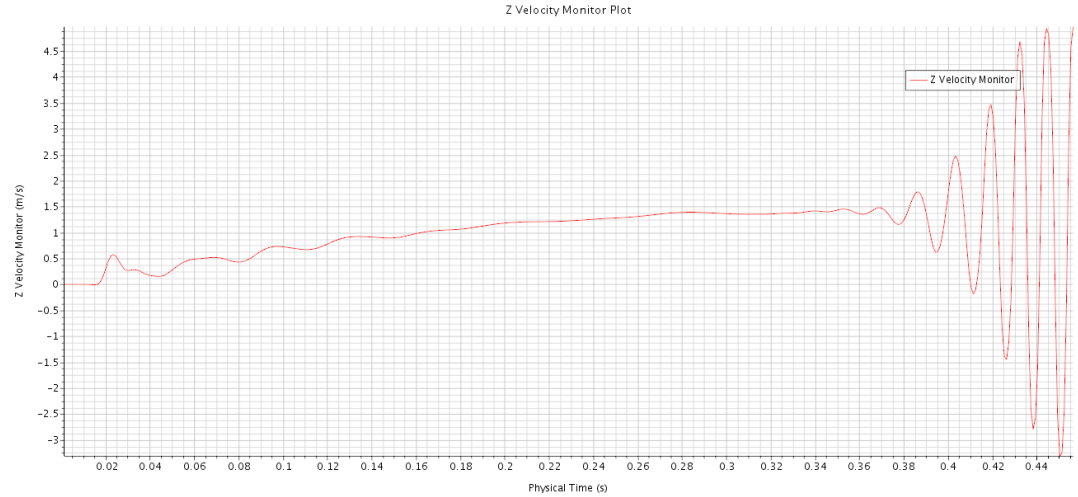
4. Y-Axis Linear Velocity of AUV Time Plot



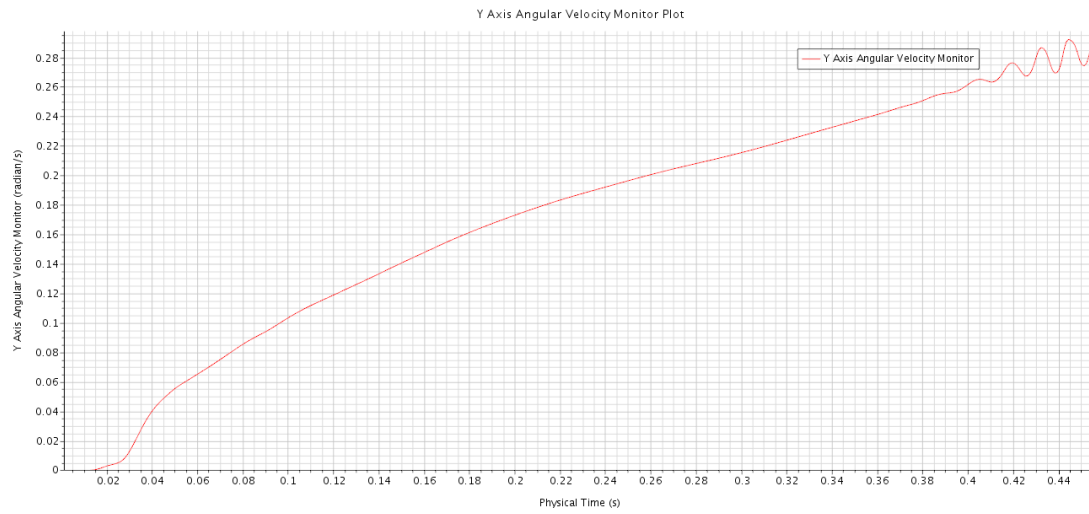
5. X-Axis Linear Velocity of AUV Time Plot



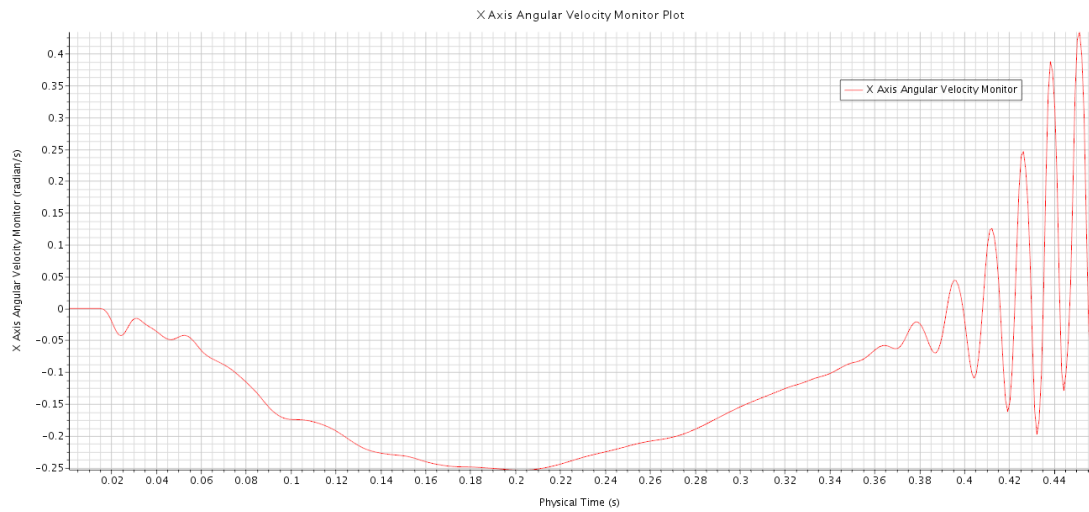
6. Z-Axis Linear Velocity of AUV Time Plot



