



UNIVERSITY OF GENOVA

PHD PROGRAM IN BIOENGINEERING AND ROBOTICS

A Unified Task Priority Control Framework Design for Autonomous Underwater Vehicles

by

Cris Thomas

Thesis submitted for the degree of *Doctor of Philosophy* (34° cycle)

April 2022

Professor Enrico Simetti

Supervisor

Professor Giuseppe Casalino

Supervisor

Professor Giorgio Cannata

Head of the PhD program

Dibris

Department of Informatics, Bioengineering, Robotics and Systems Engineering

To

My Family for their never ending prayers, love and support.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Cris Thomas

April 2022

Acknowledgements

Thank you God Almighty for your blessings.

First of all, I dedicate this work to my dear and loving parents Roy and Aleyamma. I would not have reached the stage where I would do my doctoral research and finally this document, if it was not with their constant prayers, love, support, and sacrifices. They made me who I am. The credit also extends to my dear and loving brother Jiss who has always stood by me and gave me his wholehearted support throughout my ventures. My family paved the way for me so that I can get closer and closer to achieving my dreams. The countless moments where I was supported by my family in my most difficult times nurtured me to face all the challenges. I owe this document to them.

I am incredibly grateful to my dear supervisors, Prof. Enrico Simetti and Prof. Giuseppe Casalino, for their invaluable advice, continuous support, review comments and patience during my doctoral study. Their immense knowledge and ample experience have encouraged me to achieve great heights in my academic research. I am deeply grateful to Prof. Enrico Simetti for his unwavering support and belief in me to complete this work. He was always available for the discussions and supported me all the way whenever I faced difficulties in my research. I have been fortunate to have supervisors who cared so much about my work and promptly responded to my questions and queries. This work would have been incomplete without their careful supervision.

I would like to thank my thesis reviewers, Prof. Paolo Di Lillo and Prof. Riccardo Costanzi for carefully reviewing this document and providing their valuable comments for further improving this document. I would like to thank the head of my PhD program Prof. Giorgio Cannata and administrative personnel Valentina Scanarotti and Roberta Fraguglia who were there to help me with various administrative challenges in the University. The University of Genova staff were also among the people who helped in times of need.

When I relocated to Italy to pursue my research, I was helped by a lot of people, both at personal and academic levels. I would like to extend my heartfelt thanks to Jomon uncle and family who helped me very much, especially during my starting days in Italy. Their presence were always a comfort for me.

I would like to extend my immense love and gratitude to my dear and loving friend Vidya who has always stood by me, motivated me and supported me, starting from my graduate studies. She has always listened to my concerns patiently and helped me overcoming them, especially during my hard times while living away from my home for extended period of time. Her patience in listening to the never ending conversations about my research is commendable. She always extended her helping hand whenever I was in need.

If I know Italy, it is because of my dear and loving friend Dafrosa. She was always a pillar whom I could lean on during my hard times. The trips we had together throughout Italy helped me to clear my mind whenever I felt overwhelmed with work. She always listened patiently to my problems and helped me to overcome them with her constant motivation and support. She is also the reason for developing my interest in a variety of topics that extends beyond my knowledge.

My dear and loving friend Roopa was always there for me, as a helping hand whenever I faced difficulties coping to the rigorous academic research culture. Her knowledge and experience in this regard always helped me to push forward whenever I felt stuck or confused.

My dear and loving friends Antony and Praveen were always there for my support whenever I was in need of assistance. They are also one among many reasons that helped my stay in Italy. Being brothers to me, they always listened to my problems and guided me with their knowledge and experience. Other friends who extended their immense support includes Divya, Kirti, Nikesh, Vinil, Sherona, Reshma, Julia, Clin and Yusha. I also thank my colleagues Mina, Carlotta, Hossein and Alex for their support.

Last, but not least I thank all my friends, relatives and everyone who have supported me in my life.

Abstract

In this thesis, we investigate the problem of bringing various behaviours of Autonomous Underwater Vehicles under a common control framework. Thereby, we propose a unified guidance and control framework for AUVs based on the task priority control approach. This incorporate various behaviors such as path following, terrain following, obstacle avoidance, as well as homing and docking to stationary and moving docking stations. The integration of homing and docking maneuvers into the task priority framework is thus a novel contribution of this thesis. This integration allows, for example, to execute homing maneuvers close to uneven seafloor or obstacles, ensuring the safety of the AUV by giving the highest priority to the safety tasks. Furthermore, the proposed approach tackles a wide range of scenarios without ad hoc solutions. Indeed, the proposed approach is well suited for both the emerging trend of resident AUVs, which stay underwater for a long period inside garage stations, exiting to perform inspection and maintenance missions and homing back to them, and for AUVs that are required to dock to moving stations such as surface vehicles, or towed docking stations. The proposed techniques are further studied in a simulation setting, taking into account the rich number of aforementioned scenarios.

Table of contents

List of figures	9
List of tables	14
List of Abbreviations	15
1 Introduction	1
1.1 Thesis Contributions	3
1.2 Thesis Structure	4
2 Mathematical Models for AUVs	5
2.1 General notations	5
2.2 Kinematic model	5
2.3 Dynamic model	7
3 Task Priority Framework	9
3.1 Introduction	9
3.2 General Framework Structure	11
3.3 Tasks	13
3.4 Task Hierarchy and Action Sequencing	14
3.5 Task Priority Inverse Kinematics	17
4 Unified Framework Design	19
4.1 Introduction	19
4.2 General Architecture Overview	22
4.3 Task definitions	22
4.3.1 Vehicle Non Actuation Task	23
4.3.2 Minimum Altitude Task	24

4.3.3	Obstacle Avoidance Task	25
4.3.4	Velocity Alignment Task	26
4.3.5	Seafloor Alignment Task	28
4.3.6	Horizontal Alignment Task	30
4.3.7	Follow 2D Path Task	30
4.3.8	Vehicle Depth Task	32
4.3.9	Vehicle Altitude Task	33
4.3.10	Survey Leg Alignment Task	35
4.3.11	Hold Position Task	36
4.3.12	Field Velocity Task	37
4.3.13	Terminal Dock Task	39
4.3.14	Dock Attitude Task	39
4.4	Action based framework design	40
4.5	Mission modelling	47
4.5.1	Constant altitude survey mission for pipeline inspection	48
4.5.2	Constant depth survey mission for bathymetry	50
4.5.3	Resident AUV survey mission for oil rig inspection	51
5	Application to the X300 AUV: Results and Discussion	54
5.1	Simulation Setup	55
5.2	Simulation Results	62
5.2.1	Parking tests in Free Space	62
5.2.2	Docking approaches	69
5.2.3	Docking tests in free space under partially enclosed or constrained situations	73
5.2.4	Complete Mission tests	78
6	Conclusions	95
6.1	Summary of main achievements	96
6.2	Future Scope	98
	References	99

List of figures

2.1	AUV coordinate frame assignments.	6
3.1	Activation function.	12
3.2	Example of a local minima problem at non-conflicting zone.	16
3.3	Example of initial configurations to ensure goal reachability by avoiding local minima.	17
4.1	Control architecture depicting the Planning Layer, the Control Layer, and the Simulated System.	22
4.2	Task: Minimum Altitude.	25
4.3	Task: Obstacle Avoidance.	27
4.4	Task: Velocity Alignment	27
4.5	Task: Seafloor Alignment	28
4.6	Task: Horizontal Alignment.	31
4.7	Task: Follow 2D Path.	32
4.8	Task: Vehicle Depth.	33
4.9	Task: Vehicle Altitude.	34
4.10	Task: Survey Leg Alignment.	35
4.11	Task: Hold Position.	36
4.12	Velocity field-based approach in 2D planar scenario.	37
4.13	Task: Terminal Dock.	40
4.14	Task: Dock Attitude.	41
4.15	Maneuver with <i>Vehicle Altitude</i> task and <i>Obstacle Avoidance</i> Task.	42
4.16	Maneuver with <i>Vehicle Depth</i> task and <i>Minimum Altitude</i> Task.	42
4.17	Parking maneuvers: a) short-range and b) long-range distances.	42
4.18	Velocity field-based approach in 3D scenario.	43

4.19	Docking approach from various sides of the docking station: (a) docking from the back (e.g., to dock on a stationary docking station situated at mid water column, such as submarines or to surface vessels), (b) docking from one side (e.g., to dock to the side of a surface vessel or to submarines), and (c) docking from the top (e.g., to dock to a towed underwater docking station).	44
4.20	Docking configurations for holonomic and non-holonomic approaches: (a) top, (b) bottom, (c) left, (d) rear and (e) right of docking station.	45
4.21	Mission, Actions and Tasks: A general overview of the relationship.	47
4.22	Constant altitude survey mission plan.	49
4.23	Constant depth survey mission plan.	50
4.24	Resident AUV survey mission plan.	52
5.1	Software Program Structure.	55
5.2	Unity GUI.	56
5.3	Live Visual Feedback Display.	57
5.4	Mission Progress Display.	58
5.5	ROS computation graph of nodes from a live simulation of a mission.	58
5.6	AUV: X300 model and allocation of thrusters.	59
5.7	X300: Allocation of altitude sensors w.r.t. center of vehicle body frame and estimation of vehicle altitude as length of the vertical segment. The slanting rays emerging from the vehicle highlight the position of the altitude sensors w.r.t. the body of the vehicle.	60
5.8	X300: Comparison between different maneuvers in free space parking by changing the linear gain, angular gain and modulation parameters.	64
5.9	X300: Maneuver showing the vehicle performing Obstacle Avoidance Task for an obstacle (represented by red sphere) near a hill.	65
5.10	X300: Data showing the distance to object, vehicle depth and corresponding activation for Obstacle Avoidance and Minimum Altitude Tasks.	65
5.11	X300: Maneuver showing the vehicle performing Seafloor Alignment Task close to seafloor.	66
5.12	X300: Data showing the depth and pitch of the vehicle and corresponding activation for Seafloor Alignment and Minimum Altitude Tasks.	67
5.13	X300: Maneuver showing the vehicle performing Horizontal Alignment Task while doing a deep maneuver.	67

5.14	X300: Data showing the depth and pitch of the vehicle and corresponding Horizontal Alignment Task activation.	68
5.15	X300: Time behaviour of the docking station position data and vehicle position data with low sensor data sampling rates. The various phases of the test are denoted in the plot as follows: H - Homing, D - Docking.	69
5.16	X300: Time behaviour of the vehicle position data with very high ocean current rates. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.	70
5.17	X300: Maneuver showing the vehicle approaching a dock and docking at (a) left and (b) right.	71
5.18	X300: Maneuver showing the vehicle approaching a dock and docking at top.	72
5.19	X300: Maneuver showing the vehicle approaching a dock and docking at bottom.	73
5.20	X300: Maneuver showing the vehicle approaching a dock and docking at rear.	74
5.21	X300: Maneuver showing the vehicle performing docking into a tight space inside a moving docking station, starting from a distance.	75
5.22	X300: Maneuver showing the vehicle performing docking into a tight space, starting from close proximity of the moving docking station and going around the docking station structure without collision.	76
5.23	X300: Maneuver showing the vehicle performing a docking to a moving docking station travelling in same direction of the vehicle.	77
5.24	X300: Maneuver showing the vehicle performing a docking to a moving docking station travelling towards the vehicle from opposite side.	78
5.25	Lawn mover Profile: Representation of various locations, waypoints, and maneuvers involved for the complete survey mission.	79
5.26	Velocity Time Behaviour: Plot depicting the time behaviour of desired velocities and the actual/system velocities of the vehicle.	81
5.27	Velocity Time Behaviour: Plot depicting a magnified portion the time behaviour of desired velocities and the actual/system velocities of the vehicle.	81

5.28	Constant altitude mission: Plot showing the mission and docking (top side) to a moving docking station situated close to the seafloor in the 3D space. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station. Notice that the vehicle avoids the 2 nd red obstacle (close to the docking station) in the path, by travelling below it due to availability of free space without breaching the minimum altitude safety limit.	82
5.29	Constant altitude mission: Plot showing the mission and docking (top side) to a moving docking station situated close to the seafloor from top view. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station. Notice that the vehicle avoids the 2 nd red obstacle (close to the docking station) in the path, by travelling below it due to availability of free space without breaching the minimum altitude safety limit.	83
5.30	State Transition Diagram: Constant altitude survey.	85
5.31	Constant altitude mission: Time behaviour of the vehicle depth, pitch and obstacle avoidance data. The various phases of the mission are denoted in the plot as follows: P - Parking, CAS - Constant Altitude Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.	85
5.32	Constant altitude mission: Time behaviour of Obstacle Avoidance, Minimum Altitude, Vehicle Altitude, Seafloor Alignment and Horizontal Alignment task activation functions. The various phases of the mission are denoted in the plot as follows: P - Parking, CAS - Constant Altitude Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.	86
5.33	Constant depth mission: Plot showing the mission and docking (bottom side) to a moving docking station situated close to the surface in the 3D space. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station.	87
5.34	Constant depth mission: Plot showing the mission and docking (bottom side) to a moving docking station situated close to the surface from top view. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station.	88
5.35	State Transition Diagram: Constant depth survey.	89

5.36	Constant depth mission: Time behaviour of the vehicle depth, pitch and obstacle avoidance data. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.	89
5.37	Constant depth mission: Time behaviour of Obstacle Avoidance, Minimum Altitude, Vehicle Depth, Seafloor Alignment and Horizontal Alignment task activation functions. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.	90
5.38	Resident AUV mission: Plot showing the mission and docking (top) to a docking station situated close to the seafloor in the 3D space. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station. . .	91
5.39	Resident AUV mission: Plot showing the mission and docking (top) to a docking station situated close to the seafloor from top view. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station. . .	92
5.40	State Transition Diagram: Resident AUV survey.	92
5.41	Resident AUV mission: Time behaviour of the vehicle depth, pitch and obstacle avoidance data. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.	93
5.42	Resident AUV mission: Time behaviour of Obstacle Avoidance, Minimum Altitude, Vehicle Depth, Seafloor Alignment and Horizontal Alignment task activation functions. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.	94

List of tables

4.1	Table depicting various tasks used in this research.	23
4.2	List of tasks in actions that encompass the various behaviours of the vehicle. In task type, I refers to an inequality objective and E refers to an equality objective. The numbers denote the hierarchical priority of each task in their respective action where lower values represent a higher priority for that task in the action.	46
5.1	X300: System Parameters and Dynamic Model Parameters. Thruster distance represents the longitudinal distance of each thruster from the center of body frame.	60
5.2	Table showing details of Commercially available AUV sensors.	61
5.3	Activation points and Threshold ranges of various parameters of the tasks used in the simulations. The threshold range represent the transition region of the task.	63
5.4	Mission parameters. All the positions are defined w.r.t. the world frame. . .	80
6.1	Table depicting the recommendations for the docking approach based on the vehicle type and docking station type.	98

List of Abbreviations

AHRS	Attitude and Heading Reference System
ASV	Autonomous Surface Vehicle
AUV	Autonomous Underwater Vehicle
DCL	Dynamic Control Layer
DVL	Doppler Velocity Log
IMU	Multi Beam Echo Sounder
INS	Inertial Navigation System
KCL	Kinematic Control Layer
LOS	Line of Sight
MBES	Multi Beam Echo Sounder
QP	Quadratic Program
ROV	Remotely Operated Vehicle
SNAME	Society of Naval Architects and Marine Engineers
TPIK	Task Priority Inverse Kinematics
USBL	Ultra Short Base Line

Chapter 1

Introduction

In the past couple of decades, we have seen a tremendous increase in the use of Autonomous Underwater Vehicles (AUVs) in a wide variety of applications ranging from sub-sea explorations to ocean research and development [1]. Some notable areas where AUVs are widely used [2] involve mapping, inspections, surveys, and military roles. Damages from incidents such as the Deepwater Horizon tragedy [3] could have considerably reduced if there was a possibility of using an AUV for inspecting the blowout preventer, riser, and surrounding seabed of the oil rig. Necessary precautions and safety measures could have been taken well in advance to reduce the impact of such mishaps. Currently, AUVs have evolved to accomplish complex and highly demanding tasks such as search and rescue [4], underwater manipulation [5], homing, docking [6], etc. Apart from this, these underwater vehicles are required to autonomously carry out mission tasks that includes path following, terrain following, obstacle avoidance, and such. A growing trend [7] has been observed with a new class of vehicles named field resident AUV systems that carry out these tasks. Complementing the Remotely Operated Vehicles (ROVs) in the midterm [8, 9], resident AUVs now hold a position as a completely unique class of underwater robotic systems merging the advances in technology and demand for new applications. The various components of such systems [10] additionally include a home base for docking when not performing the mission, and various power and communication facilities. In [11], the authors conducted a workshop that studied various applications of resident AUVs, focusing more on mid-water and seafloor observations. In [12], the authors discuss some capabilities of such systems by studying the Marlin Mk3 AUV by Lockheed Martin and Saab Seaeye's Sabertooth AUV. The work in [13] is about the deployment of resident AUV Systems by Chevron for pipeline and facility inspection operations. FlatFish [14] is used to perform an on-demand close visual inspection on sub-sea structures within an oil and gas field. The work in [15] discuss the monitoring of

underwater infrastructure using a resident AUV Tri-TON 2. Saipem has developed a resident AUV named Hydrone-R¹ capable of carrying out light construction works and advanced inspections on sub-sea assets. In [16], the authors conduct a review of resident AUV systems, along with their application areas, capabilities, and challenges. Such class of AUVs are thus needed by the current industry, favoring a system that is capable of carrying out all tasks and operations autonomously, thus motivating the problem of autonomous mission capability, homing, and docking. Another increasing trend sees AUVs deployed from surface vessel or submarines, and requires them to home back autonomously. Retrieval of REMUS 100 AUV from an Autonomous Surface Vehicle (ASV) using a taut tether line is presented in [17], while docking of AUVs to a submarine is presented in [18]. The work in [19] discusses the launch, deployment, and recovery of AUVs from submarines via conventional torpedo tubes and via a hangar mounted to the submarine body. In [20], the authors present the recovery of AUVs from submarines by operating a taught wire for retrieval. Thus, autonomous deployment, homing, and docking are desirable features for AUVs. For long-term deployments, it is required to have these capabilities carried out without any intervention from any human operators. Having these autonomous capabilities could save time, and reduce human error and fatigue, even for systems such as ROVs or other tethered underwater or surface vehicles. It is thus interesting to study and find solutions in the autonomous capabilities of the vehicle that would encompass all behaviors that are needed right from deployment till recovery, and also with a long-term vision of intervention capabilities [21, 22].

Given the aforementioned trends, it is clear that there is a need to integrate into AUVs the capability to home into and be able to dock into a stationary or moving docking station, be it underwater such as to submarines, a station on the seafloor, or on the surface such as a surface vehicle. The docking maneuver is usually performed in two phases: The first phase is a long-range homing that involves the identification of the homing signal from the docking system by the AUV to proceed in the general direction of the homing signal. As the AUV approaches closer to the docking station, the terminal docking phase is performed where the AUV proceeds further towards the docking system and attains the dock position. Recent trends denote that this homing and approach maneuver is performed largely by ad hoc controllers. In [23], the authors discuss a docking system designed for REMUS AUV that allows acoustic homing to the docking station. The work in [24] presents a docking assessment algorithm for assessing the feasibility of the desired path for docking of a torpedo-shaped AUV using depth tracking via neural network-generated paths, docking feasibility region analysis to provide a possibility region of successful docking, and docking success probability evaluation based

¹<http://www.saipem.com/en/media/news/2019-07-31/sonsub/>

on the probability of sensor data. A vision-based docking system [25] was developed for Starbug AUV where a target is detected using a fast pole detection algorithm based on Haar rectangular features and image rotations. In [26], the authors discuss an optical quadrant tracker method for terminal guidance of underwater vehicles to control the horizontal and vertical movements of the vehicle by appropriate proportional fin commands, thereby steering the vehicle. A vision-based localization for resident underwater vehicles, consisting of beacon lights attached to a tether management system, creating a unique light marker fixed at the target location for pose estimation, is discussed in [27]. Docking of intervention AUVs using sonar and video image processing algorithms accompanied with navigational data obtained from conventional sensors is explained in [28]. The work in [29] discusses the concept and testing of a bio-inspired electromagnetic homing guidance system developed for AUVs at close range to the dock, using a differential magnetometer system fitted with magnetic field sensors at the forward and aft of the AUV for detecting the dipole magnetic field of fixed frequency and power generated from the dock. A robust docking guidance system is identified as a core and crucial component for ensuring successful AUV homing and docking.

1.1 Thesis Contributions

The main contributions of this thesis are the following:

- C1** Develop a unified control system based on the flexible task priority framework incorporating different kind of tasks, thereby achieving flexibility in the control system itself (tasks).
- C2** Generate different behaviours to handle various phases of the mission within the same framework, focusing on a general control strategy (action sequencing).
- C3** Develop various homing and docking configurations that allows the vehicle to dock onto both stationary and moving docking stations situated or moving at seafloor, mid-water or at sea level (docking from sides, top, bottom, rear, etc).
- C4** Integrate the homing and docking maneuver into the flexible task priority framework for performing missions such as Constant Altitude, Constant Depth, Resident AUV survey etc. and also tackling various challenging scenarios such as avoiding obstacles, moving parallel to seafloor, maintaining a minimum altitude, maintain the pitch within safety limits etc., taking into account all the constraints and objectives (Missions).

- C5** Exploit the flexibility allowed by the task priority framework, allowing a holonomic vehicle behave like a non-holonomic vehicle to improve performance by reducing the power consumption and drag (Align task).

The following work [30] is published as part of this thesis:

Cris Thomas, Enrico Simetti, and Giuseppe Casalino. "A Unifying Task Priority Approach for Autonomous Underwater Vehicles Integrating Homing and Docking Maneuvers" in *Journal of Marine Science and Engineering*, 2021

1.2 Thesis Structure

The thesis is structured as follows. Chapter 2 will introduce the general notations used in the thesis and also discusses the mathematical models for the AUV. The kinematic and dynamic model of the AUV are then used in the simulations later. Chapter 3 gives an insight into the task priority framework that this thesis bases on. It will provide information regarding the tasks, task hierarchy, collecting multiple tasks to form action, action sequencing using priority of the tasks and the required vehicle behaviour. Last, but not least, it will discuss the task priority inverse kinematics. Chapter 4 discusses the development of a unified control system based on the flexible task priority framework incorporating different kind of tasks, thereby achieving flexibility in the control system itself (tasks). It introduces the various tasks in detail and also gives information on the framework design based on actions. Various examples will be discussed where various missions are modelled based on the required behaviour of the vehicle. This will include the contributions **C1**, **C2** and **C3**. Chapter 5 encompass the simulation results of various tests that were carried out to prove the working and application of the proposed unified control framework model based on task priority. It includes free space tests as well as mission based tests and includes the contributions **C4** and **C5**. Chapter 6 will conclude the thesis.

Chapter 2

Mathematical Models for AUVs

2.1 General notations

Consider two generic frames $\langle f_1 \rangle$ and $\langle f_2 \rangle$ with unit vectors representing each directions as $(\mathbf{i}_{f_1}, \mathbf{j}_{f_1}$ and $\mathbf{k}_{f_1})$ and $(\mathbf{i}_{f_2}, \mathbf{j}_{f_2}$ and $\mathbf{k}_{f_2})$ respectively. Let ${}^{f_2}\mathbf{R}_{f_1}$ or simply $\mathbf{R}$ represents the orientation matrix of a frame $\langle f_1 \rangle$ w.r.t. another frame $\langle f_2 \rangle$ as

$${}^{f_2}\mathbf{R}_{f_1} = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\theta + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}, \quad (2.1)$$

where c and s denote the cosine and sine values of the respective Euler angles, respectively. The transformation matrix of frame $\langle f_1 \rangle$ w.r.t. frame $\langle f_2 \rangle$ frame is thus given by ${}^{f_2}\mathbf{T}_{f_1} = \begin{bmatrix} {}^{f_2}\mathbf{R}_{f_1} & \mathbf{p} \\ 0 & 1 \end{bmatrix}$, where $\mathbf{p}$ represents the position vector of origin of frame $\langle f_1 \rangle$ w.r.t. frame $\langle f_2 \rangle$.

2.2 Kinematic model

This section discusses the general kinematic model of the vehicle presented in the thesis. A class of torpedo-shaped vehicles that can translate is considered having 5 actuated degrees of freedom which are along the surge, sway, heave, pitch, and yaw directions. The notations used in this thesis to describe rigid-body dynamics follow the conventions of the Society of Naval Architects and Marine Engineers (SNAME) presented in [31].

To determine the position, orientation, and velocity of the vehicle, two reference frames are used: a world-fixed reference frame $\langle w \rangle$ and a body fixed reference frame $\langle b \rangle$ as shown in Figure 2.1. The $\langle w \rangle$ frame has its x axis pointing to the North, the y axis pointing to the

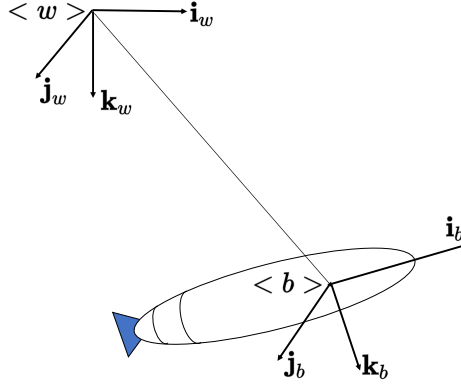


Figure 2.1 AUV coordinate frame assignments.

East, and the z axis pointing to the center of the Earth. The unit vectors representing each directions in $\langle w \rangle$ are $\mathbf{i}_w$, $\mathbf{j}_w$ and $\mathbf{k}_w$, respectively. The $\langle b \rangle$ frame relates to the vehicle with its x axis pointing to the forward direction, the y axis to the right direction, and the z axis points vertically down. The unit vectors representing each directions in $\langle b \rangle$ are $\mathbf{i}_b$, $\mathbf{j}_b$ and $\mathbf{k}_b$, respectively. Both these frames are related by the Euler angles yaw(ψ), pitch(θ), and roll(ϕ) as indicated in equation 2.1.

In the state space representation, $\boldsymbol{\eta} = [\boldsymbol{\eta}_1^\top \boldsymbol{\eta}_2^\top]^\top \in \mathbb{R}^{6 \times 1}$ represents the vehicle configuration vector composed of vehicle linear position $\boldsymbol{\eta}_1 = [x, y, z]^\top$ and vehicle angular position $\boldsymbol{\eta}_2 = [\phi, \theta, \psi]^\top$ respectively in $\langle w \rangle$ frame. ${}^b\mathbf{v} = [{}^b\mathbf{v}^\top {}^b\boldsymbol{\omega}_{b/w}^\top]^\top \in \mathbb{R}^{6 \times 1}$ represents the generalized velocities expressed in $\langle b \rangle$, where ${}^b\mathbf{v} = [u, v, w]^\top \in \mathbb{R}^{3 \times 1}$ is the absolute linear velocity and ${}^b\boldsymbol{\omega}_{b/w} = [p, q, r]^\top \in \mathbb{R}^{3 \times 1}$ is the angular velocity of the $\langle b \rangle$ w.r.t. $\langle w \rangle$ expressed in $\langle b \rangle$.

The ocean current is assumed to be constant and irrotational. For a generalized vector of ocean current velocity denoted as ${}^w\mathbf{v}_c \in \mathbb{R}^{6 \times 1}$ in $\langle w \rangle$, ${}^w\dot{\mathbf{v}}_c = 0$ as the current is constant and the angular components are 0 as it is irrotational. Now the ocean current velocities ${}^b\mathbf{v}_c \in \mathbb{R}^{6 \times 1}$ in $\langle b \rangle$ is obtained as,

$${}^b\mathbf{v}_c = \begin{bmatrix} \mathbf{R}^{-1}(\boldsymbol{\eta}_2) & 0 \\ 0 & \mathbf{R}^{-1}(\boldsymbol{\eta}_2) \end{bmatrix} {}^w\mathbf{v}_c. \quad (2.2)$$

The kinematic model links the derivative of the absolute position and the yaw–pitch–roll Euler derivatives to the relative velocity of the vehicle w.r.t. the water, the ocean current

velocity and the angular velocity of the vehicle using the following relationship,

$$\dot{\boldsymbol{\eta}} = \begin{bmatrix} \mathbf{R}(\boldsymbol{\eta}_2)^b \mathbf{v}_r \\ \mathbf{J}_e(\boldsymbol{\eta}_2)^b \boldsymbol{\omega}_{b/w} \end{bmatrix} + {}^w \mathbf{v}_c \quad (2.3)$$

where $\dot{\boldsymbol{\eta}} = [\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi}]^\top$, ${}^b \mathbf{v}_r \in \mathbb{R}^{3 \times 1}$ is the relative linear velocity vector (w.r.t. the water). $\mathbf{J}_e(\boldsymbol{\eta}_2)$ is the Jacobian connecting the body angular velocities ${}^b \boldsymbol{\omega}_{b/w}$ to the y-p-r Euler rates $\dot{\boldsymbol{\eta}}_2$, given as

$$\mathbf{J}_e(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix}, \quad (2.4)$$

where c , s and t denote the cosine, sine and tangent values of the respective Euler angles.

2.3 Dynamic model

The dynamic model of the vehicle is based on the general Fossen model for underwater vehicles [32–34]. The Newton–Euler dynamic equations of motion expressed in $\langle b \rangle$ are given by

$${}^b \mathbf{v}_r = {}^b \mathbf{v} - {}^b \mathbf{v}_c, \quad (2.5)$$

$$(\mathbf{M}_{rb} + \mathbf{M}_A) {}^b \dot{\mathbf{v}}_r + \left(\mathbf{C}_{rb}({}^b \mathbf{v}_r) + \mathbf{C}_A({}^b \mathbf{v}_r) \right) {}^b \mathbf{v}_r + \mathbf{D}({}^b \mathbf{v}_r) {}^b \mathbf{v}_r + {}^b \boldsymbol{\tau}_{rf} = {}^b \boldsymbol{\tau}_v + {}^b \boldsymbol{\tau}_L + {}^b \boldsymbol{\tau}_E, \quad (2.6)$$

where ${}^b \mathbf{v}_r \in \mathbb{R}^{6 \times 1}$ is the generalized vector of body relative velocities, ${}^b \mathbf{v}_c \in \mathbb{R}^{6 \times 1}$ is the generalized vector of ocean current velocity in $\langle b \rangle$, $(\mathbf{M}_{rb} + \mathbf{M}_A)$ is the system inertia matrix (including added mass), $(\mathbf{C}_{rb}({}^b \mathbf{v}_r) + \mathbf{C}_A({}^b \mathbf{v}_r))$ represents the Coriolis and centripetal matrix (including added mass), $\mathbf{D}({}^b \mathbf{v}_r)$ is the drag matrix, ${}^b \boldsymbol{\tau}_{rf}$ is the generalized vector of restoring forces (gravitational and buoyancy), ${}^b \boldsymbol{\tau}_v$ represents the generalized vector of propulsion forces, ${}^b \boldsymbol{\tau}_L$ is the generalized vector of lifting forces, and ${}^b \boldsymbol{\tau}_E$ is the generalized vector of environment forces on a rigid body. As the fluid flow is expected to be traversal/longitudinal w.r.t. the length of the vehicle during typical operating conditions such as hovering or forward motion characterized by slow velocities and accelerations (thus small angles of attack), the lift forces and moments are expected to be negligible w.r.t. drag forces and moments and can be neglected. The drag matrix $\mathbf{D}({}^b \mathbf{v}_r)$ is modeled as the sum of the linear and quadratic

damping terms with diagonal structure as

$$\mathbf{D}({}^b\mathbf{v}_r) = \left(\mathbf{D}_I + \mathbf{D}_q({}^b\mathbf{v}_r) \right), \quad (2.7)$$

The generalized vector of propulsion forces produced by the thrusters, is given by

$${}^b\boldsymbol{\tau}_v = \mathbf{B}({}^b\mathbf{v}_r)\mathbf{u}, \quad (2.8)$$

where $\mathbf{u} \in \mathbb{R}^{5 \times 1}$ is the vector of control inputs in which each component is the force produced by the respective thruster and $\mathbf{B}$ is the control allocation or mapping matrix that allows computing the resulting body wrench ${}^b\boldsymbol{\tau}_v$, given a certain $\mathbf{u}$. The relation between the force of a given thruster and its velocity is highly nonlinear and depends on the thruster characteristics, thruster motor velocity, and the fluid velocity passing through the thruster.

Chapter 3

Task Priority Framework

3.1 Introduction

A robust guidance system can be identified as a core and crucial component for ensuring successful autonomous operations. For systems such as industrial and residential applications where the automatic procedures are important, it is desirable to have a fully fledged control framework to solve all different control problems. Since such systems are required to complete different types of goals, a simple, flexible and efficient way to specify these tasks is paramount. Due to this increase in the capabilities and constraints assigned to the vehicle, a transition is observed with the use of a flexible guidance and control framework [35–40].

The research in industrial robotics focused on how to effectively specify the control objectives of a robotic system, especially for redundant systems. The task-based control [41], also known as operational space control [42], defines control objectives in a coordinate system that is directly relevant to the task that needs to be performed, rather than in the generalized coordinates of the robotic system. [43, 44] exploits the redundancy of the system. Such an idea was immediately enhanced by the introduction of the concept of task priority [45]. In that work, a primary task was executed, and a secondary task was accomplished only in the null space of the primary one, in order to guarantee the invariance of the main task w.r.t. the secondary one. This concept was later generalized to any number of task-priority levels in [36]. However to deal with singularities, a sub-optimal approach of computing the secondary task separately (assuming it is alone) and projecting it in the null space of the higher priority one is presented in [46]. This work uses a variable damping factor as proposed in [41] to deal with singularities and was later generalized as null-space-based behavioral control in [47]. The task priority approach can then be seen to many systems including mobile manipulators [5, 48–51] and modular robots [52].

Classical task priority framework doesn't take into account of inequality control objectives due to the problem of deactivation of a task causing a discontinuity in the null space projector, ultimately leading to a discontinuity in the control law [37]. Methods like a new inverse operator [53] dealing with the activation and deactivation of rows of a single multidimensional task was introduced and even extending it to a hierarchy of tasks in [54] which was computationally intensive as the number of tasks increased. [55] attempts to solve a hierarchy of tasks at any number of levels by incorporating the inequality control objectives as inequality constraints in a Quadratic Program (QP). The work cannot handle the activation or deactivation of tasks as the cascade of QP problems grows in dimension. Although [56] tackles the inequality constraints within a constrained minimization problem, it does not explicitly activate or deactivate an equality task. Set-based objectives are proposed in [57], but does not deal with activation and deactivation of tasks.

A novel task priority resolution mechanism, which manages equality and scalar-only inequality control objectives, is developed and exploited for the first time for the coordinated control of a floating base and a manipulator, for performing autonomous floating intervention [58]. Successively, the concepts developed in the TRIDENT project were further enhanced within the MARIS project, with the definition of a task priority framework able to activate and deactivate equality/inequality control objectives of any dimension (i.e. not limited to scalar ones) depending on the system requirements [59]. This feature allows the user to put safety and operational-enabling objectives at the highest priority, as they should be. Experiments in free floating grasping have been conducted, with multiple attempts to test the repeatability and robustness of the control [60]. The MARIS project also studied the extension of the control architecture to cooperative agents [61].

The task priority framework [62–65] is flexible and can accommodate various tasks and operations that are required, favoring a priority-based systematic execution of tasks. Not only does it provide a unified approach, but also a systematic control that will, if required, take into account additional features such as physical/operational constraints of the vehicle (e.g., vehicle under-actuation), safety limits (e.g., maintaining a safe distance from dock to avoid collision), constraints in approach (e.g., docking from stern), and energy expenditure (e.g., minimize drag by exploiting translation only when necessary), to name but a few. A priority-based control to integrate inequality control objectives and task transitions is presented in [59], displaying how inequality control objectives could be well defined within the task priority framework without causing any discontinuities. This was further expanded in the MARIS, DexROV, and ROBUST projects [60, 66–68]. The work in [61] shows the capabilities of the task priority framework in the cooperative control of multiple underwater

manipulators where a cooperative transportation scenario is considered (the same strategy has been demonstrated with ground mobile manipulators showing experimental results in [69]). Inspired from these literature, a task priority control framework is used as the control framework for our research. Various tasks are defined in the framework where they are achieved in the order of priority. Thanks to the flexibility in incorporating various tasks, it is now easier to expand the capabilities of our controller.

However, the detailed specifics of the framework is not the focus of this thesis. This is a technique that has emerged in the past few years and is proposed to be used in this work due to its flexibility and simplicity in terms of computational point of view. Hereafter, recall the basic principles and working of the framework to be able to understand the rest of the manuscript.

3.2 General Framework Structure

The task priority framework is presented in [63]. In this framework approach, objectives define a specific condition that needs to be satisfied. Consider a position dependent scalar/associated variable $x(\boldsymbol{\eta})$, then an equality control objective should satisfy $x(\boldsymbol{\eta}) = x_0$ as $t \rightarrow \infty$ and, an inequality control objective should satisfy $x(\boldsymbol{\eta}) \leq x_{max}$ or $x(\boldsymbol{\eta}) \geq x_{min}$ as $t \rightarrow \infty$. To satisfy this control objective, the time derivative of the variable $x(\boldsymbol{\eta})$ should allow it to converge to the desired region or value. A control task will track a desired reference rate $\dot{x}$, to drive an associated variable $x(\boldsymbol{\eta})$ towards its corresponding objective. Tasks will be discussed further in Section 3.3.

Another key element of the task priority framework are the activation functions $a_i(x)$. The activation value for a task defines if that particular task is active, in transition or deactivated. This in turn implies if the corresponding reference signal be tracked or not. These are continuous sigmoid functions of the scalar variable $x(\boldsymbol{\eta})$ and are in the range $[0, 1]$. The function assumes 0 in the valid region of the objective. The value of the activation function is non-zero until the associated objective is achieved. The function is used to activate or deactivate the control tasks. It avoids over-constraining the system and also allows a smooth transition between the activation and deactivation phases to prevent chattering around the transition values.

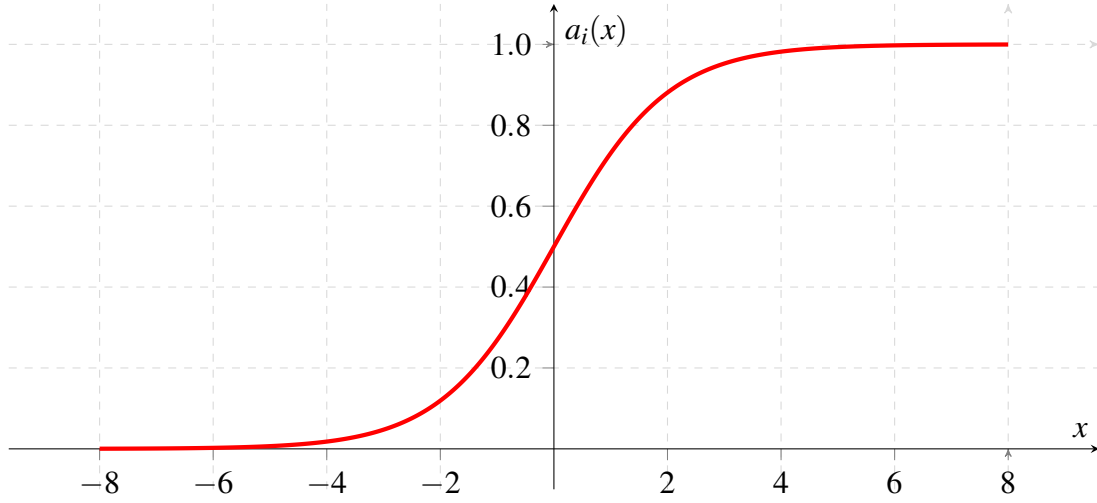


Figure 3.1 Activation function.

For each inequality control objective, for objectives of the type $x(\boldsymbol{\eta}) \geq x_{\min}$, consider as activation function a_i defined as follows,

$$a_i(x) \triangleq \begin{cases} 1, & x(\boldsymbol{\eta}) < x_{\min} \\ s(x), & x_{\min} \leq x(\boldsymbol{\eta}) \leq x_{\min} + \Delta \\ 0, & x(\boldsymbol{\eta}) > x_{\min} + \Delta \end{cases}, \quad (3.1)$$

where Δ is the size of the transition zone, allowing for a smooth activation/deactivation of the task itself, and $s(x)$ is a sigmoid function joining the two extrema with continuity. A similar function can be defined for objectives of the class $x(\boldsymbol{\eta}) \leq x_{\max}$.