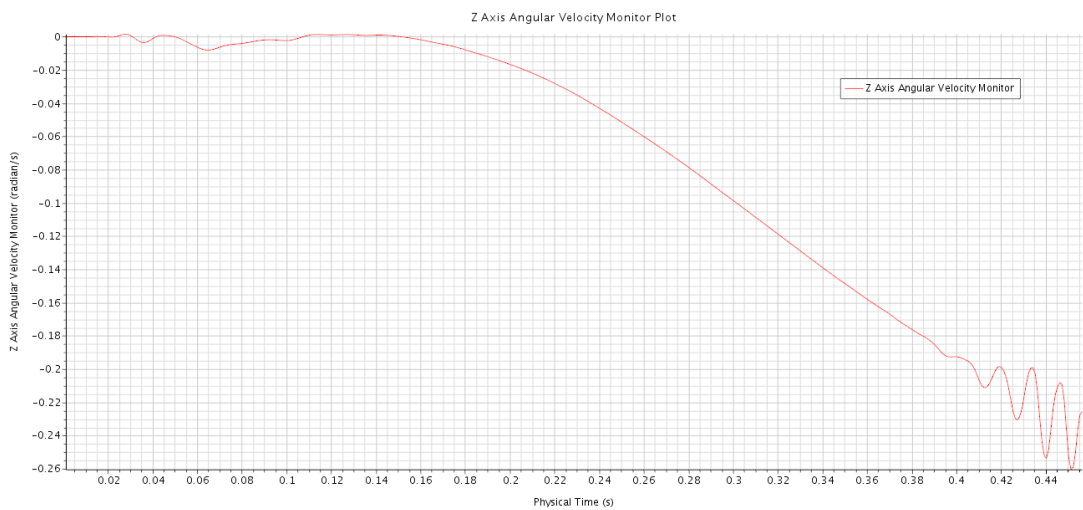
7. Y-Axis (Roll) Angular Velocity of AUV Time Plot



8. X-Axis (Pitch) Angular Velocity of AUV Time Plot



9. Z-Axis (Yaw) Angular Velocity of AUV Time Plot



Appendix N: T200 Thruster Specifications [19]



T200 Thruster Specifications

Specification Table

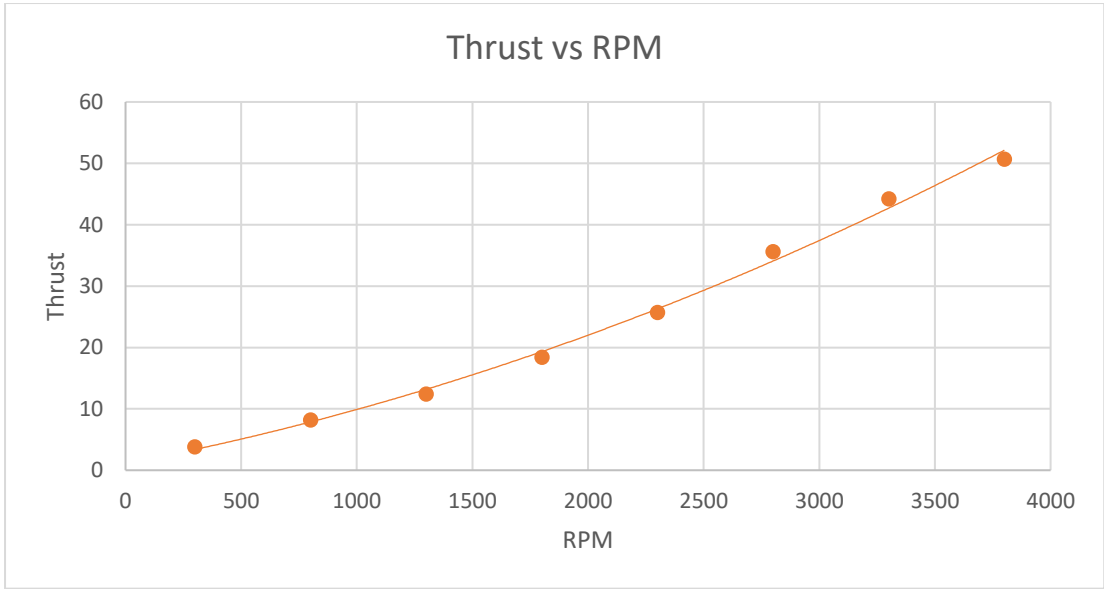
Performance		
Maximum Forward Thrust @ 16V	5.1 kgf	11.2 lbf
Maximum Reverse Thrust @ 16V	4.1 kgf	9.0 lbf
Maximum Forward Thrust @ 12V	3.55 kgf	7.8 lbf
Maximum Reverse Thrust @ 12V	3.0 kgf	6.6 lbf
Minimum Thrust	0.01 kgf	0.02 lbf
Rotational Speed	300-3800 rev/min	