The sigmoid function $s(x)$ used for activation as depicted in Figure 3.1 is,

$$s_g(x) \triangleq \begin{cases} 1, & x < x_{\min} \\ \frac{1}{2} \left(\cos \left(\frac{(x-x_{\min})\pi}{\Delta} \right) + 1 \right), & x_{\min} \leq x \leq x_{\min} + \Delta \\ 0, & x > x_{\min} + \Delta \end{cases}, \quad (3.2)$$

for objectives of the type $x \geq x_{\min}$ and $s_l(x) \triangleq 1 - s_g(x)$ for objectives of the type $x \leq x_{\max}$.

For equality control objectives $a_i(x) = 1$

Furthermore, the control objective a_i can be constructed as the product of two functions,

$$a_i \triangleq a_i^p a_i^s, \quad (3.3)$$

which have the following specific purposes:

- a_i^s is used to activate/deactivate the task based on its internal state, i.e. the current value of the actual i -th component x_i ;
- a_i^p is instead used to activate/deactivate the task based on external parameters to the task itself, e.g. an activation based on the current elapsed mission time.

As for the activation value a_i^p , is instead a value which can be exploited to perform task transitions based on variables external to the task itself. For example, consider an example where a vehicle needs to maintain a minimum altitude from the seafloor. Here, the activation function a_i^p of the minimum altitude task will be a function of the current altitude of the vehicle.

3.3 Tasks

Consider implementing a certain behaviour to our system, for example, moving the vehicle to a certain position. If we need our system to implement that behaviour using a control objective, the derivatives of the variables of interest should be under our direct control. In case where the variables of interest cannot be controlled directly, we can influence them by acting on some other variables, i.e., control variables, which are defined by the vehicle linear and angular velocities at the kinematic level. Control tasks can thus be best defined as tracking a suitable reference rate (e.g. velocity reference rate) so as to drive the associated control variable towards its corresponding objective (e.g. required final position).

The generalized formulae that derives the reference rate for a task is defined as,

$$\dot{\bar{x}} = \gamma(\bar{x} - x), \gamma > 0, \quad (3.4)$$

where $\dot{\bar{x}}$ is the reference rate, γ the associated gain, $\bar{x}$ the reference variable and $\bar{x} - x$ the corresponding control objective. This represents the desired rate of change of variable x .

For control tasks, the control objective Jacobian links the control variable $\dot{x}$ to the system velocity vector $\mathbf{v}$ using the relationship,

$$\dot{x} = \mathbf{g}^\top \mathbf{v}, \quad (3.5)$$

where the vector $\mathbf{g} \in \mathbb{R}^n$ denotes the Jacobian for the task. Thus, the control task will ensure that the derivative $\dot{x}$ is as close as possible to the reference rate $\dot{\bar{x}}$.

Furthermore, the reference rate can be vector or scalar. For example, let us consider a non-zero geometric vector $\mathbf{s}$ as:

$$\mathbf{s} \triangleq \mathbf{n}\sigma, \sigma > 0. \quad (3.6)$$

The differentiation of $\mathbf{s}$ w.r.t. a generic frame $\langle a \rangle$ gives

$$D_a \mathbf{s} = D_a(\mathbf{n}\sigma) = \mathbf{n}\dot{\sigma} + \sigma D_a \mathbf{n}, \quad (3.7)$$

where the first term is proportional to $\mathbf{n}$, hence aligned to $\mathbf{s}$, and the second term is proportional to the time-derivative of the non-zero unit vector $\mathbf{n}$, hence orthogonal to $\mathbf{s}$. If we want to drive $\mathbf{s}$ to 0, we can either drive $\mathbf{s}$ as vector to zero following a prescribed time behaviour for all components in $\mathbf{s}$. Equivalently, we can drive the norm σ to 0. Here, we do not impose time behaviour to the vector and since the unit vector $\mathbf{n}$ is required, problems would arise when this unit vector is ill-defined in the proximity of 0.

3.4 Task Hierarchy and Action Sequencing

After the tasks are defined, an action can be simply put as a prioritized order of active control tasks in a certain hierarchy (priority level) that needs to be managed concurrently by the system. If two or more control tasks are assigned to the same priority level, then they are grouped into a multidimensional control task. In the task priority based approach, the lower priority tasks do not interfere with the higher priority tasks. However, whenever a task is deactivated, it will not use any degree of freedom, eliminating the problem of over-constraining the system. By defining various actions, a lower priority task can be easily made a higher priority one. This allows us to define multiple actions having same objectives but with different priorities. A finite state machine is used to encapsulate each set of actions and their functions. Transition between various actions also means a transition between the states that represent them.

Consider the following two hypothetical lists of scalar tasks (now labeled with alphabetic letters) for two different actions $\mathcal{A}_1$ and $\mathcal{A}_2$, where $A \prec B$ denotes that task A has higher priority than task B and B, C means that task B has equal priority to task C .

$$\begin{aligned} \mathcal{A}_1 : A \prec B, C, D \\ \mathcal{A}_2 : A \prec D \prec C, E \end{aligned}$$

where A, D, C are in common, but with D at a different priority ordering w.r.t. C within the two lists. A unified list can be constructed incorporating both $\mathcal{A}_1$ and $\mathcal{A}_2$ is denoted as,

$$\mathcal{U} : A \prec D1 \prec B, C, D2, E;$$

where $D1$ and $D2$ represent the same objective D , but with a different priority. It is thus evident that a suitable insertion/deletion of some of its entries, the two original lists can be reconstructed. For example, $\mathcal{A}_1$ can be obtained by removing $D1$ and E from the unified list. Conversely, $\mathcal{A}_2$ can be obtained by removing the control objectives $D2$ and B . Such a trivial logic mechanism for extracting the appropriate task sequences from the unified one is implemented through the use of the continuous activation functions as discussed in 3.2.

Once an action is defined, for each priority level k , the following quantities are computed [67]:

- $\dot{\mathbf{x}}_k \triangleq [\dot{x}_{1,k} \dots \dot{x}_{m_k,k}]^\top \in \mathbb{R}^{m_k}$ is the collection of reference rates of the m_k scalar task for priority level k ;
- $\mathbf{J}_k$ is the Jacobian relation that connects the time derivative $[\dot{x}_{1,k} \dots \dot{x}_{m_k,k}]^\top$ of the k th task vector with the system velocity vector $\mathbf{v}$, obtained by stacking the rows of the control objective Jacobian from equation 3.5; and
- $\mathbf{A}_k \triangleq \text{diag}(a_{1,k}, \dots, a_{m_k,k})$ is the diagonal matrix constituting the activation functions.

For multi-dimensional control objective, the activation value associated to each i -th of its components denoted as a_i are organized into a diagonal matrix $\mathbf{A}_k$ of scalar activation functions as depicted in 3.8,

$$\mathbf{A}_k = \begin{bmatrix} a_{k,1} & 0 & 0 \\ 0 & \dots & 0 \\ 0 & 0 & a_{k,l} \end{bmatrix}, \quad (3.8)$$

where k represents the k^{th} priority level and l the number of dimension of the priority level.

- if $a_k = 1$, the associated scalar task is called active and the corresponding actual $\dot{x}$ should therefore track $\dot{\bar{x}}$ as close as possible;
- if $a_k = 0$, the scalar task is termed inactive and the actual $\dot{x}$ should be unconstrained;
- if $0 < a_k < 1$, the scalar task is termed in *transition* and the actual $\dot{x}$ should smoothly evolve between the two previous cases.

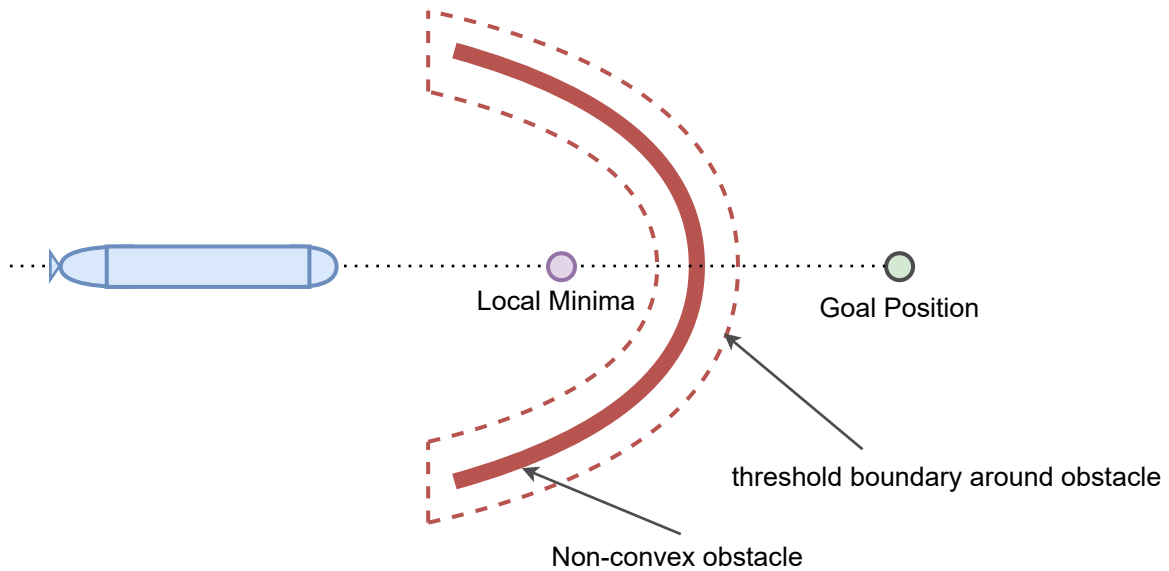


Figure 3.2 Example of a local minima problem at non-conflicting zone.

An action is concluded successfully when its action defining objectives are satisfied. Thus, a necessary condition to guarantee that an action can be terminated is the existence of a non-void set of configurations where all high priority and action-defining objectives are contemporaneously satisfied. The presence of such a non-conflicting zone is, however, not a sufficient condition, as it does not imply its reachability in correspondence of all system initial configurations starting outside it. Such a reachability problem affects all the task priority strategies. E.g.: Consider two objectives where one is to reach a goal position and another is to avoid any obstacles. If the goal is on the opposite side of a non-convex obstacle, the vehicle would proceed into the obstacle as depicted in Figure 3.2. However due to the presence of obstacle avoidance, the vehicle will not crash into this non-convex obstacle, but would rather get trapped in the local minima (due to the starting configuration of the vehicle) as the obstacle pushes the vehicle back and the goal pulls the vehicle forward. Thus this is an example situation of a non-conflicting zone (since the goal is reachable), but not reachable from the initial configuration of the vehicle. However, the vehicle can start from various other initial configurations where the reachability of goal is not affected by the local minima as depicted in Figure 3.3. Here the starting configurations are not attracted by the local minima present on one side of the non-convex obstacle.

Objectives are also not satisfied in the presence of a non-conflicting zone defined by an ill-posed problem. For example, defining a goal position below the minimum safety altitude of the vehicle or defining a goal position inside an obstacle is an ill-posed problem. These problems need to be addressed at a higher level. The creation of different control

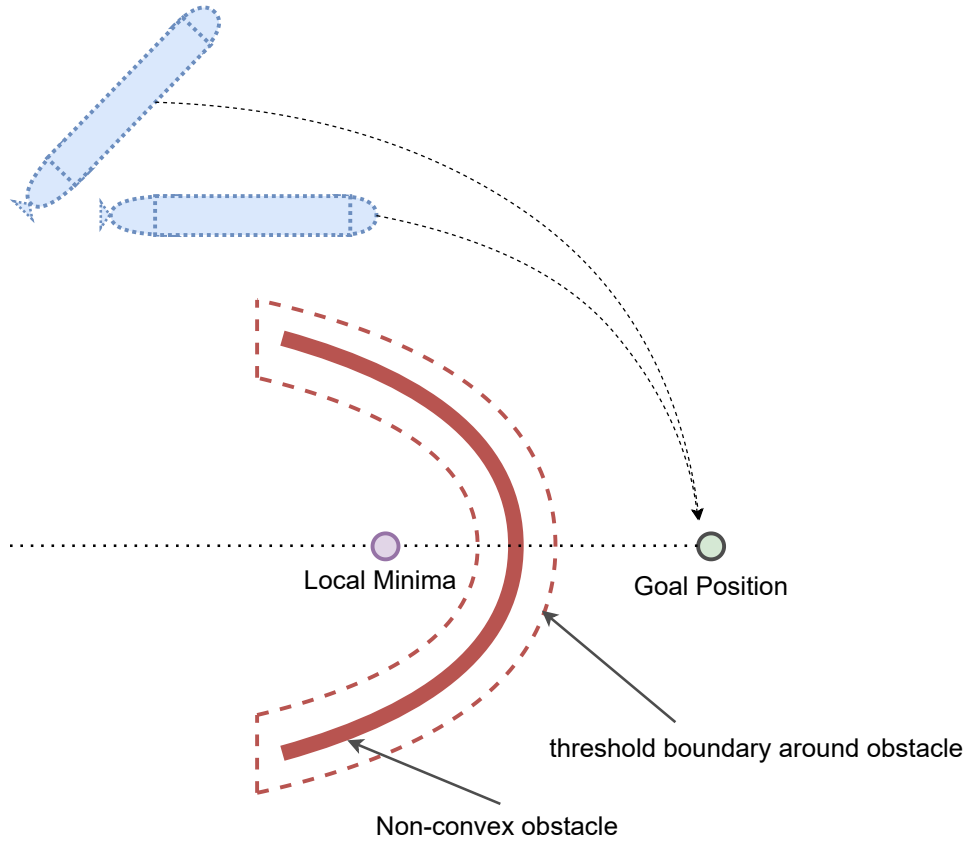


Figure 3.3 Example of initial configurations to ensure goal reachability by avoiding local minima.

actions helps in defining a set of well structured blocks, that can be sequenced at the higher level. In case of inability of completing the required action, the higher level logic devises an alternative solution, e.g. a different sequencing of actions or different action parameters (i.e. a new path to follow). In this way, the framework allows to incorporate various behaviours to the system with the help of various actions and associated tasks, such as obstacle avoidance, maintaining minimum altitude etc.

3.5 Task Priority Inverse Kinematics

Exploiting the task priority inverse kinematics defined in [59, 63], the desired/reference system velocity vector $\bar{\mathbf{v}}$ that satisfies the task priority requirement can be found. In detail, the following sequence of nested minimization problems needs to be solved, corresponding

to the so called Task Priority Inverse Kinematics (TPIK):

$$S_k \triangleq \left\{ \arg R - \min_{\bar{\mathbf{v}} \in S_{k-1}} \|\mathbf{A}_k(\dot{\mathbf{x}}_k - \mathbf{J}_k \bar{\mathbf{v}})\|^2 \right\}, k = 1, 2, \dots, N, \quad (3.9)$$

where S_{k-1} is the manifold of solutions of all the previous tasks in the hierarchy, $S_0 \triangleq \mathbb{R}^n$, N is the total number of priority levels, and finally where the notation $R - \min$ is introduced for underlining that each minimization is performed in a special regularized form as explained in [59]. Within such a specialized framework (named iCAT: inequality Constraints And Task transitions), the above TPIK problem (3.9) results in the following algorithm 1. The special pseudo-inverse operator $(\cdot)^{\#,\mathbf{A},\mathbf{Q}}$ has been introduced in [59] to cope with certain invariance problems arising in (3.9). The function $\text{Sat}(\cdot)$ implements the management of control variable saturation suggested in [70].

Algorithm 1 Algorithm: Task Priority Inverse Kinematics from iCAT.

Require: $\boldsymbol{\rho}_0 = 0$, $\mathbf{Q}_0 = \mathbf{I}$

Ensure: $\bar{\mathbf{v}} = \boldsymbol{\rho}_N$

while $k = 1$ **to** N **do**

▷ do for each priority level

$$\mathbf{W}_k = \mathbf{J}_k \mathbf{Q}_{k-1} (\mathbf{J}_k \mathbf{Q}_{k-1})^{\#,\mathbf{A}_k,\mathbf{Q}_{k-1}}$$

$$\mathbf{Q}_k = \mathbf{Q}_{k-1} \left(\mathbf{I} - (\mathbf{J}_k \mathbf{Q}_{k-1})^{\#,\mathbf{A}_k,\mathbf{I}} \mathbf{J}_k \mathbf{Q}_{k-1} \right)$$

$$\boldsymbol{\rho}_k = \boldsymbol{\rho}_{k-1} + \text{Sat} \left(\mathbf{Q}_{k-1} (\mathbf{J}_k \mathbf{Q}_{k-1})^{\#,\mathbf{A}_k,\mathbf{I}} \mathbf{W}_k (\dot{\mathbf{x}}_k - \mathbf{J}_k \boldsymbol{\rho}_{k-1}) \right)$$

end while

At the end of the above iterative process, the desired final system velocity (reference) vector is $\bar{\mathbf{v}} = \boldsymbol{\rho}_N$. This can also be further extended to vehicles with manipulators [60, 61, 66–69].

Chapter 4

Unified Framework Design

4.1 Introduction

From the past couple of decades, AUVs are being used widely for a variety of applications including environmental mapping, pipeline inspection, cable inspection, marine geological survey, seabed mapping, polar explorations, and also military roles such as mine hunting. There is a growing interest in the method of control techniques that could work seamlessly for various autonomous underwater vehicles. However there are many related problems that are yet not solved such as autonomous underwater docking. Irrespective of manual launch and recovery of the vehicle, an autonomous launch, docking and recovery will enable the operators with more flexibility in the operations. It greatly reduces manpower, turn around time and increases the safety and scalability of the vehicle. The existing solutions do not take in to account of the possibility of environmental disturbances and other factors such as uneven seafloor (if the docking has to be performed close to the seafloor) etc. Although majority of the state of the art deals on the underwater navigation or control, docking problems are not integrated into the system along with other vehicle behaviours and uses dedicated controllers [71–74]. Thus there is a need for a common framework that can be used to deal with all scenarios of operations for an AUV, simultaneously trying to solve all these problems. [75] derives control laws to steer a marine craft along a given path in the presence of constant (but unknown) ocean currents. To cope with the issue of global attraction to the path, nonlinear control algorithms are used. A nonlinear kinematic controller is then derived for a hypothetical vehicle to steer it to a reference path in the presence of a constant, known ocean current, followed by the design of a linear estimator for the current that yields exponential asymptotic stability of the estimation errors to zero. This derived kinematic controller is extended to incorporate the dynamics by casting the vehicle dynamic

equations into standard integrator form using nonlinear dynamic inversion and applying back-stepping techniques. A 3D path following system for autonomous underwater vehicles based on is proposed in [76]. The presence of complex hydrodynamic terms in the vehicle dynamics are considered. Here a 2D case is extended to a 3D situation by aligning the main body axis of the hypothetical vehicle to the total velocity vector and then steered to its reference path using a nonlinear kinematic controller. This derived kinematic controller is also extended to incorporate the dynamics of the vehicle where the position and attitude of the underwater vehicle, as well as its angles of attack and side slip, are measurable. The problem of straight-line path following for fully actuated marine craft is studied in [77]. The designed controller adjusts the speed of the marine craft according to the geometric distance and the rate of convergence to the path. The control law is derived using the method of least squares, which is used to find an approximate solution for over-determined systems. The conditions under which the closed loop system is globally asymptotically stable are found in this research which also proposes a method to ensure zero cross-track error in the presence of ocean currents. A non-collision trajectory planning algorithm is presented in [78] for three-dimensional space based on velocity potential field for robotic manipulators. This can be applied to collision avoidance among serial industrial robots and obstacles, and path optimization in multi-robot collaborative operation. The algorithm is achieved by planning joint velocities of manipulators based on attractive, repulsive, and tangential velocity of velocity potential field. A path priority strategy of collision avoidance is also considered for better performance and higher efficiency when multi-robots cooperation is scheduled.

As seen in Section 3.1, there is a growing interest in the method of control techniques that could work seamlessly for various autonomous underwater vehicles. A unified approach leveraging from the already available task-priority based control is studied in a simulation setting to solve the common AUV problems, including homing and docking. Task priority based control allows us to tackle a wide range of scenarios by defining actions within the framework. At the kinematic level, it allows greater flexibility and modularity in the capabilities without compromising the performance, but also simplifies the implementation by giving the ability to define tasks that can be used across multiple actions. Various behaviors of the vehicle can be achieved by changing the priorities of these tasks at the kinematic level. One of the main advantage of using this framework is the capability to switch between different behaviors with ease by just changing the priorities of certain tasks without causing any discontinuities. For executing certain maneuvers close to the seafloor or near structures, underwater vehicles would require sophisticated control systems to guide the vehicle around obstacles, maintain a certain altitude or depth, all in the possible presence of

ocean currents. Another interesting fact is that the previously discussed techniques in take only a specific type of behavior into account. Those techniques do not incorporate diverse behaviors, ranging from the execution of the main tasks of the mission (e.g., surveying an area, structure, etc.), to homing and docking at the end of the mission. Thus, apart from homing and docking to stationary and moving docks, additional capabilities such as way point guidance, path following at constant depth or altitude, terrain following, obstacle avoidance and pitch limitation should also be integrated into a unique framework.

Furthermore, just like ASVs, AUVs needs to be docked to surface docking stations such as piers, mother ships, etc. for system inspection or safe retrieval. Autonomous docking of an AUV to surface docking stations can be assumed as a special case of autonomous docking of ASV. In [71], the authors discuss docking of an ASV using linear and circular-based vector field guidance methods to overcome the motion constraints in performing docking to certain regions where it is difficult to dock without collision with structures such as pier walls. A combination of circular and linear vector fields are suggested to drive the vehicle out from the collision region. However, if the vehicle starts too close to the structures, there is a chance of collision as the vehicle might perform some tight maneuvers. Coming to moving docking stations, the authors of [72] integrate no-fly zones with virtual obstacles into the navigation function for avoiding collisions while trying to dock from the stern direction into a docking station hoisted on top of a submarine. A method of recovery for AUVs using a visual docking approach is discussed in [73] where the terminal guidance is performed via two cameras that help in estimating the position of a moving docking station. The position and heading of the AUV is corrected before controlling its vertical movement to finally dock to the docking platform. In their work, the authors consider the moving dock to be in the mid-water column (i.e., in the free space). However, if the dock is close to the seafloor, the terrain may not be smooth. Hence, as the vehicle approaches for docking in such scenarios, it could have unwanted collisions with the seafloor while attempting the docking maneuver. The work in [74] discusses the docking of ASVs into a large mother ship by placing a virtual target to a point positioned at the projection of the ASV position onto a safety circle surrounding the mother ship. This virtual target point can move along the safety circle until its bearing matches up with that of the desired docking point. Once the ASV matches its position and velocity with this virtual target point, the target point starts reducing its distance from the dock by translating, until it becomes equal to that of the docking point. However, the fully lateral translation maneuver presented in the homing and docking stages will be difficult to achieve due to the drag, presence of water currents, and high energy requirements. Further, this chapter also gives an insight on contributions **C1**, **C2** and **C3**.

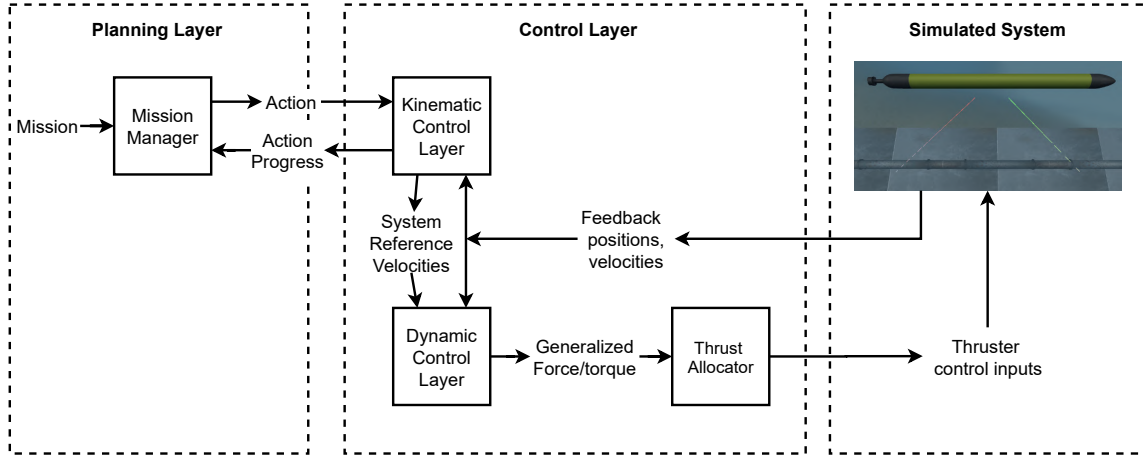


Figure 4.1 Control architecture depicting the Planning Layer, the Control Layer, and the Simulated System.

4.2 General Architecture Overview

A general schematic of the control architecture [30] is provided in Figure 4.1 and is composed by three layers: the Planning Layer, the Control Layer, and the Simulated System. The force that each thruster must exert, which is an input to the Simulated System, is found using the thruster allocation mechanism that maps the desired wrench on the body to the forces that need to be generated by each of the thrusters. Such a desired wrench is the output of the Dynamic Control Layer (DCL), which tracks the desired system velocities. The Kinematic Control Layer (KCL) implements the task priority control framework discussed in 3.2 encompasses the definition of the control tasks, which are the basic building block of the architecture. Within the KCL, basic tasks are arranged into task hierarchies with priorities that constitute the control actions. The KCL receives the action to be executed from the Planning Layer, and it exploits the proposed task priority framework to find out the desired system velocity to accomplish it. The Planning Layer contains the mission manager that supervises and manages the execution of the current mission. It also addresses conflicting tasks by sequencing different actions as discussed in 3.4 or by providing a new set of parameters, thus providing an alternate solution.

4.3 Task definitions

As stated in Contribution C2 under section 1.1, taking into account the various behaviors of underwater vehicles and implement all of these features within the common unified

Task Name	Task Type I		Task Type II	
	Position	Velocity	Equality	Inequality
Vehicle Non Actuation		✓	✓	
Minimum Altitude	✓			✓
Obstacle Avoidance	✓			✓
Velocity Alignment	✓		✓	
Seafloor Alignment	✓			✓
Horizontal Alignment	✓			✓
Follow 2D Path	✓		✓	
Vehicle Depth	✓		✓	
Vehicle Altitude	✓		✓	
Survey Leg Alignment	✓		✓	
Hold Position	✓		✓	
Field Velocity		✓	✓	
Terminal Dock	✓		✓	
Dock Attitude	✓		✓	

Table 4.1 Table depicting various tasks used in this research.

framework of task priority based control. Various tasks are defined in order to implement the behaviours.

Recalling control tasks from Section 3.3, they can be generally divided into position control tasks and velocity control tasks. Table 4.1 depicts the tasks that are used in this research. In the context of application, some tasks can be further categorized into safety tasks and complementary tasks. Safety tasks are important and are used to provide safety for the vehicle while navigating. Complementary tasks on the other side are used to attain a particular behaviour of the vehicle which are necessary for the mission.

4.3.1 Vehicle Non Actuation Task

In cases where the vehicle does not have thrusters that allows control of certain linear or angular velocities, the vehicle underactuation should be taken into account and managed. Considering the torpedo-shaped vehicle presented in Section 2.2, the degree of freedom along the x axis for roll needs to be de-actuated. The reference rate $\dot{\mathbf{x}}_{na}$ is defined by the following equation,

$$\dot{\mathbf{x}}_{na} = \mathbf{v}, \quad (4.1)$$

where $\mathbf{v}$ denotes the measured velocities, including the non-actuated ones, assuming they are measurable.

The activation function is simply a diagonal matrix of the form,

$$a_{na}(x_{na}) = \text{diag}(0, 0, 0, 1, 0, 0), \quad (4.2)$$

selecting the non-actuated velocities.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b J_{na} = \begin{bmatrix} I_{6 \times 6} \end{bmatrix}. \quad (4.3)$$

The final velocity $\bar{\mathbf{v}}$ will have roll velocity equal to the measured value. This allows the lower priority tasks to take into account of the non actuated measured velocities during their computation of task reference rate as shown in Algorithm 1. However, at the end of the procedure, the velocity values corresponding to the non-actuated ones are trivially discarded.

4.3.2 Minimum Altitude Task

Concerning safety tasks, the *Minimum Altitude* Task allows the vehicle to maintain a safe altitude above the seafloor. This prevents the vehicle from crashing into the seafloor or other structures below itself. For this purpose, a control objective is defined with $x_{ma} \geq \bar{x}_{ma}$, where x_{ma} is the current vehicle altitude and $\bar{x}_{ma}$ is the minimum allowed altitude to prevent the vehicle from crashing.

The activation function for this control task is given as,

$$a_{ma}(x_{ma}) = \begin{cases} 1, & x_{ma} < \bar{x}_{ma} \\ s_g(x_{ma}), & \bar{x}_{ma} \leq x_{ma} \leq \bar{x}_{ma} + \Delta_{ma} \\ 0, & x_{ma} > \bar{x}_{ma} + \Delta_{ma} \end{cases}. \quad (4.4)$$

Δ_{ma} allows to define a support zone where the objective is satisfied, but the task is smoothly activated/deactivated. As a result, the task is active only if the distance between the vehicle and the seafloor is less than $\bar{x}_{ma} + \Delta_{ma}$. For other cases, the task will be inactive.

Hence reference rate $\dot{\bar{x}}_{ma}$ is defined as follows,

$$\dot{\bar{x}}_{ma} = \gamma_{ma}(\bar{x}_{ma} + \Delta_{ma} - x_{ma}), \gamma_{ma} > 0, \quad (4.5)$$

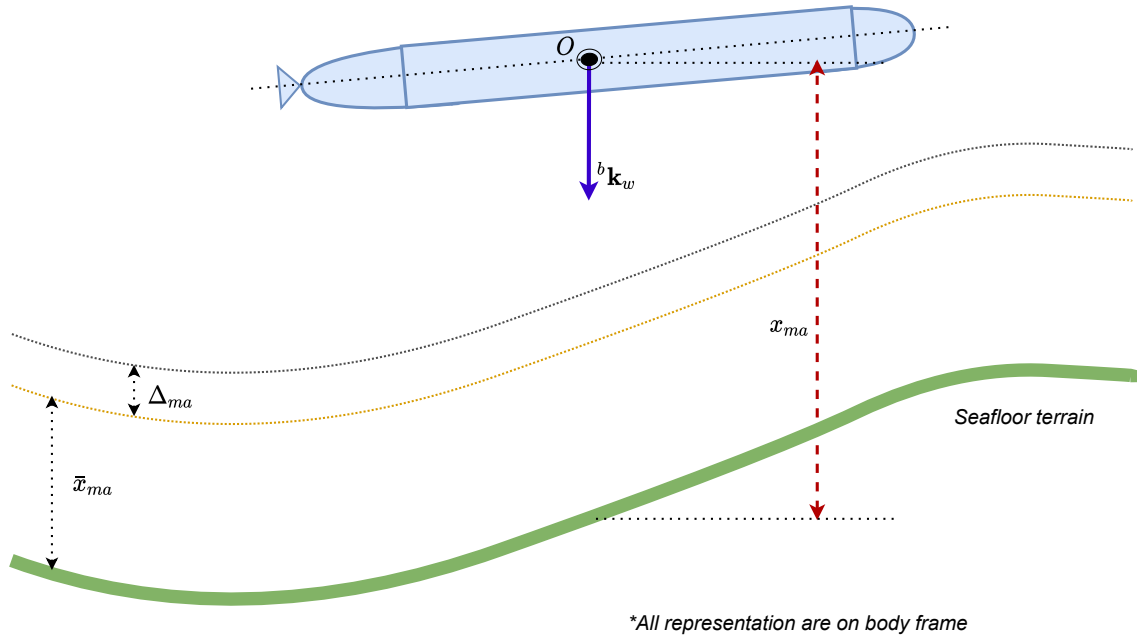


Figure 4.2 Task: Minimum Altitude.

driving the objective towards $\bar{x}_{ma} + \Delta_{ma}$ and allowing the deactivation of the task.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$$^b\mathbf{J}_{ma} = ^b\mathbf{k}_w^\top \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}. \quad (4.6)$$

If the task is satisfied, the final velocity $\bar{\mathbf{v}}$ will have linear components along the projection of world z axis on $\langle b \rangle$, causing the vehicle to move away from the seafloor as depicted in Figure 4.2.

4.3.3 Obstacle Avoidance Task

Due to the varying environments through which the vehicle operates, it will encounter obstacles along its path. Therefore, these obstacles need to be taken into account during the vehicle navigation in order to prevent crashing into them. The *Obstacle Avoidance Task* enables the vehicle to avoid any obstacles along the path. Multiple obstacles are considered, and once they are known with the help of on-board sensors, the task will be able to avoid them in a reactive fashion and proceed towards the goal. Consider a vector $\dot{\bar{\mathbf{x}}}_{oa}$ which is a collection of reference rates for driving the scalar objective of each obstacle. Here for each i^{th} obstacle, a control objective is defined by $x_{oa_i} \geq \bar{x}_{oa_i}$, where x_{oa_i} is the distance of the vehicle from i^{th} obstacle and $\bar{x}_{oa_i}$ is the minimum allowed distance to that i^{th} obstacle.

The activation function for this control task is given as,

$$a_{oa_i}(x_{oa_i}) = \begin{cases} 1, & x_{oa_i} < \bar{x}_{oa_i} \\ s_l(x_{oa_i}), & \bar{x}_{oa_i} \leq x_{oa_i} \leq \bar{x}_{oa_i} + \Delta_{oa_i} \\ 0, & x_{oa_i} > \bar{x}_{oa_i} + \Delta_{oa_i} \end{cases} \quad (4.7)$$

Δ_{oa_i} allows to define a support zone where the objective is satisfied, but the task is smoothly activated/deactivated. As a result, the i^{th} dimension of the task is active only if the distance between the vehicle and the i^{th} obstacle is less than $\bar{x}_{oa_i} + \Delta_{oa_i}$. In other cases, it will be inactive.

Hence the reference rate $\dot{\bar{x}}_{oa_i}$ is defined as follows,

$$\dot{\bar{x}}_{oa_i} = \gamma_{oa}(\bar{x}_{oa_i} + \Delta_{oa_i} - x_{oa_i}), \gamma_{oa} > 0, \quad (4.8)$$

driving the objective towards $\bar{x}_{oa_i} + \Delta_{oa_i}$ and allowing the deactivation of the task.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b\mathbf{J}_{oa_i} = {}^b\mathbf{n}_{oa_i}^\top \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}, \quad (4.9)$$

where $\mathbf{n}_{oa_i}$ is the unit vector radial to the i^{th} obstacle and passing through the vehicle. The final Jacobian ${}^b\mathbf{J}_{oa}$ is obtained by the vertical juxtaposition of the sub-Jacobians ${}^b\mathbf{J}_{oa_i}$ for each of the obstacles present. If the task is satisfied, the final velocity $\bar{\mathbf{v}}$ will have linear components along the unit vector $\mathbf{n}_{oa_i}$, thereby pushing the vehicle away from the obstacle as depicted in Figure 4.3.

4.3.4 Velocity Alignment Task

Moving to complementary tasks, it is desirable for the vehicle to have the behaviour where it can, always or often align to the slope of seafloor terrain, or align to the horizontal, etc. In order to avoid lateral/vertical translation of the vehicle as much as possible, it is desirable for the vehicle to move in the direction of the linear components of the desired final system velocity vector. The linear components of the desired final system velocity vector is represented as $\bar{\mathbf{v}}_l$. The *Velocity Alignment* Task aligns the longitudinal axis $\mathbf{i}_b$ of the vehicle to $\bar{\mathbf{n}}_l$ being the unit vector representing the direction of $\bar{\mathbf{v}}_l$. In this way, lateral translations are limited, allowing the main thruster to perform the majority of the work, increasing the overall efficiency since the vehicle moves in the least damping direction. Let $\mathbf{x}_{val}$ represent

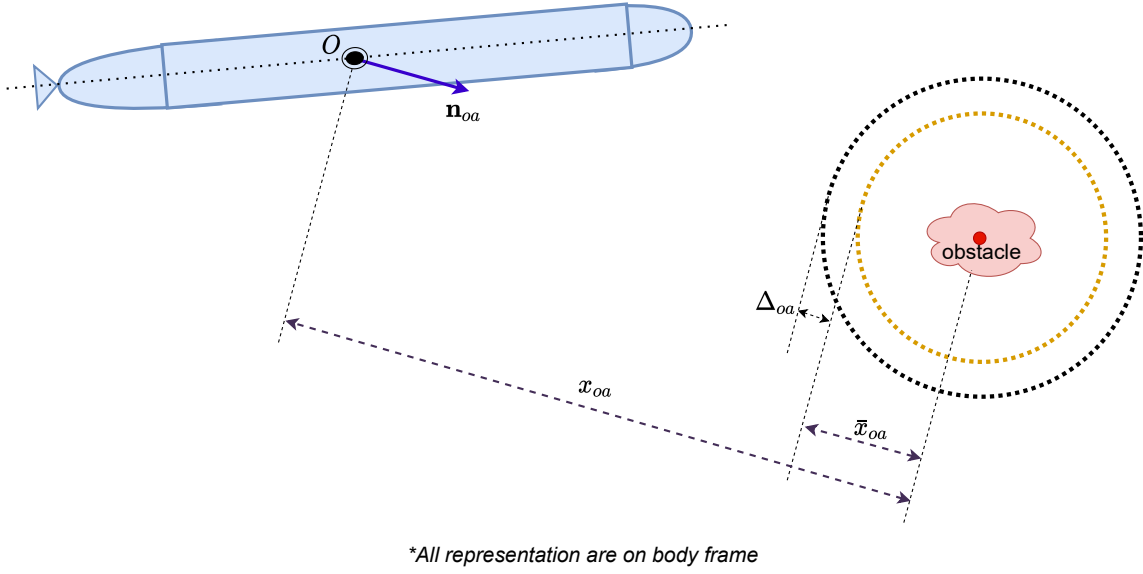


Figure 4.3 Task: Obstacle Avoidance.

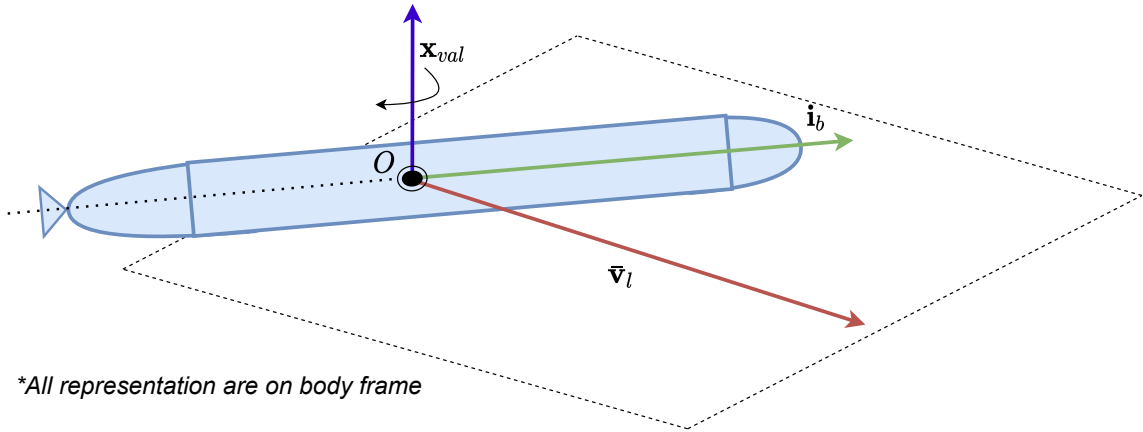
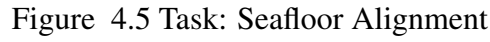


Figure 4.4 Task: Velocity Alignment

the misalignment vector between $\mathbf{i}_b$ and $\bar{\mathbf{v}}_l$ as depicted in Figure 4.4. A multi-dimensional control objective is considered which is defined by $\mathbf{x}_{val} = \bar{\mathbf{x}}_{val} = 0$, where $\bar{\mathbf{x}}_{val}$ is the reference misalignment vector between $\mathbf{i}_b$ and $\bar{\mathbf{v}}_l$. The reference rate $\dot{\bar{\mathbf{x}}}_{val}$ to drive the control objective $\mathbf{x}_{val} \rightarrow \bar{\mathbf{x}}_{val}$ is defined by the following equation,

$$\dot{\bar{\mathbf{x}}}_{val} = \gamma_{val}(\bar{\mathbf{x}}_{val} - \mathbf{x}_{val}), \gamma_{val} > 0. \quad (4.10)$$

The unit vector $\mathbf{n}_l$ representing the direction of $\bar{\mathbf{v}}_l$, computed as $\mathbf{n}_l = \frac{\bar{\mathbf{v}}_l}{|\bar{\mathbf{v}}_l|}$ is ill-defined when the $\bar{\mathbf{v}}_l$ approaches 0 or cease to exist. Therefore the task should be deactivated when $\bar{\mathbf{v}}_l$ goes below a threshold value Δ_{v_l} .


$$a_{val}(|\mathbf{x}_{val}|) = \begin{cases} 1, & |\bar{\mathbf{v}}_l| = 1 \\ s_l(|\mathbf{x}_{val}|), & \Delta_{v_l} \leq |\bar{\mathbf{v}}_l| < 1 \\ 0, & |\bar{\mathbf{v}}_l| < \Delta_{v_l} \end{cases} \quad (4.11)$$

The Jacobian for the task expressed on $\langle b \rangle$ is,

If the task is satisfied, the final velocity $\bar{\mathbf{v}}$ will have rotational components along the direction of vector $\mathbf{x}_{val}$, causing the vehicle to orient its longitudinal axis towards the direction of $\bar{\mathbf{v}}_l$.