Appendix O: Experimental Benchmarking of T200 Thruster Results

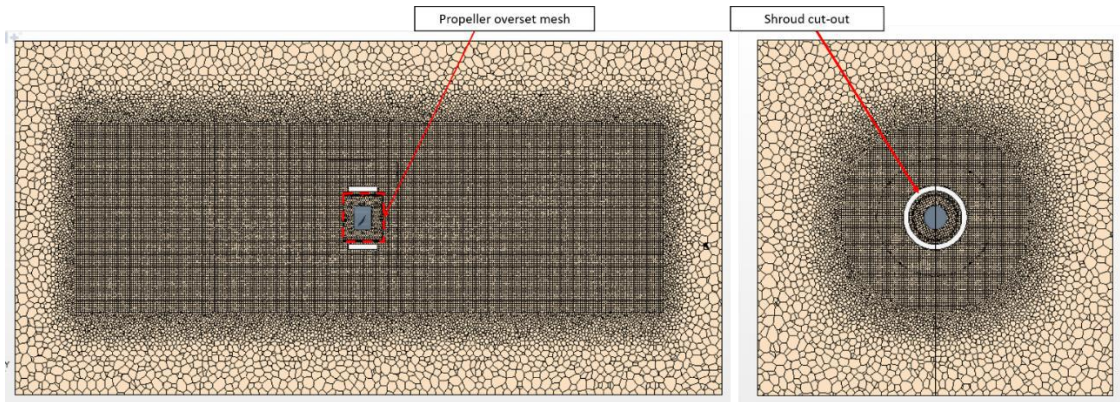
Item of interest

RPM	Voltage(V)	Current(A)	Power(W)	Force(kg)	Calibrated Force(kg)	Thrust (N)	Thrust/A	Thrust(g)/PWR(W)
300	16	0.35	5.6	-0.02	0.38	3.8	1.09	68
800	16	1	16	0.22	0.62	6.2	0.62	39
1300	16	2.3	36.8	0.5	0.9	9	0.39	24
1800	16	4.15	66.4	1.04	1.44	14.4	0.35	22
2300	16	6.7	107.2	1.87	2.27	22.7	0.34	21
2800	16	9.8	156.8	2.66	3.06	30.6	0.31	20
3300	16	13.5	216	3.56	3.96	39.6	0.29	18
3800	16	17.8	284.8	4.48	4.88	48.8	0.27	17