The *Seafloor Alignment* Task is a complementary task to the *Minimum Altitude* task, and allows the vehicle to follow the seafloor terrain, thereby its slope. As the objective is to maintain the vehicle parallel the seafloor terrain when it is sufficiently close to the seafloor

itself, it helps the vehicle from crashing to the seafloor while maneuvering over rough terrain. This also simplifies to maintain the altitude of the vehicle. The task is executed by aligning the body z axis vector $\mathbf{k}_b$ to the direction of the seafloor terrain normal vector represented by the unit vector $\mathbf{n}_{st}$, as depicted in Figure 4.5. The unit vector $\mathbf{n}_{st}$ will be further discussed in Section 5.1. To achieve this objective, the kinematics of the task are the following. To obtain a perfect alignment, a control objective defined by $x_{sa} = \bar{x}_{sa} = 0$ is considered, where x_{sa} is the norm of $\mathbf{r}_{sa}$ being the misalignment vector between $\mathbf{k}_b$ and $\mathbf{n}_{st}$, and $\bar{x}_{sa}$ is the reference misalignment norm.

Since the task is considered relevant only in close proximity with the seafloor, the activation function for this control task is not a function of the variable under control of the objective, but instead a function of an external variable altitude x_a which makes the task relevant. The external activation function (refer 3.3) is thus given as,

$$a_{sa}(x_a) = \begin{cases} 1, & x_a < \bar{x}_a \\ s_l(x_a), & \bar{x}_a \leq x_a \leq \bar{x}_a + \Delta_a \\ 0, & x_a > \bar{x}_a + \Delta_a \end{cases}, \quad (4.13)$$

where $\bar{x}_a$ is the minimum altitude below which the vehicle should follow the terrain of the seafloor. Δ_a allows to define a support zone where the objective is satisfied, but the task is smoothly activated/deactivated. As a result, the task is active only if the altitude of the vehicle x_a is less than than $\bar{x}_a + \Delta_a$. For other cases, the task will be inactive.

Hence the reference rate $\dot{\bar{x}}_{sa}$ is defined as,

$$\dot{\bar{x}}_{sa} = \gamma_{sa}(\bar{x}_{sa} - x_{sa}), \gamma_{sa} > 0, \quad (4.14)$$

driving the objective towards $\bar{x}_{sa}$ and allowing the deactivation of the task.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b \mathbf{J}_{sa} = {}^b \mathbf{n}_{sa}^\top \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix}, \quad (4.15)$$

where $\mathbf{n}_{sa}$ is the unit vector representing the direction of the misalignment vector $\mathbf{r}_{sa}$, computed as $\mathbf{n}_{sa} = \frac{\mathbf{r}_{sa}}{|\mathbf{r}_{sa}|}$ and is normal to the plane produced by $\mathbf{k}_b$ and $\mathbf{n}_{st}$. If the task is satisfied, the final velocity $\dot{\mathbf{v}}$ will have rotational components along the unit vector $\mathbf{n}_{sa}$, causing the vehicle to orient its z axis towards the direction represented by $\mathbf{n}_{st}$.

4.3.6 Horizontal Alignment Task

Furthermore, it is always convenient not to allow the vehicle to reach high values of pitch in order to prevent it from reaching unstable configurations. The *Horizontal Alignment Task* allows the vehicle to control its pitch so that it will not exceed its permissible limits. The task aligns the body z axis vector $\mathbf{k}_b$ to the unit vector $\mathbf{k}_w$ normal to the horizontal plane as depicted in Figure 4.6. A control objective defined by $x_{ha} \leq \bar{x}_{ha}$ is considered, where x_{ha} is the norm of $\mathbf{r}_{ha}$ being the misalignment vector between $\mathbf{k}_b$ and $\mathbf{k}_w$, and $\bar{x}_{ha}$ is the norm of the maximum misalignment in pitch allowed for the vehicle with respect to the horizontal plane (assuming negligible roll).

The activation function for this control task is given as,

$$a_{ha}(x_{ha}) = \begin{cases} 1, & x_{ha} > \bar{x}_{ha} \\ s_g(x_{ha}), & \bar{x}_{ha} - \Delta_{ha} \leq x_{ha} \leq \bar{x}_{ha} \\ 0, & x_{ha} < \bar{x}_{ha} - \Delta_{ha} \end{cases}, \quad (4.16)$$

Δ_{ha} allows to define a support zone where the objective is satisfied, but the task is smoothly activated/deactivated. As a result, the task is active only if x_{ha} is more than $\bar{x}_{ha} - \Delta_{ha}$. For other cases, the task will be inactive.

Hence the reference rate $\dot{\bar{x}}_{ha}$ is defined as follows,

$$\dot{\bar{x}}_{ha} = \gamma_{ha}(\bar{x}_{ha} - \Delta_{ha} - x_{ha}), \gamma_{ha} > 0, \quad (4.17)$$

driving the objective towards $\bar{x}_{ha} - \Delta_{ha}$ and allowing the deactivation of the task.

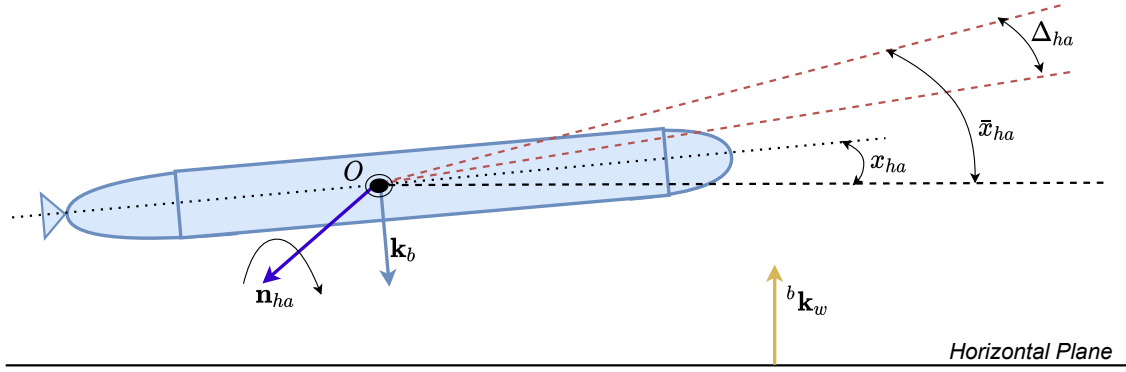
The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b\mathbf{J}_{ha} = {}^b\mathbf{n}_{ha}^\top \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix}, \quad (4.18)$$

where $\mathbf{n}_{ha}$ is the unit vector representing the direction of the misalignment vector $\mathbf{r}_{ha}$, computed as $\mathbf{n}_{ha} = \frac{\mathbf{r}_{ha}}{|\mathbf{r}_{ha}|}$ and is normal to the plane produced by $\mathbf{k}_b$ and $\mathbf{k}_w$. If the task is satisfied, the final velocity $\bar{\mathbf{v}}$ will have rotational components along the unit vector $\mathbf{n}_{ha}$, causing the vehicle to orient its z axis towards the direction represented by $\mathbf{k}_w$.

4.3.7 Follow 2D Path Task

In terms of general mission tasks, the need for the vehicle to follow a certain path in 2D is also considered. For instance, a typical surveying mission might require covering an area



*All representation are on body frame

Figure 4.6 Task: Horizontal Alignment.

by moving in lawn-mowing paths between way-points. The *Follow 2D Path* Task allows the vehicle to achieve this by implementing the Line of Sight (LOS) [79] path following technique, reducing the along-track and cross-track errors to 0. Let $\mathbf{x}_{fpl}$ be the current position of the vehicle and $\bar{\mathbf{x}}_{fpl}$ be the reference LOS position for the vehicle to follow the LOS path, where the LOS position error $\bar{\mathbf{x}}_{fpl} - \mathbf{x}_{fpl}$ is represented by $\mathbf{e}_{fpl}$ as depicted in Figure 4.7. Let $\mathbf{x}_{fpa}$ represent the misalignment vector between the vehicle longitudinal axis $\mathbf{i}_b$ and $\mathbf{e}_{fpl}$. The vehicle needs to generate linear velocities along the direction of $\mathbf{e}_{fpl}$ and also angular velocities to align the heading to $\mathbf{e}_{fpl}$. Hence two control objectives are defined with $\mathbf{x}_{fpl} = \bar{\mathbf{x}}_{fpl}$ and $\mathbf{x}_{fpa} = \bar{\mathbf{x}}_{fpa} = 0$, where $\bar{\mathbf{x}}_{fpa}$ is the reference misalignment vector between $\mathbf{i}_b$ and $\mathbf{e}_{fpl}$. The reference rates $\dot{\bar{\mathbf{x}}}_{fpl}$ and $\dot{\bar{\mathbf{x}}}_{fpa}$ to achieve the control objectives are defined by the following equations,

$$\begin{aligned}\dot{\bar{\mathbf{x}}}_{fpl} &= \gamma_{fpl}(\bar{\mathbf{x}}_{fpl} - \mathbf{x}_{fpl}), \gamma_{fpl} > 0 \\ \dot{\bar{\mathbf{x}}}_{fpa} &= \gamma_{fpa}(\bar{\mathbf{x}}_{fpa} - \mathbf{x}_{fpa}), \gamma_{fpa} > 0.\end{aligned}\tag{4.19}$$

The activation function for this control task is given as,

$$\begin{aligned}a_{fpl}(|\mathbf{x}_{fpl}|) &= 1 \\ a_{fpa}(|\mathbf{x}_{fpa}|) &= 1.\end{aligned}\tag{4.20}$$

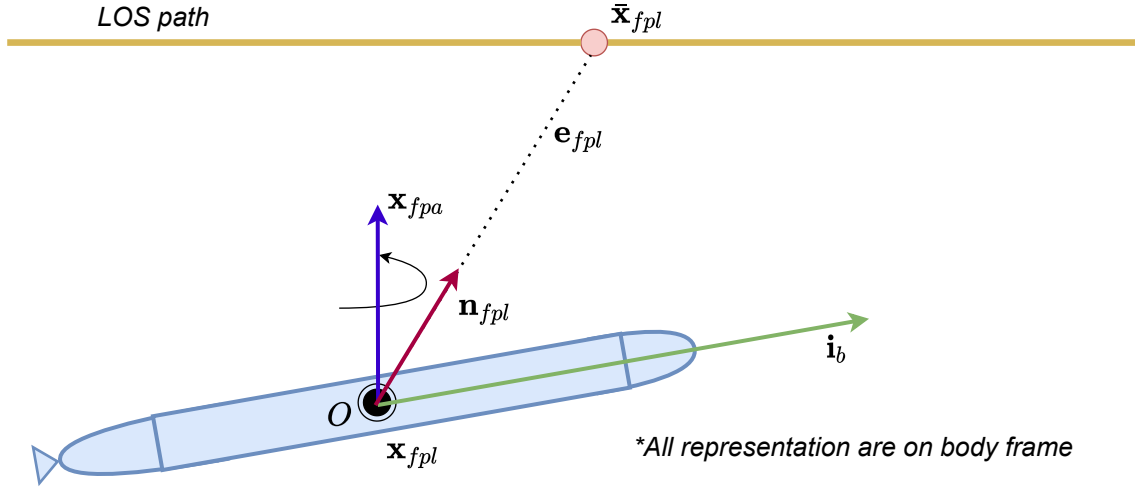


Figure 4.7 Task: Follow 2D Path.

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b J_{fp} = \begin{bmatrix} {}^b n_{fpl}^\top & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{1 \times 3} & {}^b n_{fpa}^\top \end{bmatrix}, \quad (4.21)$$

where $\mathbf{n}_{fpl}$ represents the unit vector in the direction of the error vector $\mathbf{e}_{fpl}$, computed as $\mathbf{n}_{fpl} = \frac{\mathbf{e}_{fpl}}{|\mathbf{e}_{fpl}|}$ and $\mathbf{n}_{fpa}$ represents the unit vector in the direction of the misalignment vector $\mathbf{x}_{fpa}$, computed as $\mathbf{n}_{fpa} = \frac{\mathbf{x}_{fpa}}{|\mathbf{x}_{fpa}|}$ which is normal to the plane formed $\mathbf{i}_b$ and $\mathbf{e}_{fpl}$. The final velocity $\dot{\mathbf{v}}$ will have linear velocity components along $\mathbf{n}_{fpl}$ and rotational components along $\mathbf{n}_{fpa}$, allowing the vehicle to approach the LOS reference position and gradually aligned to the direction of the LOS path.

4.3.8 Vehicle Depth Task

A *Follow 2D Path* Task can be combined with a *Vehicle Depth* Task, if the survey needs to be conducted at a constant depth. Unlike the *Minimum altitude* task that was discussed before, the *Vehicle Depth* task always keep the vehicle depth close to a reference value. Here, a control objective defined by $x_{vd} = \bar{x}_{vd}$ is considered where x_{vd} is the current depth of the vehicle and $\bar{x}_{vd}$ is the reference vehicle depth for the task. The reference rate $\dot{\bar{x}}_{vd}$ to drive the control objective $x_{vd} \rightarrow \bar{x}_{vd}$ is defined by the following equation,

$$\dot{\bar{x}}_{vd} = \gamma_{vd}(\bar{x}_{vd} - x_{vd}), \gamma_{vd} > 0, \quad (4.22)$$

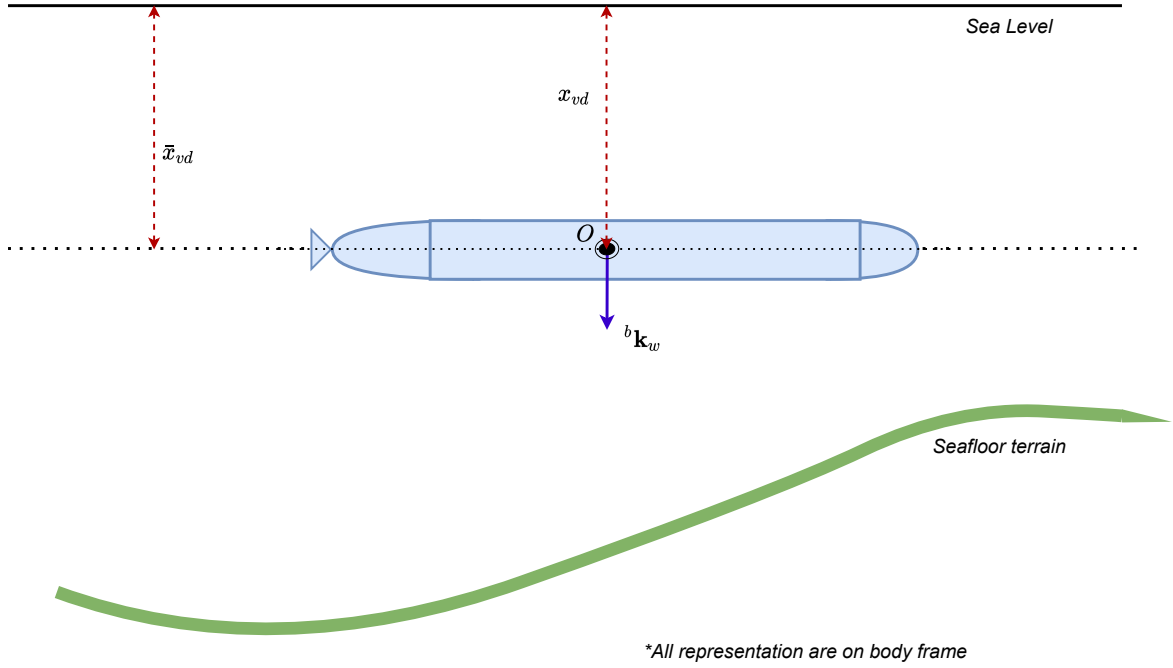


Figure 4.8 Task: Vehicle Depth.

where $\bar{x}_{vd} - x_{vd}$ represents the error between the current vehicle depth and the reference depth as depicted in Figure 4.8. The activation function for this control task is given as,

$$a_{vd}(x_{vd}) = 1. \quad (4.23)$$

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b J_{vd} = {}^b \mathbf{k}_w^\top \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}. \quad (4.24)$$

The final velocity $\bar{\mathbf{v}}$ will have linear components along the projection of world z axis on $\langle b \rangle$, causing the vehicle to stay at the required depth.

4.3.9 Vehicle Altitude Task

If the survey needs to be conducted at a constant altitude from the seafloor, a *Follow 2D Path* Task can be combined with a *Vehicle Altitude* Task. Again, unlike the *Minimum altitude* task that was discussed before, the *Vehicle Altitude* tasks always keep the vehicle altitude close to a reference value. Consider a control objective defined by $x_{va} = \bar{x}_{va}$ is considered where x_{va} is the current vehicle altitude and $\bar{x}_{va}$ is the reference vehicle altitude for the task.

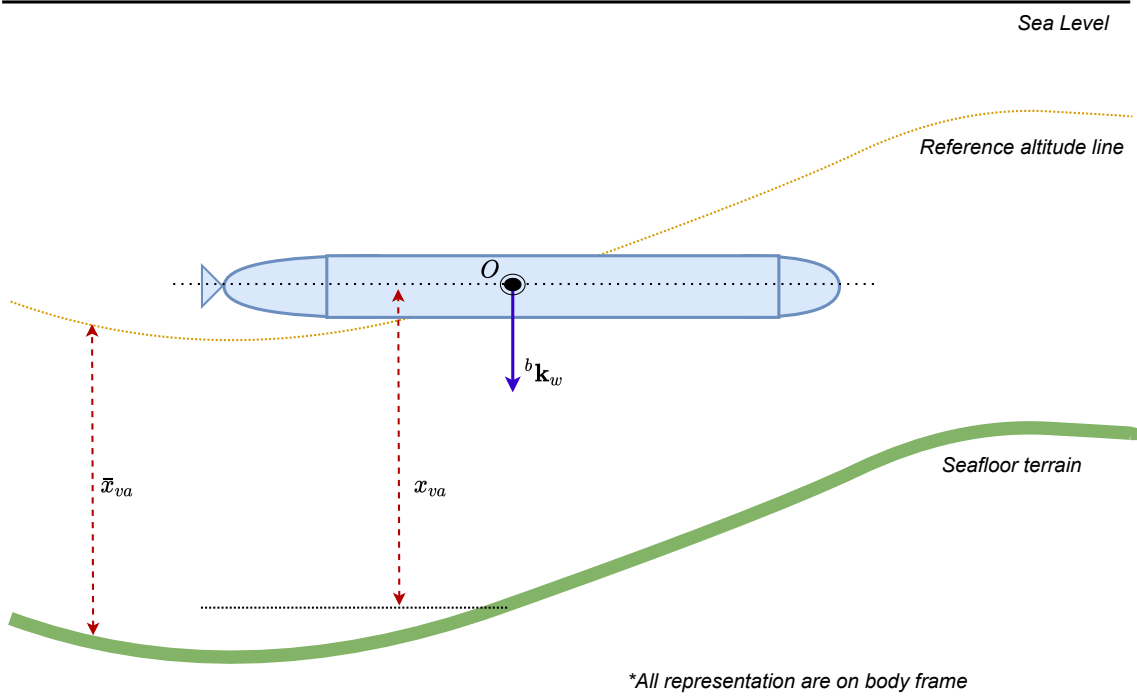


Figure 4.9 Task: Vehicle Altitude.