Item of interest

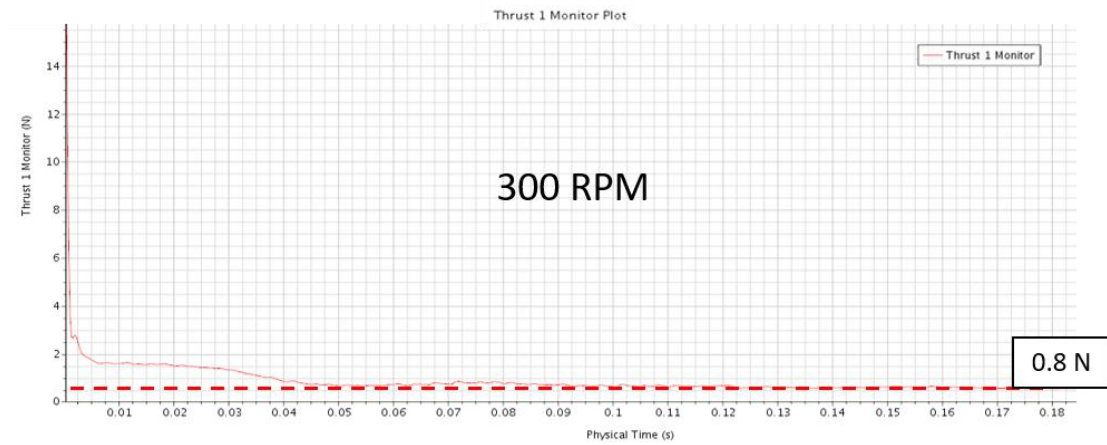


Appendix P: Mesh Visualization for CFD Thrust Values Verification

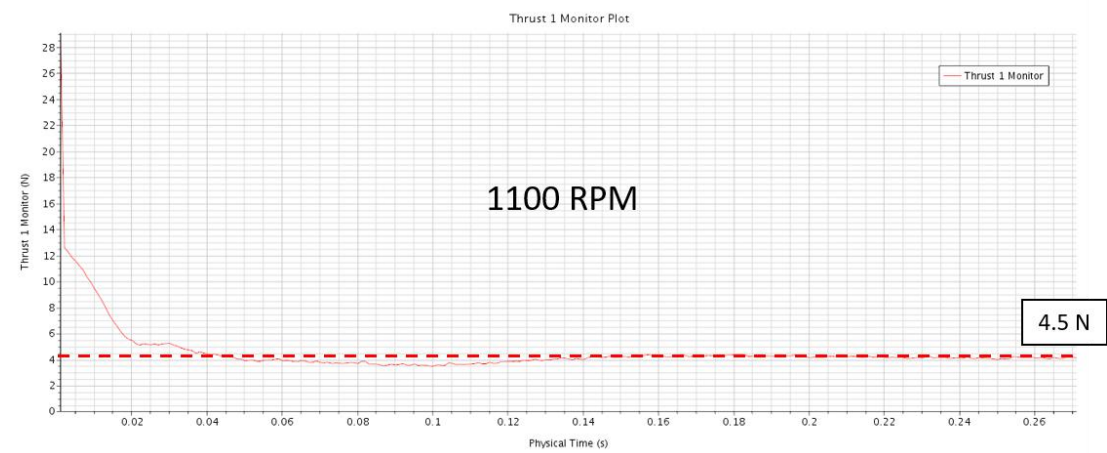


Appendix Q: Full Results for CFD Thrust Values Verification

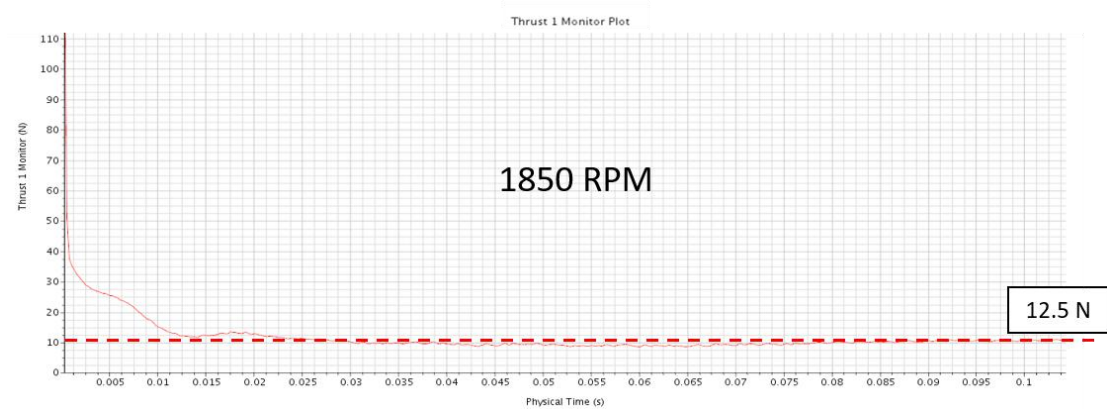
1. Thrust value at 300 RPM



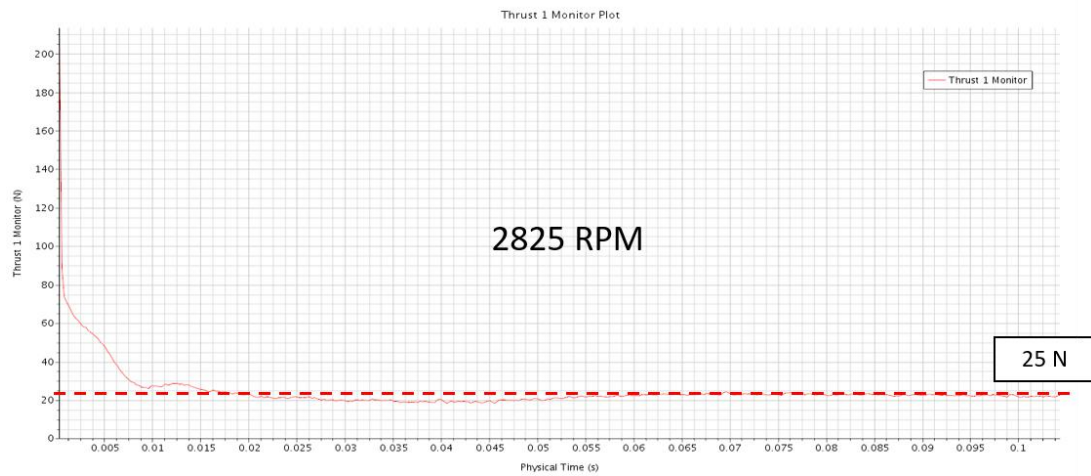