The reference rate $\dot{\bar{x}}_{va}$ to drive the control objective $x_{va} \rightarrow \bar{x}_{va}$ is defined by the following equation,

$$\dot{\bar{x}}_{va} = \gamma_{va}(\bar{x}_{va} - x_{va}), \gamma_{va} > 0, \quad (4.25)$$

where $\bar{x}_{va} - x_{va}$ represents the error between the current vehicle altitude and the reference altitude as depicted in Figure 4.9. The activation function for this control task is given as,

$$a_{va}(x_{va}) = 1. \quad (4.26)$$

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b J_{va} = {}^b \mathbf{k}_w^\top \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}. \quad (4.27)$$

The final velocity $\bar{\mathbf{v}}$ will have linear components along the projection of z axis on $\langle b \rangle$, causing the vehicle to stay at the required altitude above the seafloor.

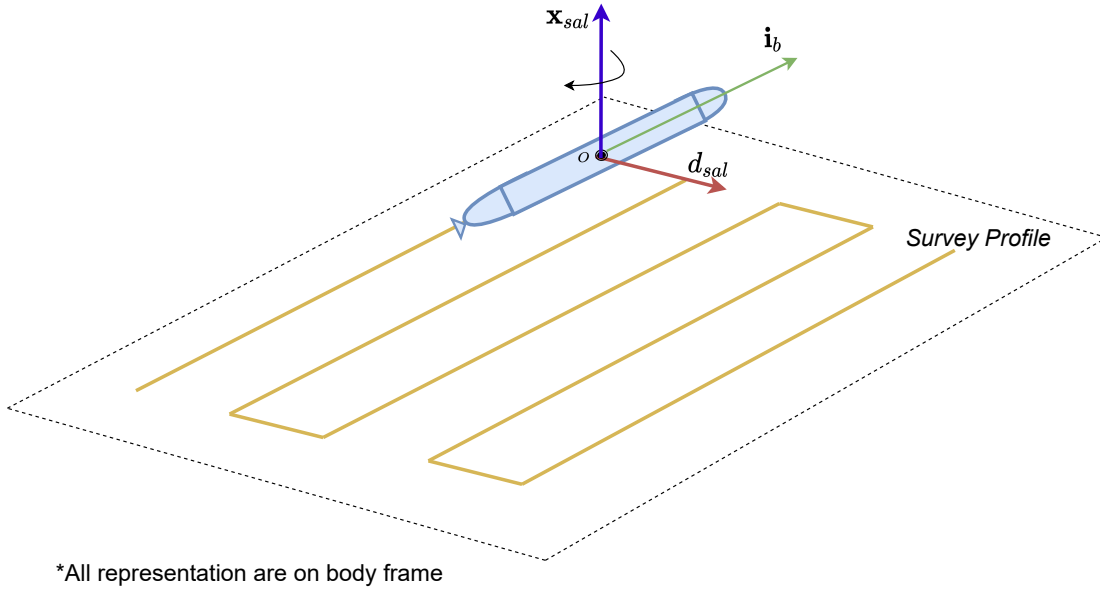


Figure 4.10 Task: Survey Leg Alignment.

4.3.10 Survey Leg Alignment Task

Further, once the AUV has reached the end of one leg of the survey, *Survey Leg Alignment* Task is considered to control the overall orientation of the vehicle, aligning to the next leg before proceeding further with the path following technique in *Follow 2D Path* Task. Here, the vehicle longitudinal axis $\mathbf{i}_b$ needs to align to the vector $\mathbf{d}_{sal}$ representing the reference orientation for the vehicle to align to the next leg of the survey as depicted in Figure 4.10. For the *Survey Leg Alignment* Task, a multi-dimensional control objective defined by $\mathbf{x}_{sal} = \bar{\mathbf{x}}_{sal} = 0$ is considered where $\mathbf{x}_{sal}$ is the misalignment vector between $\mathbf{i}_b$ and $\mathbf{d}_{sal}$, and $\bar{\mathbf{x}}_{sal}$ is the reference misalignment vector between $\mathbf{i}_b$ and $\mathbf{d}_{sal}$. The multi-dimensional reference rate $\dot{\bar{\mathbf{x}}}_{sal}$ to drive the control objective $\mathbf{x}_{sal} \rightarrow \bar{\mathbf{x}}_{sal}$ is defined by the following equation,

$$\dot{\bar{\mathbf{x}}}_{sal} = \gamma_{sal}(\bar{\mathbf{x}}_{sal} - \mathbf{x}_{sal}), \gamma_{sal} > 0. \quad (4.28)$$

The activation function for this control task is given as,

$$a_{sal}(|\mathbf{x}_{sal}|) = 1. \quad (4.29)$$

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b\mathbf{J}_{sal} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix}. \quad (4.30)$$

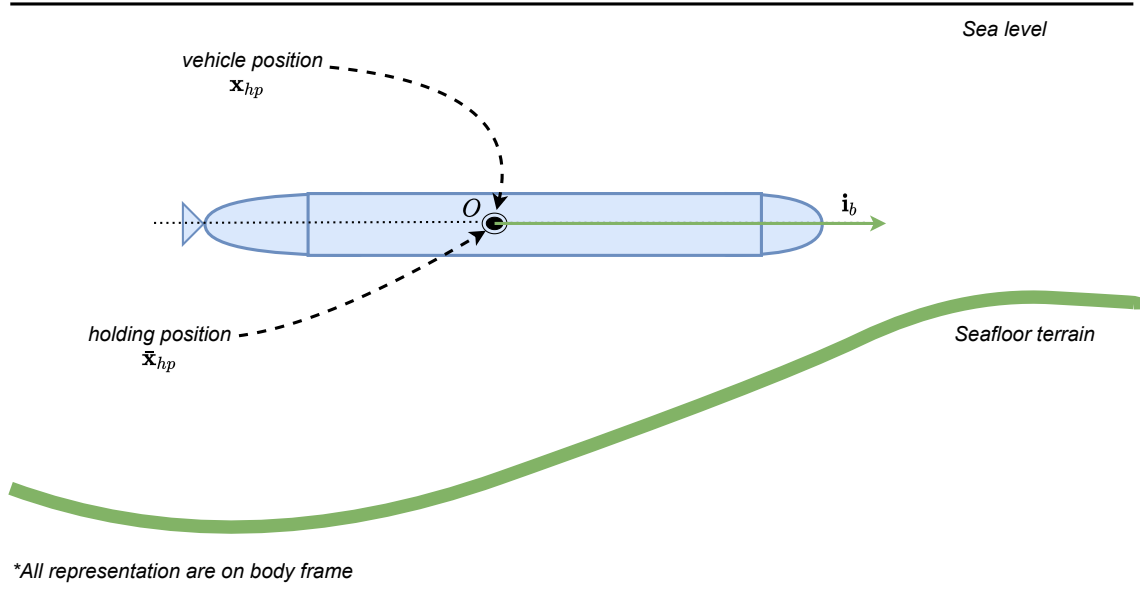


Figure 4.11 Task: Hold Position.

The final velocity $\bar{\mathbf{v}}$ will have rotational components along the direction of $\mathbf{x}_{sal}$, causing the vehicle to orient its longitudinal axis towards $\mathbf{d}_{sal}$. This task being a multi-dimensional would allow correction of both pitch and yaw orientations to match the reference survey leg orientation while executing alignment.

4.3.11 Hold Position Task

In the *Hold Position* Task, for holding the current position of the vehicle while aligning to the next leg of the survey, a multi-dimensional control objective defined by $\mathbf{x}_{hp} = \bar{\mathbf{x}}_{hp}$ is considered where $\mathbf{x}_{hp}$ is the current position of the vehicle and $\bar{\mathbf{x}}_{hp}$ is the reference holding position of the vehicle while performing the *Survey Leg Alignment* Task as depicted in Figure 4.11. The reference rate $\dot{\bar{\mathbf{x}}}_{hp}$ to drive the control objective $\mathbf{x}_{hp} \rightarrow \bar{\mathbf{x}}_{hp}$ is defined by the following equation,

$$\dot{\bar{\mathbf{x}}}_{hp} = \gamma_{hp}(\bar{\mathbf{x}}_{hp} - \mathbf{x}_{hp}), \gamma_{hp} > 0, \quad (4.31)$$

where $\bar{\mathbf{x}}_{hp} - \mathbf{x}_{hp}$ represents the position error $\mathbf{e}_{hp}$ between the current vehicle position and the reference holding position. The activation function for this control task is given as,

$$a_{hp}(|\mathbf{x}_{hp}|) = 1. \quad (4.32)$$

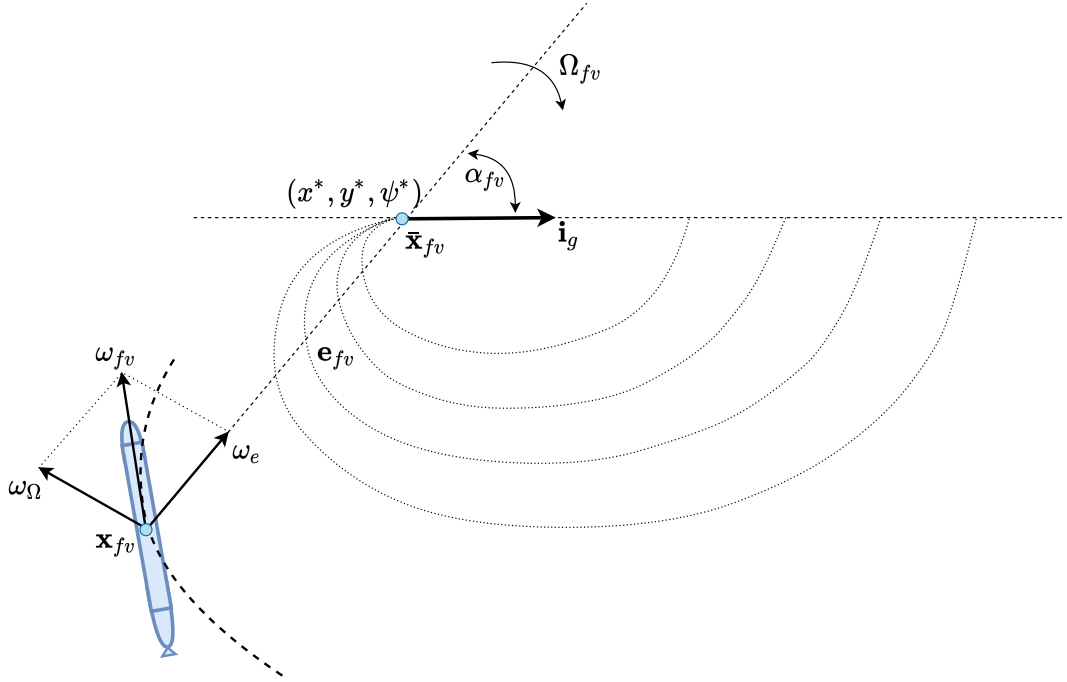


Figure 4.12 Velocity field-based approach in 2D planar scenario.

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b J_{hp} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix}. \quad (4.33)$$

The final velocity $\bar{\mathbf{v}}$ will have linear components along the unit vector $\mathbf{n}_{hp}$ in the direction of error vector $\mathbf{e}_{hp}$, computed as $\mathbf{n}_{hp} = \frac{\mathbf{e}_{hp}}{|\mathbf{e}_{hp}|}$ causing the vehicle to hold its position. This is used as a complementary task while performing other tasks such as the *Survey Leg Alignment* Task.

4.3.12 Field Velocity Task

The *Field Velocity* Task employs the velocity field control discussed in [80, 81], recalled in the following. Consider a planar parking problem with a goal at the planar absolute coordinates and absolute yaw angle represented by given and/or measured x^*, y^*, ψ^* , as depicted in Figure 4.12. Then, for constructing a planar horizontal velocity field asymptotically converging towards the goal with the required orientation approach, the linear and angular velocities that are required to drive the position error vector $\mathbf{e}_{fv}$ between the current vehicle position $\mathbf{x}_{fv}$ and reference goal position $\bar{\mathbf{x}}_{fv}$ (defined in Figure 4.12) towards 0 are of the

form

$$\begin{cases} \boldsymbol{\Omega}_{fv} \doteq \lambda_{fv} \boldsymbol{\alpha}_{fv} \\ \mathbf{w}_{fv} \doteq \gamma_{fv} \mathbf{e}_{fv} + (\mathbf{e}_{fv} \wedge \boldsymbol{\Omega}_{fv}) \\ \doteq \mathbf{w}_e + \mathbf{w}_{\Omega} \end{cases}, \quad (4.34)$$

where $\boldsymbol{\alpha}_{fv}$ represents the misalignment vector between the position error vector $\mathbf{e}_{fv}$ and goal orientation vector $\mathbf{i}_g$. The linear and angular gains denoted by γ_{fv} and λ_{fv} can be modulated for consequently modulating the rate of convergence of the field by using a parameter σ_{fv} as $\lambda_{fv} = \sigma_{fv} \bar{\lambda}_{fv}$ and $\gamma_{fv} = \sigma_{fv} \bar{\gamma}_{fv}$. Corresponding to the current position of the vehicle, the associated velocity $\mathbf{w}_{fv}$ consequently represents the velocity reference to be tracked by the origin of the vehicle frame $\langle b \rangle$, as well as the heading to be tracked by the same vehicle frame. Now, a multi-dimensional control objective defined by $\mathbf{x}_{fv} = \bar{\mathbf{x}}_{fv}$ is considered while performing the *Field Velocity* Task. The reference rate to achieve the corresponding control objectives are defined as,

$$\dot{\bar{\mathbf{x}}}_{fv} = \mathbf{w}_{fv}. \quad (4.35)$$

Moreover, for definitely converging towards a moving docking station, the reference rate will become,

$$\dot{\bar{\mathbf{x}}}_{fv} = \mathbf{w}_{fv} + \mathbf{w}_{ds}, \quad (4.36)$$

where $\mathbf{w}_{ds}$ is the linear velocity of the docking station. For simplicity, it is assumed that the heading of the docking station remains constant as it is typically the case. $\dot{\bar{\mathbf{x}}}_{fv}$ allows the vehicle to do the velocity field-based movement, under the assumption that the heading of the vehicle is always aligned with the direction of the desired final system velocity $\bar{\mathbf{v}}$. This is accomplished by coupling the *Field Velocity* Task with the *Velocity Alignment* Task whose reference for the alignment is the direction of the unit vector representing the linear components of $\bar{\mathbf{v}}$. The activation function for this control task is given as,

$$a_{fv}(|\mathbf{x}_{fv}|) = 1. \quad (4.37)$$

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b\mathbf{J}_{fv} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}. \quad (4.38)$$

The final velocity $\bar{\mathbf{v}}$ will have linear and rotational components, allowing the vehicle to approach the goal position and orientation while performing the *Field Velocity* Task, coupled with the *Velocity Alignment* Task.

4.3.13 Terminal Dock Task

Terminal Dock Task allows the vehicle to execute a final short surge or lateral/vertical translation for achieving docking on to the docking station. Here a control objective with $\mathbf{x}_{td} = \bar{\mathbf{x}}_{td}$ is defined where $\mathbf{x}_{td}$ is the current position of the vehicle and $\bar{\mathbf{x}}_{td}$ is the reference dock position for docking the vehicle on to the docking station. The reference rate $\dot{\bar{\mathbf{x}}}_{td}$ to drive the control objective $\mathbf{x}_{td} \rightarrow \bar{\mathbf{x}}_{td}$ is defined by the following equation,

$$\dot{\bar{\mathbf{x}}}_{td} = \gamma_{td}(\bar{\mathbf{x}}_{td} - \mathbf{x}_{td}), \gamma_{td} > 0, \quad (4.39)$$

where $\bar{\mathbf{x}}_{td} - \mathbf{x}_{td}$ represents the error vector $\mathbf{e}_{td}$ between the current vehicle position and the reference dock position as depicted in Figure 4.13. The activation function for this control task is given as,

$$a_{td}(|\mathbf{x}_{td}|) = 1. \quad (4.40)$$

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b\mathbf{J}_{td} = \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}. \quad (4.41)$$

The final velocity $\bar{\mathbf{v}}$ will have linear velocity components along the unit vector $\mathbf{n}_{td}$ in the direction of the error vector $\mathbf{e}_{td}$, computed as $\mathbf{n}_{td} = \frac{\mathbf{e}_{td}}{|\mathbf{e}_{td}|}$ allowing the vehicle to approach the required dock position while performing the *Terminal Dock Task*.

4.3.14 Dock Attitude Task

While docking, to complement the terminal docking maneuver, the *Dock Attitude Task* is used for maintaining the vehicle orientation to the required dock orientation while doing the final *Terminal Dock Task*. This is executed by aligning the vehicle longitudinal axis $\mathbf{i}_b$ to the direction of the reference dock attitude vector $\mathbf{d}_{da}$ as depicted in Figure 4.14. For the *Dock Attitude Task*, a control objective defined by $x_{da} = \bar{x}_{da} = 0$ is considered where x_{da} is the norm of $\mathbf{r}_{da}$ being the misalignment vector between $\mathbf{i}_b$ and $\mathbf{d}_{da}$, and $\bar{x}_{da}$ is the reference misalignment norm between $\mathbf{i}_b$ and $\mathbf{d}_{da}$. The reference rate $\dot{\bar{x}}_{da}$ to drive the control objective $x_{da} \rightarrow \bar{x}_{da}$ is defined by the following equation,

$$\dot{\bar{x}}_{da} = \gamma_{da}(\bar{x}_{da} - x_{da}), \gamma_{da} > 0. \quad (4.42)$$

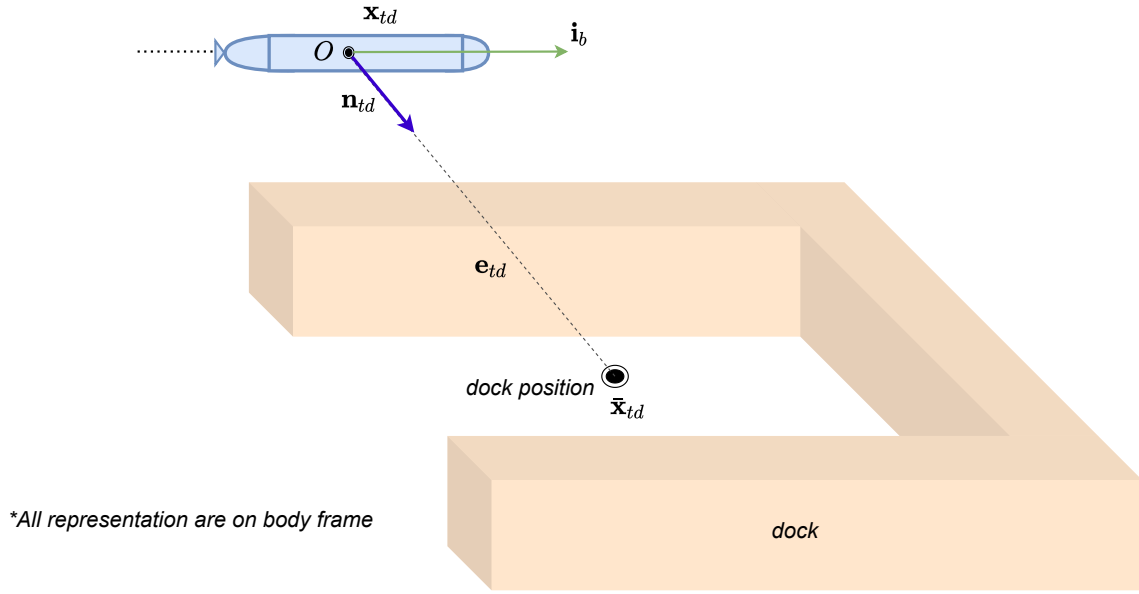


Figure 4.13 Task: Terminal Dock.

The activation function for this control task is given as,

$$a_{da}(x_{da}) = 1. \quad (4.43)$$

As a result, the task is always active.