2. Thrust value at 1100 RPM



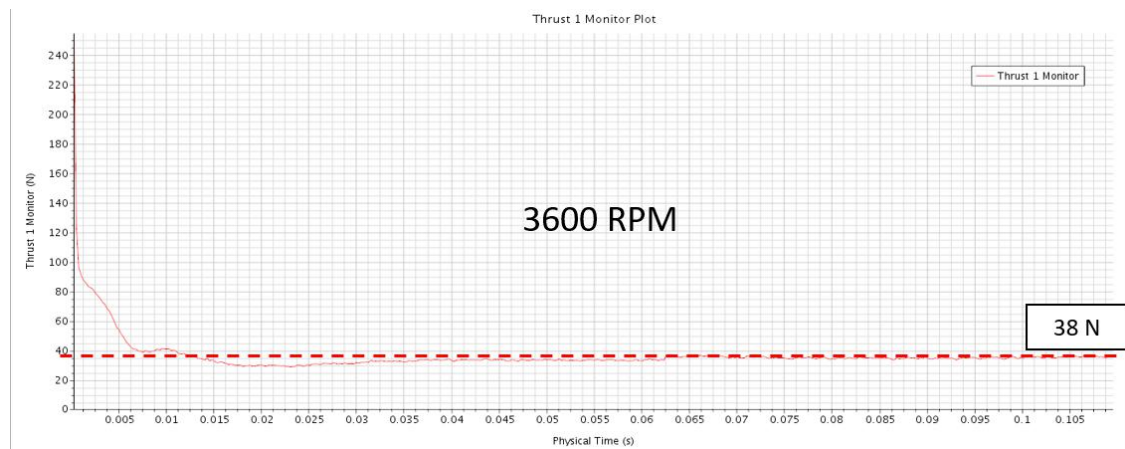
3. Thrust value at 1850 RPM



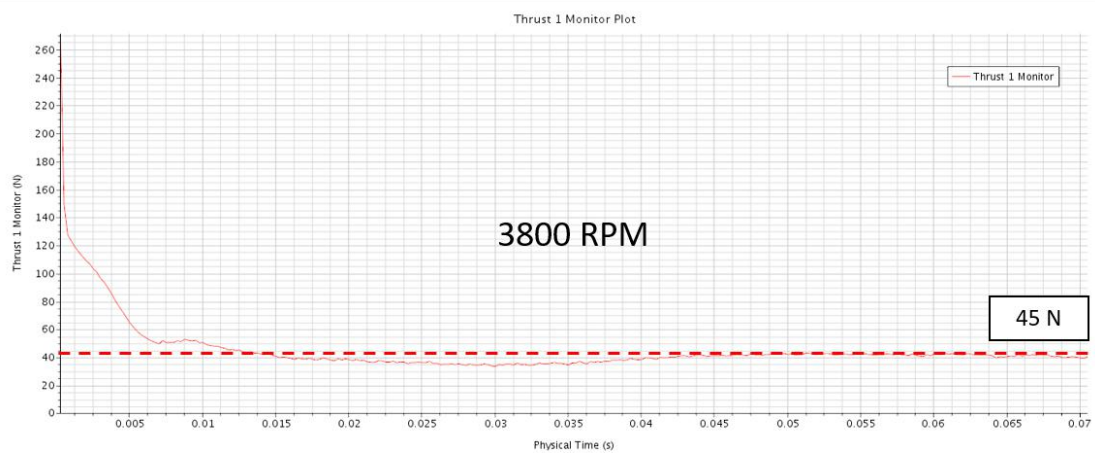
4. Thrust value at 2825 RPM



5. Thrust value at 3600 RPM

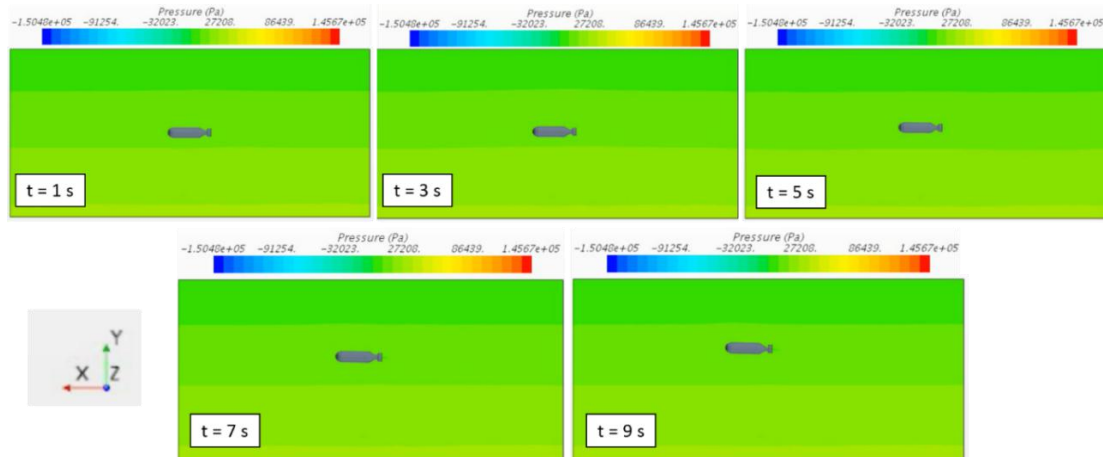


6. Thrust value at 3800 RPM

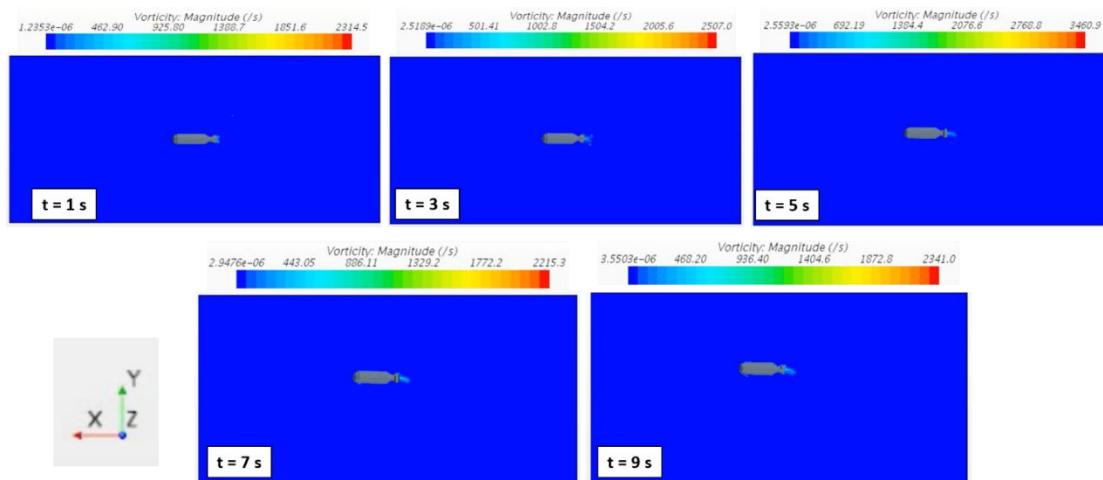


Appendix R: Pressure and Vorticity Contour Plots for Section 3.4.4

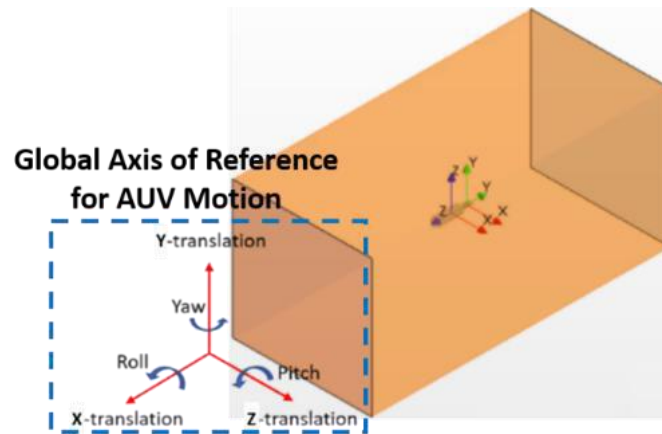
1. Pressure Contour



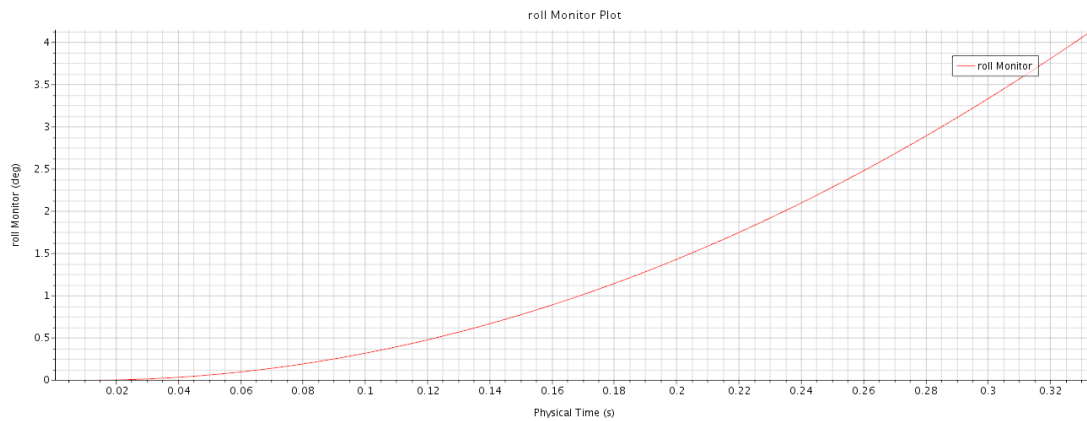
2. Vorticity Magnitude Contour



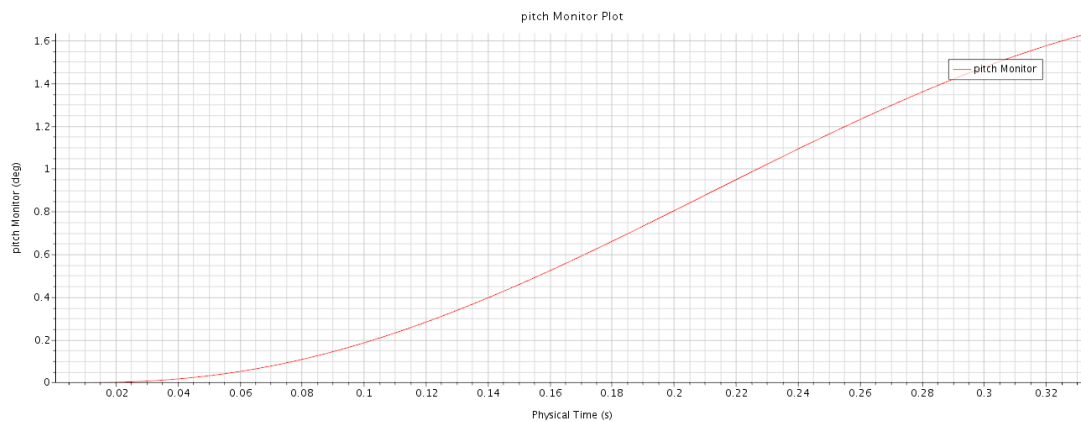
Appendix S: Control Parameters Tracked for Sub-project 4



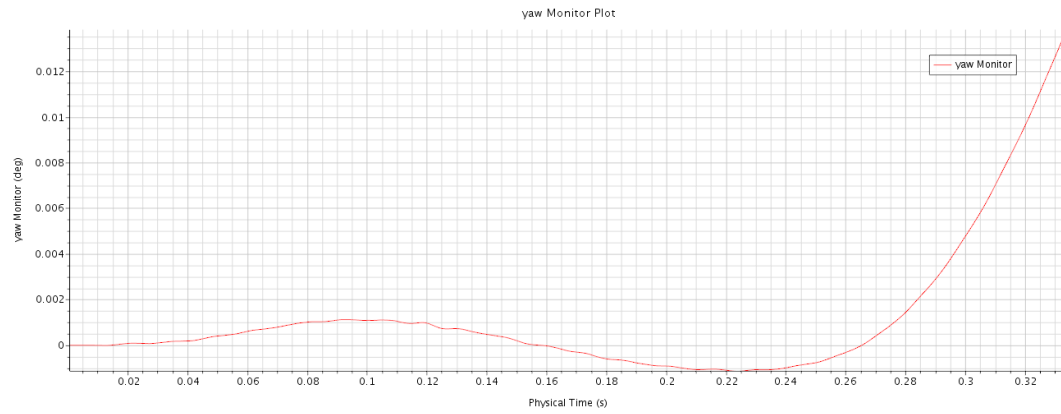
1. Roll (X-axis rotation) of AUV Time Plot:



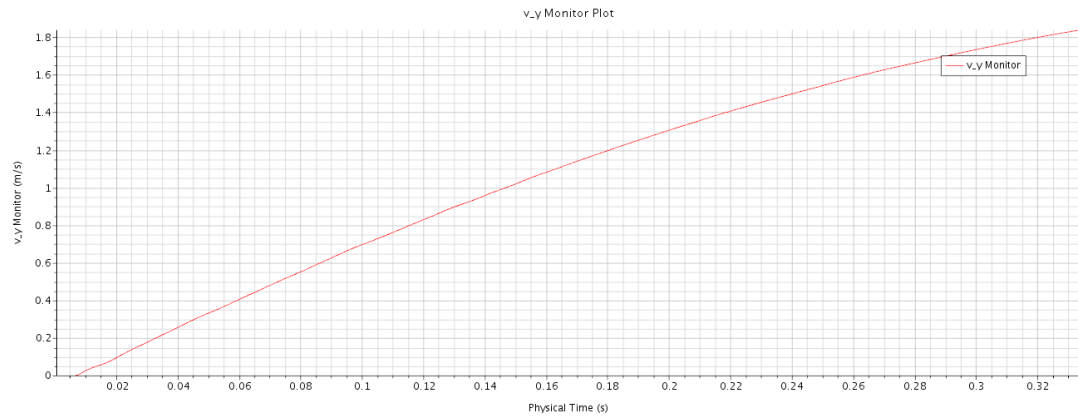
2. Pitch (Z-axis rotation) of AUV Time Plot:



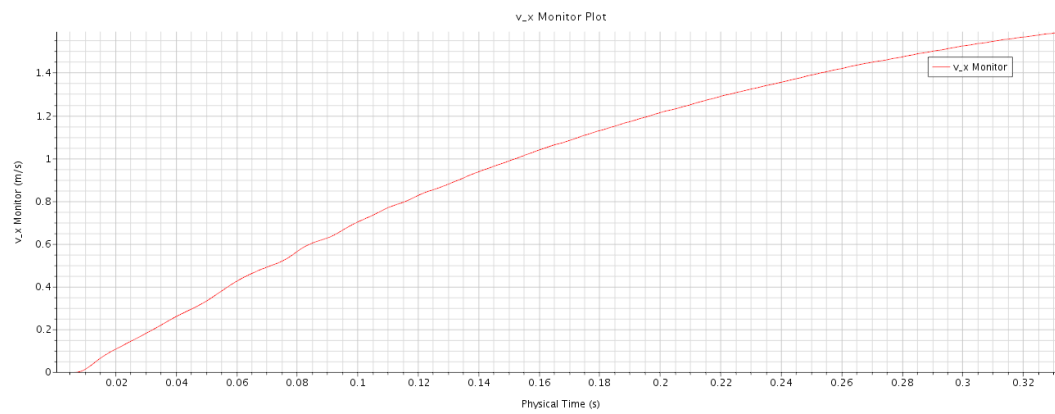
3. Yaw (Y-axis rotation) of AUV Time Plot



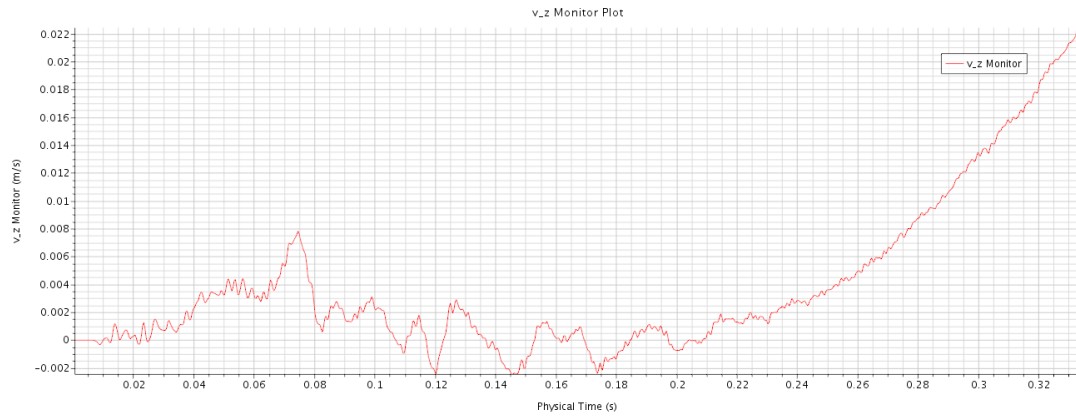
4. Y-Axis Linear Velocity of AUV Time Plot



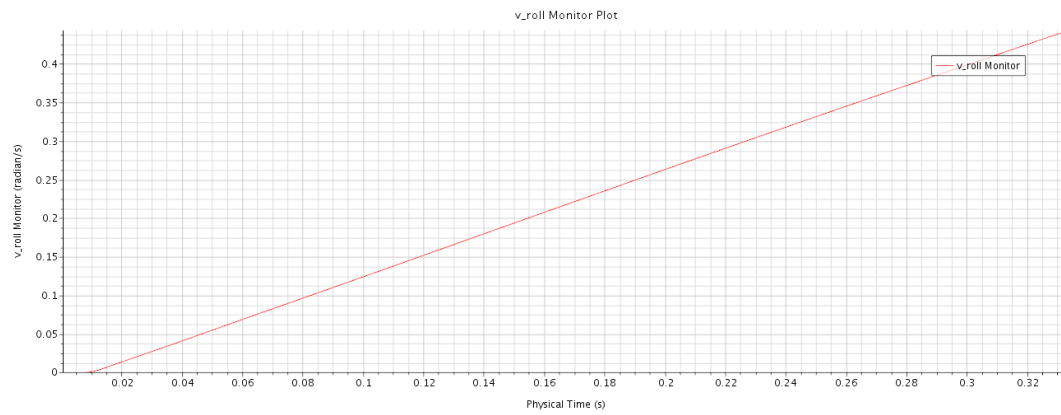
5. X-Axis Linear Velocity of AUV Time Plot