The Jacobian for the task expressed on $\langle b \rangle$ is,

$${}^b J_{da} = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \end{bmatrix}. \quad (4.44)$$

The final velocity $\bar{\mathbf{v}}$ will have rotational components along the unit vector $\mathbf{n}_{da}$ representing the direction of the misalignment vector $\mathbf{r}_{da}$, computed as $\mathbf{n}_{da} = \frac{\mathbf{r}_{da}}{|\mathbf{r}_{da}|}$ allowing the vehicle to orient its longitudinal axis towards $\mathbf{d}_{da}$.

4.4 Action based framework design

This section provides an in-depth look on how the various behaviours of our system are achieved by combining control tasks for various complex scenarios, exploiting the features of unified task priority framework. These behaviours are crucial in defining the actions, associated tasks and hierarchy. Take an example where the vehicle is doing a survey mission at constant altitude using *Vehicle Altitude* Task. If the vehicle encounters an obstacle, in order to safely navigate around the obstacle, the *Obstacle Avoidance* Task needs to be present at a

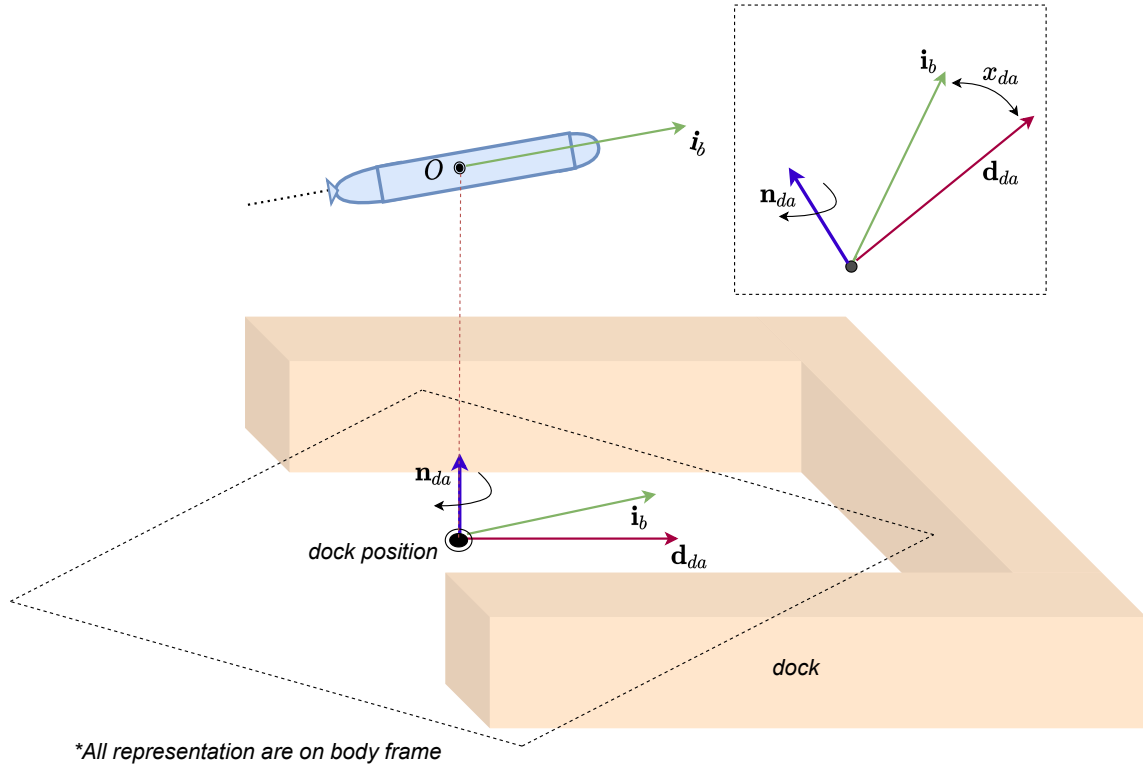
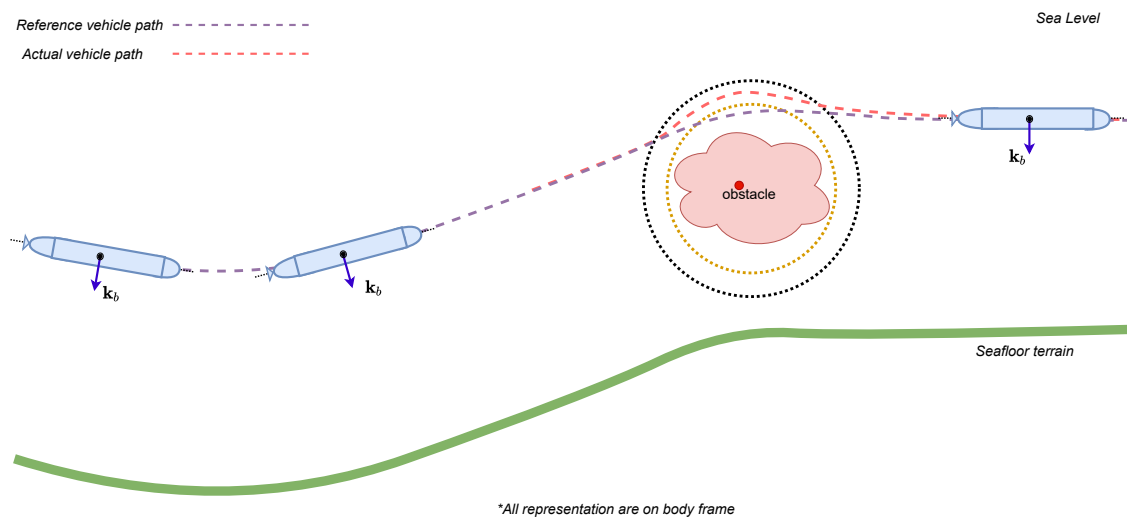
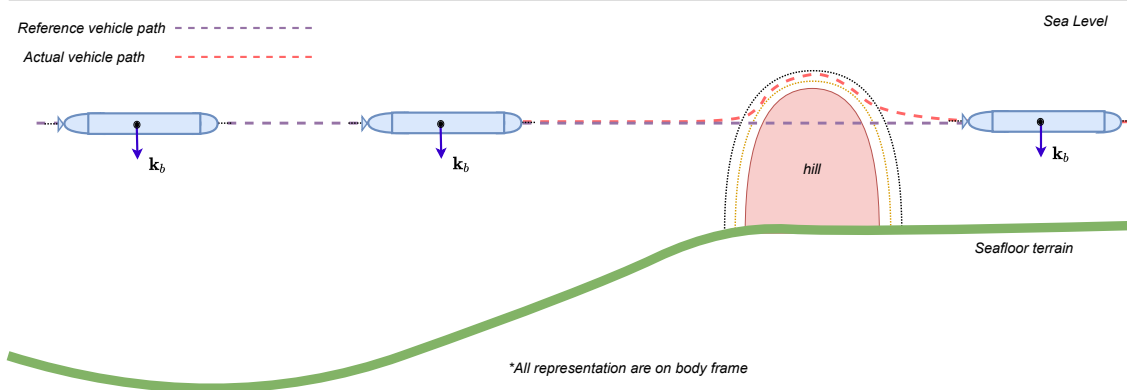
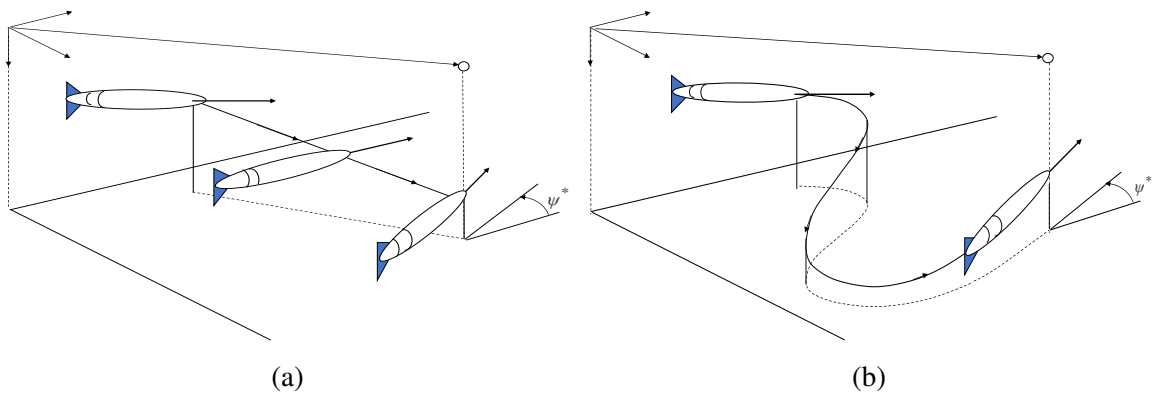


Figure 4.14 Task: Dock Attitude.

higher priority level in the action. This safety task would allow the vehicle to reactively avoid the obstacle and since the task is only active in the near vicinity of the obstacle, the vehicle can continue the constant altitude survey once it is at a safe distance from the obstacle as depicted in Figure 4.15.

Another case where multiple tasks combine to generate a behaviour is in the case of constant depth survey using *Vehicle Depth* Task where the it might encountered a hill along its path. In this scenario, to prevent the vehicle from crashing into the hill, the *Minimum Altitude* Task needs to be included at a higher priority in that particular action. This enables the vehicle to keep its altitude above a certain level as it approaches the hill. Once the hill is cleared, the task would get inactive and the vehicle is allowed to continue its constant depth path as depicted in Figure 4.16.

Considering homing and docking maneuvers, recall that they can be subdivided into a long-range parking (homing) and a short-range parking (terminal docking) as depicted in Figure 4.17. The latter can be realized by explicitly taking advantage of the holonomic capabilities exhibited by the vehicle within the execution of short range and slow final docking maneuver. For the former maneuver, as already stated, it is instead advisable to reduce lateral translations as much as possible that will otherwise lead to unnecessary energy

Figure 4.15 Maneuver with *Vehicle Altitude* task and *Obstacle Avoidance* Task.Figure 4.16 Maneuver with *Vehicle Depth* task and *Minimum Altitude* Task.Figure 4.17 Parking maneuvers: **a)** short-range and **b)** long-range distances.

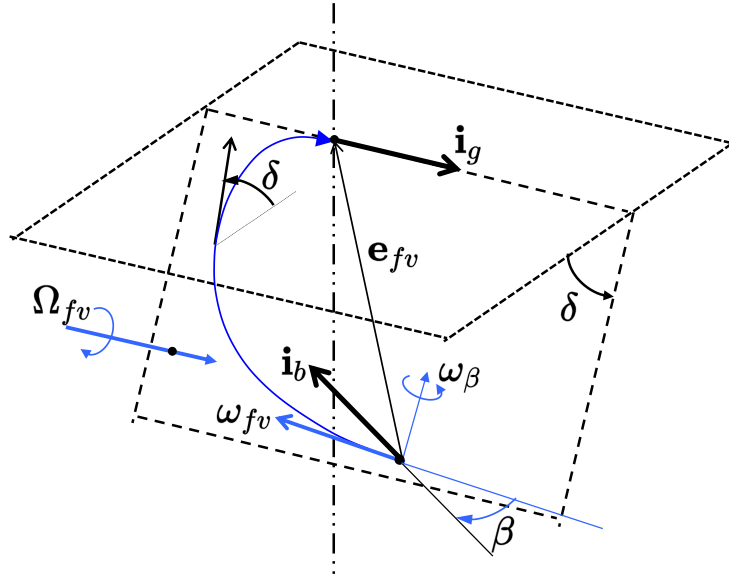


Figure 4.18 Velocity field-based approach in 3D scenario.

consumption. Moreover, as our vehicle is also required to follow the reference velocity by aligning its heading in the velocity direction, let us consider a case when the vehicle is initially misaligned at an angle of β with the reference velocity vector as depicted in Figure 4.18. An appropriate angular velocity $\omega_\beta = \kappa_\beta \beta \mathbf{n}_\beta$ must be applied to reduce the misalignment, where κ_β is the gain required to align to the reference velocity vector and $\mathbf{n}_\beta$ is the unit vector normal to the plane containing ω_β . Recalling the *Velocity Alignment* Task described earlier, it can also be used to achieve such required alignment. Thus, combining the *Field Velocity* and *Velocity Alignment* as current references to be tracked, in case of a reliable tracking of both, will allow the vehicle to successfully carry out homing/parking with minimum energy requirements, as the vehicle will eventually move along its minimal drag direction. While extending the 2D planar parking maneuver to a 3D parking maneuver, the 3D parking problem can be reduced to a 2D planar parking problem where the linear field velocity always lie on the plane formed between the linear error vector $\mathbf{e}$ and goal orientation vector $\mathbf{i}_g$ as depicted in Figure 4.12. Thus in order to perform this non-holonomic long range parking, it is possible by combining the *Field Velocity* Task and *Velocity Alignment* Task. The parking maneuver can also be utilized for missions where the vehicle need to approach the initial way point after being deployed from the docking station.

For the very short-range terminal docking, the holonomic behavior of the vehicle can be exploited using a surge or lateral/vertical translation maneuver at the final step. As the lateral/vertical translation maneuver is used only if absolutely necessary, this can eliminate

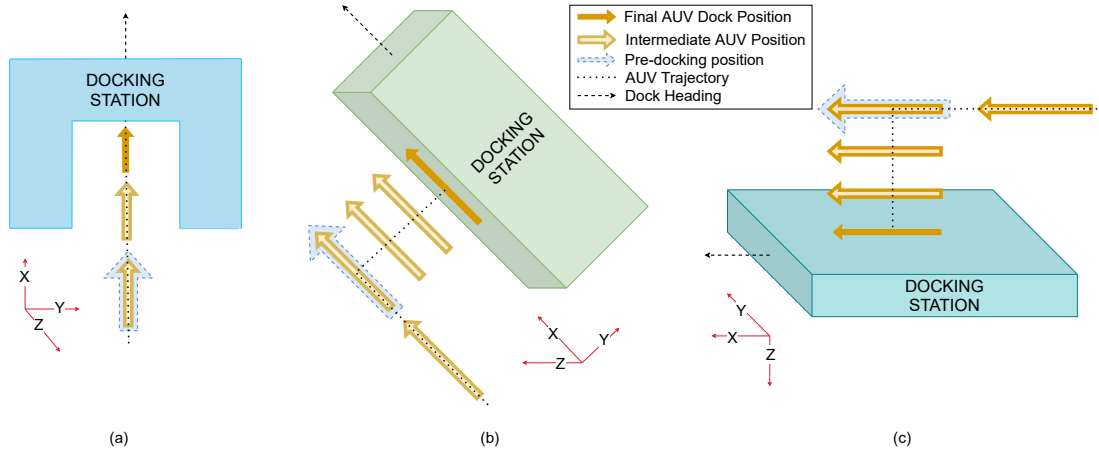


Figure 4.19 Docking approach from various sides of the docking station: (a) docking from the back (e.g., to dock on a stationary docking station situated at mid water column, such as submarines or to surface vessels), (b) docking from one side (e.g., to dock to the side of a surface vessel or to submarines), and (c) docking from the top (e.g., to dock to a towed underwater docking station).

a lot of drag and energy consumption. The dock location is assumed to be a point (dock position) in free space with an associated orientation (dock orientation) that the vehicle needs to reach with its heading aligned to the dock orientation. To aid the terminal docking maneuver, a pre-docking position is defined near the required dock position, at a safe distance from the docking station. A pre-docking position is essentially a point in free space with an associated orientation (pre-docking orientation) that the vehicle needs to reach (in position and orientation) prior to the execution of the docking maneuver. This pre-docking position can be used to attain different dock positions as depicted in Figure 4.19 where both non-holonomic and holonomic approaches can be attained by selecting the appropriate docking configuration as depicted in Figure 4.20. Once the vehicle matches its position and orientation to the pre-docking position and pre-docking orientation with the help of *Field Velocity* Task and *Velocity Alignment* Task discussed previously, it gradually surges or translates to the final dock position on the docking station with the help of *Terminal Dock* and *Dock Attitude* Task. Thus in order to perform this holonomic short range parking, it is possible by combining the *Terminal Dock* and *Dock Attitude* Task.

Considering the required behaviours of our vehicle, actions $\mathcal{A}_p$, $\mathcal{A}_{sa}$, $\mathcal{A}_{sd}$, $\mathcal{A}_a$ and $\mathcal{A}_d$ are defined and used later for the executing missions as explained in the following sections. The name of the action corresponds to the action defining objective that provides the main functionality. This main functionality, however, does not necessarily need to be the highest

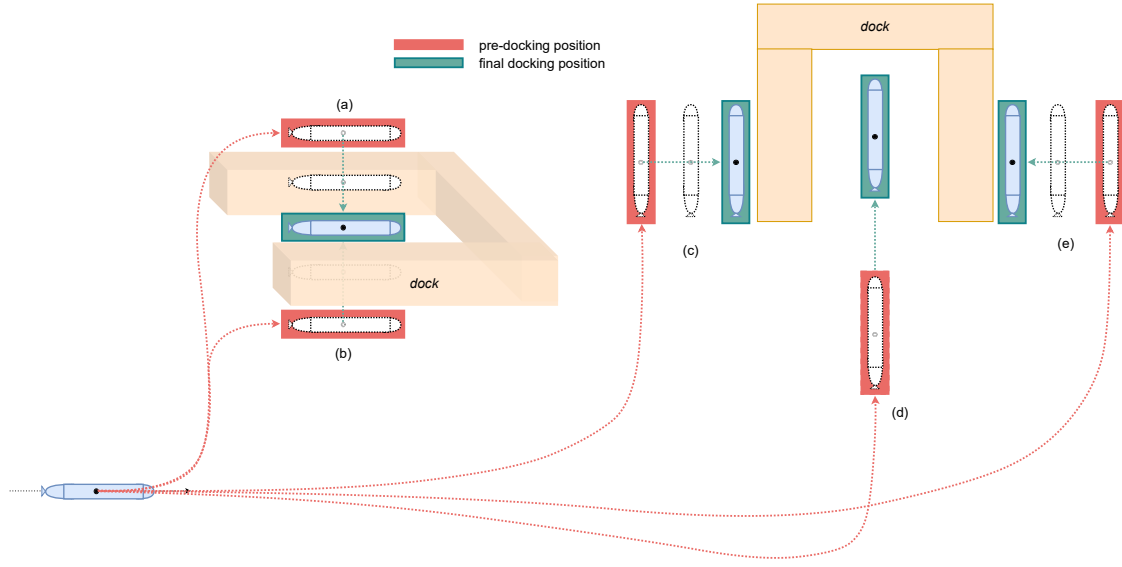


Figure 4.20 Docking configurations for holonomic and non-holonomic approaches: (a) top, (b) bottom, (c) left, (d) rear and (e) right of docking station.

priority task in that particular action. On the contrary, it is usually at a low priority, below the tasks which are considered to be related to the safety of the system. Table 4.2 provides the list of the tasks and their priority positioning within the control actions defined and used in this research.

The parking action $\mathcal{A}_p$ is composed of the two core tasks *Field Velocity Task* and *Velocity Alignment Task* that perform the parking maneuver. Further more, while doing the long range parking maneuver, the vehicle will need to avoid obstacles, avoid high pitch angles, stay away from hitting the seafloor and also avoid crashing into it while travelling at high speeds close to the seafloor. To structure these behaviours, *Obstacle Avoidance Task*, *Horizontal Alignment Task*, *Minimum Altitude Task*, and *Seafloor Alignment Task* are included. The task hierarchy used in $\mathcal{A}_p$ is,

$$\mathcal{A}_p : \mathcal{T}_1 \prec \mathcal{T}_2 \prec \mathcal{T}_3 \prec \mathcal{T}_4 \prec \mathcal{T}_7 \prec \mathcal{T}_8 \prec \mathcal{T}_{10}$$

Sometimes, while executing a survey at a constant altitude, the vehicle needs to follow a certain path by avoiding lateral translations, avoiding obstacles along its path, and by preventing the vehicle from crashing into the seafloor. To structure this behaviour, the constant altitude survey action $\mathcal{A}_{sa}$ incorporates the *Vehicle Altitude Task* along with additional tasks *Minimum Altitude Task*, *Obstacle Avoidance Task*, *Seafloor Alignment Task*, *Follow 2D Path*

Task	Task Type	$\mathcal{A}_p$	$\mathcal{A}_{sa}$	$\mathcal{A}_{sd}$	$\mathcal{A}_a$	$\mathcal{A}_d$
$\mathcal{T}_1$: Vehicle Non Actuation	E	1	1	1	1	1
$\mathcal{T}_2$: Minimum Altitude	I	2	2	2		
$\mathcal{T}_3$: Obstacle Avoidance	I	3	3	3		
$\mathcal{T}_4$: Horizontal Alignment	I	4				
$\mathcal{T}_5$: Vehicle Altitude	E		4			
$\mathcal{T}_6$: Vehicle Depth	E			4		
$\mathcal{T}_7$: Seafloor Alignment	I	5	5		2	
$\mathcal{T}_8$: Field Velocity	E	6				
$\mathcal{T}_9$: Follow 2D Path	E		6	5		
$\mathcal{T}_{10}$: Velocity Alignment	E	7	7	6		
$\mathcal{T}_{11}$: Hold Position	E				3	
$\mathcal{T}_{12}$: Survey Leg Alignment	E				4	
$\mathcal{T}_{13}$: Terminal Dock	E					2
$\mathcal{T}_{14}$: Dock Attitude	E					3

Table 4.2 List of tasks in actions that encompass the various behaviours of the vehicle. In task type, I refers to an inequality objective and E refers to an equality objective. The numbers denote the hierarchical priority of each task in their respective action where lower values represent a higher priority for that task in the action.