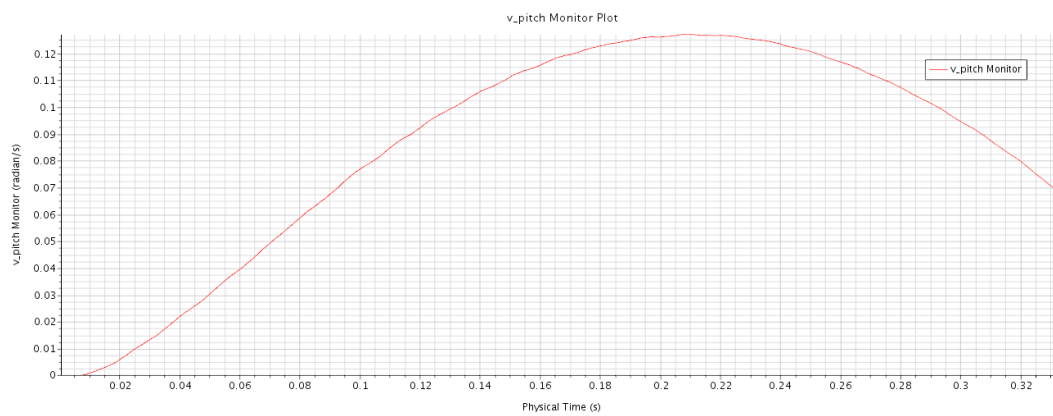
6. Z-Axis Linear Velocity of AUV Time Plot



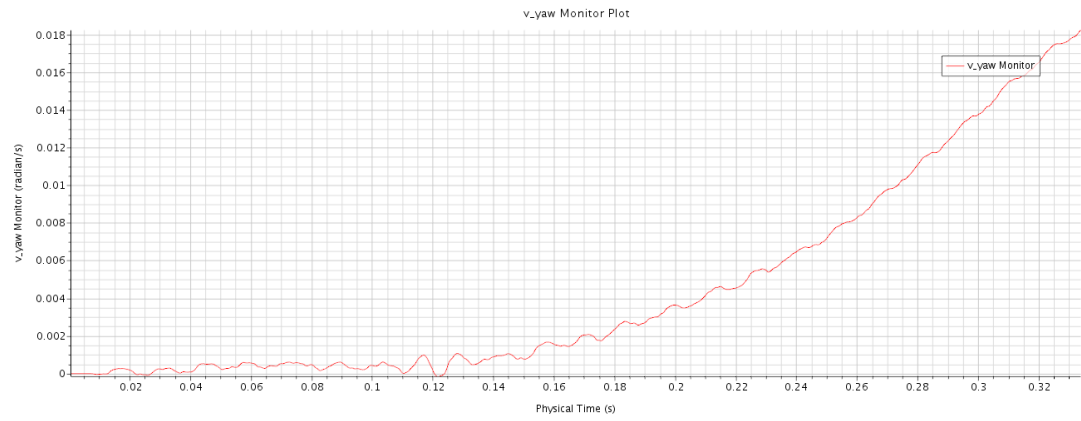
7. X-Axis (Roll) Angular Velocity of AUV Time Plot



8. Z-Axis (Pitch) Angular Velocity of AUV Time Plot

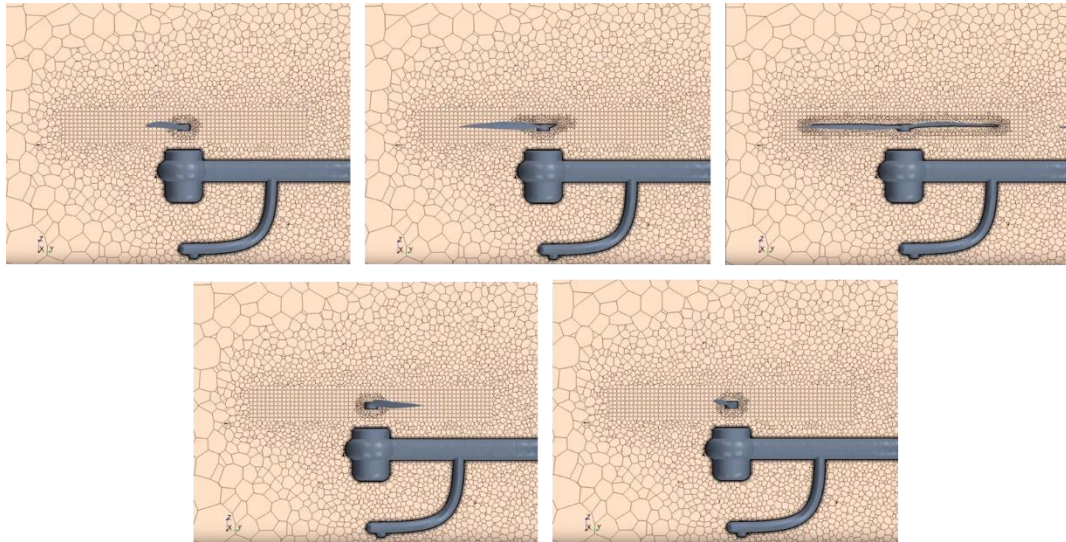


9. Y-Axis (Yaw) Angular Velocity of AUV Time Plot



Appendix T: Quadrotor Vertical Takeoff Using Overset Meshing Technique [9]

1. Rotor Overset Mesh Progress



2. Velocity Magnitude Progress Plot for Quadrotor in Vertical Takeoff