Task, and *Velocity Alignment* Task. The task hierarchy used in $\mathcal{A}_{sa}$ is,

$$\mathcal{A}_{sa} : \mathcal{T}_1 \prec \mathcal{T}_2 \prec \mathcal{T}_3 \prec \mathcal{T}_5 \prec \mathcal{T}_7 \prec \mathcal{T}_9 \prec \mathcal{T}_{10}$$

The constant depth survey action $\mathcal{A}_{sd}$ incorporates the *Vehicle Depth* Task along with additional tasks *Follow 2D Path* Task, *Obstacle Avoidance* Task, *Minimum Altitude* Task, and *Velocity Alignment* Task. This allows the vehicle to execute surveys at constant depths by following a specific path, avoiding obstacles and lateral translations and by preventing the vehicle from crashing into the seafloor. The task hierarchy used in $\mathcal{A}_{sd}$ is,

$$\mathcal{A}_{sd} : \mathcal{T}_1 \prec \mathcal{T}_2 \prec \mathcal{T}_3 \prec \mathcal{T}_6 \prec \mathcal{T}_9 \prec \mathcal{T}_{10}$$

The aligning action $\mathcal{A}_a$ incorporates the *Survey Leg Alignment* Task along with additional tasks *Seafloor Alignment* Task, and *Hold Position* Task. This allows the vehicle to align to the required orientation without changing its current position. The task hierarchy used in $\mathcal{A}_a$

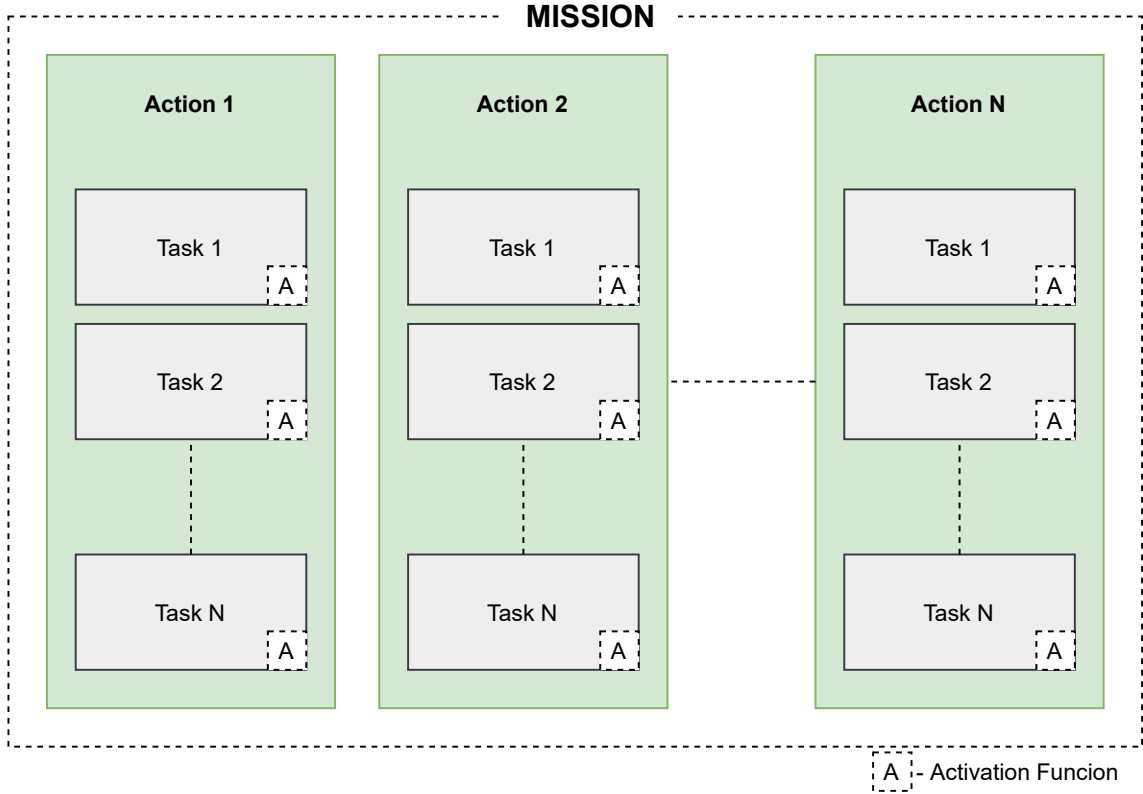


Figure 4.21 Mission, Actions and Tasks: A general overview of the relationship.

is,

$$\mathcal{A}_a : \mathcal{T}_1 \prec \mathcal{T}_7 \prec \mathcal{T}_{11} \prec \mathcal{T}_{12}$$

The docking action $\mathcal{A}_d$ incorporates the two core tasks *Terminal Dock* Task and *Dock Attitude* Task to perform the final docking maneuver. The task hierarchy used in $\mathcal{A}_d$ is,

$$\mathcal{A}_d : \mathcal{T}_1 \prec \mathcal{T}_{13} \prec \mathcal{T}_{14}$$

The priority level in task hierarchy ensures that the higher priority tasks are given extreme importance than the tasks having a lower priority.

4.5 Mission modelling

The control system is built by designing the actions. These actions are then sequenced by the mission manager, allowing us to execute the missions that are required, along with the all

the capabilities that were discussed above. A general overview of the relationship between the mission, its actions and the associated tasks are depicted in Figure 4.21. There are also improvements over the previous state of the art. Compared to [71], velocity field control technique [80, 81] is employed that produces a curvilinear trajectory for approaching in the desired orientation well in advance. To avoid collisions while approaching or performing tight maneuvers close to the docking station, a concept similar to the virtual obstacles proposed in [72] is used. In comparison to the simple approach discussed in [73], a case in which the docking station is close to the seafloor where the terrain may not be smooth is also considered. The final homing and docking maneuver is executed by following the seafloor, also maintaining a safe altitude. The fully lateral translation maneuver used in [74] is tackled by aligning the vehicle heading with the system velocity for reducing drag and high energy requirements during the homing and approach stages. The advantages of these contributions are that all these challenging conditions along with mission execution, homing and parking are integrated into the same task priority framework, improving over the state of the art. The implemented approach also allows docking from any initial condition of the vehicle to a stationary or moving docking station travelling in free space, close to the seafloor or on the surface. Additionally, pitch limitation is also implemented to avoid the vehicle from reaching pitches above its permissible limits. Let us consider some examples of missions that the vehicle should carry out where these behaviours are appreciated.

4.5.1 Constant altitude survey mission for pipeline inspection

Consider an example where a AUV is deployed from a submarine travelling in mid-water column for underwater pipeline inspections. The AUV needs to reach near the seafloor, inspect for underwater pipeline, cable installations etc, collect the data and dock to a docking station situated on the seafloor for data exchange and recharging. The docking station situated at the seafloor can be stationary or moving. For such inspection surveys, the vehicle needs to travel at a certain altitude above to the seafloor gathering data with sensors such as vision sensors. A constant altitude survey mission is performed in such scenarios. The steps for performing the Constant depth survey mission are as follows as depicted in Figure 4.22.

1. Deploy from a submarine situated at mid-water column
2. Go to the mission starting point
3. Start a constant altitude survey
4. Perform survey by following constant altitude

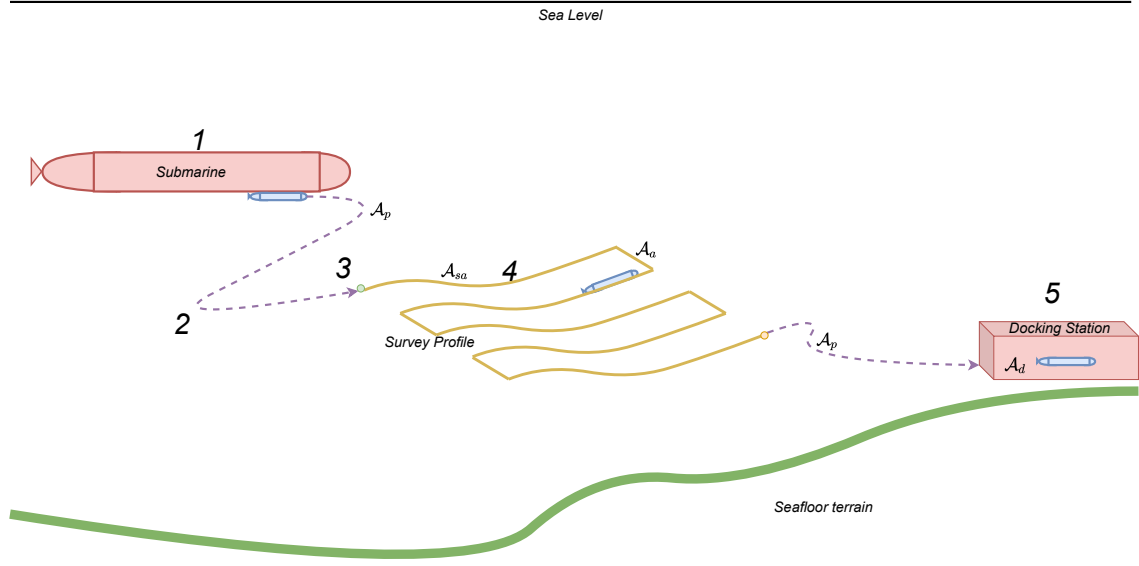


Figure 4.22 Constant altitude survey mission plan.

5. Finish survey and go back to the docking station situated close to the seafloor

Now, various actions to execute this survey mission is combined and sequenced. Once the vehicle is deployed from the submarine situated at mid-water column, the parking action is activated $\mathcal{A}_p$ to approach the mission starting point by avoiding lateral translation as much as possible. Along the way, thanks to the incorporated behaviour inside the action, the vehicle will be able to maintain a minimum altitude, avoid obstacles and prevent high pitch angles while approaching. Once the vehicle reaches the mission starting point, the constant depth altitude mission is initiated by constant altitude survey action $\mathcal{A}_{sa}$ which allows the vehicle to follow a specific path maintaining a constant altitude, but also allowing behaviours such as moving parallel to seafloor if the vehicle is travelling at a lower altitude, and also avoiding obstacles on the way. Along the survey, the $\mathcal{A}_a$ action allows the vehicle to orient to the next leg of the survey. Once the survey is finished, the parking action $\mathcal{A}_p$ is initiated to approach close to the moving docking station. Once the vehicle attains the pre-docking position, it finally executes the docking action $\mathcal{A}_d$ to carry out the terminal docking procedure. Throughout the mission, *Minimum Altitude Task* and *Seafloor Alignment Task* helps in preventing the vehicle from crashing into the seafloor. The *Obstacle Avoidance* ensures that the vehicle will keep out of a collision course from the known obstacles along its path. While doing the parking maneuver, the *Horizontal Alignment* makes sure the vehicle does not reach an unstable configuration due to high pitch angles.

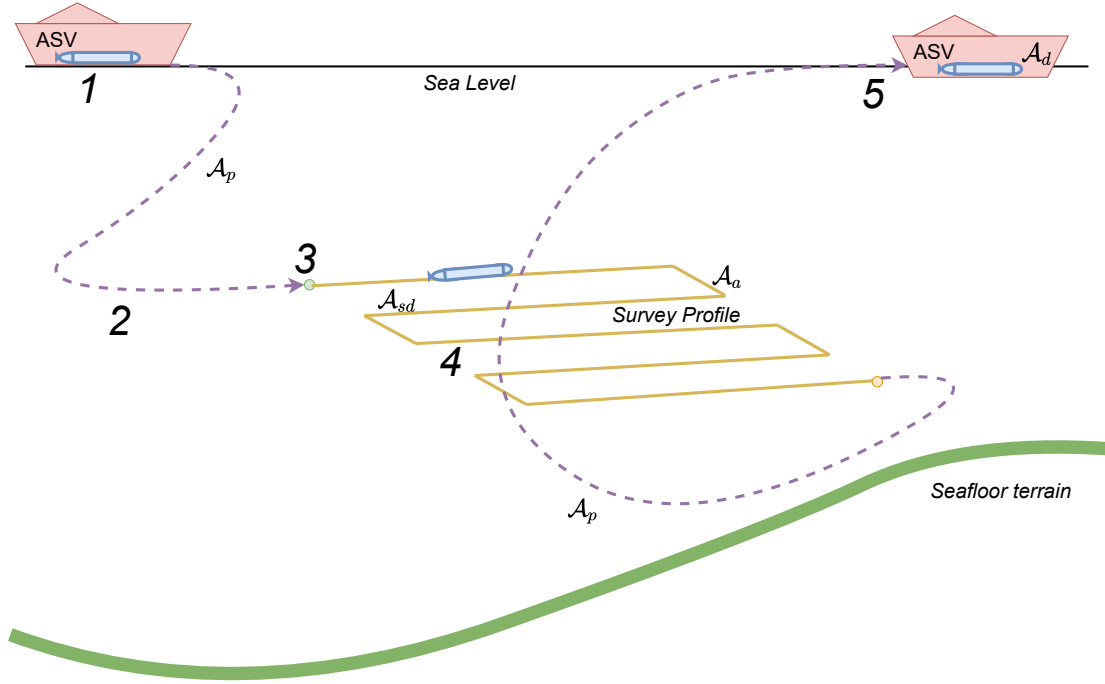


Figure 4.23 Constant depth survey mission plan.

Thus the sequence of actions to carry out this mission are as follows,

$$deploy \rightarrow \mathcal{A}_p \rightarrow \mathcal{A}_{sd} \rightarrow \mathcal{A}_a \rightarrow \mathcal{A}_p \rightarrow \mathcal{A}_d \rightarrow dock$$

4.5.2 Constant depth survey mission for bathymetry

Consider an example where the bathymetry of an area in the sea needs to be carried out. Here an AUV is deployed normally from a surface vessel such as ASV. After completing the mission, it returns to the ASV for docking and data transfer. The surface vehicle can be stationary or moving. For such bathymetry survey missions, the vehicle needs to travel at a certain depth gathering data with sensors such as SONAR or MBES (Multi Beam Echo Sounder). A constant depth survey mission is performed in such scenarios. Let us look further into the steps for performing the Constant depth survey mission as depicted in Figure 4.23.

1. Deploy from a surface vessel
2. Go to the mission starting point
3. Start a constant depth survey

4. Perform survey by following constant depth
5. Finish survey and go back to the surface vehicle

To execute this survey mission, various actions need to be combined and sequenced. Once the vehicle is deployed from the surface vessel, the parking action $\mathcal{A}_p$ is selected to approach the mission starting point by avoiding lateral/vertical translation as much as possible. Along the way, thanks to the incorporated behaviour inside the action, the vehicle will be able to maintain a minimum altitude, avoid obstacles and prevent high pitch angles while approaching. Once the vehicle reaches the mission starting point, the constant depth survey mission is initiated by constant depth survey action $\mathcal{A}_{sd}$ which allows the vehicle to follow a specific path, but also allowing behaviours such as maintain a minimum altitude above the seafloor, and avoiding obstacles. Along the survey, the $\mathcal{A}_a$ action allows the vehicle to orient to the next leg of the survey. Once the survey is finished, the parking action $\mathcal{A}_p$ is initiated to approach close to the moving docking station. Once the vehicle attains the pre-docking position, it finally executes the docking action $\mathcal{A}_d$ to carry out the terminal docking procedure. The *Minimum Altitude* task having the highest priority allows the vehicle to maintain a constant altitude above the sea floor while the *Obstacle Avoidance* ensures that the vehicle will keep out of a collision course from the known obstacles along its path throughout the mission. While doing the parking maneuver, the *Horizontal Alignment* makes sure the vehicle does not reach an unstable configuration due to high pitch angles.

Thus the sequence of actions to carry out this mission are as follows,

$$deploy \rightarrow \mathcal{A}_p \rightarrow \mathcal{A}_{sd} \rightarrow \mathcal{A}_a \rightarrow \mathcal{A}_p \rightarrow \mathcal{A}_d \rightarrow dock$$

4.5.3 Resident AUV survey mission for oil rig inspection

Resident AUVs are often deployed and retrieved to docking stations situated close to the seafloor. Consider an example where such a resident AUV needs to be deployed for inspecting an oil rig installation. For such oil rig inspection missions, the vehicle needs to travel at a certain depth gathering data on the oil rig structures. A constant depth survey mission is performed as a representative of such scenarios, though the mission profile would vary depending on the type of inspection required for the oil rig installation. The docking station for such purposes are normally stationary ones. Let us look further into the steps for performing the Constant depth survey mission for a resident AUV.

1. Deploy from a docking station situated close to the seafloor

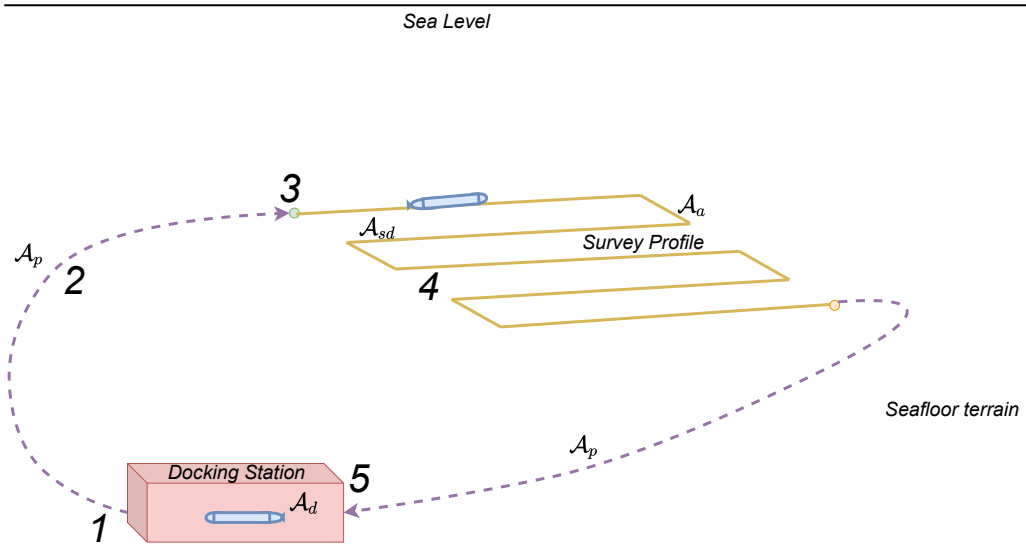


Figure 4.24 Resident AUV survey mission plan.

2. Go to the mission starting point
3. Start a constant depth survey
4. Perform survey by following constant depth
5. Finish survey and go back to the docking station situated close to the seafloor

To execute this survey mission, various actions are combined and sequenced. Once the vehicle is deployed from the docking station situated close to the seafloor, the parking action $\mathcal{A}_p$ is selected to approach the mission starting point by avoiding lateral/vertical translation as much as possible. Along the way, thanks to the incorporated behaviour inside the action, the vehicle will be able to maintain a minimum altitude, avoid obstacles and prevent high pitch angles while approaching. Once the vehicle reaches the mission starting point, the constant depth survey mission is initiated by constant depth survey action $\mathcal{A}_{sd}$ which allows the vehicle to follow a specific path, but also allowing behaviours such as maintain a minimum altitude above the seafloor, and avoiding obstacles. Along the survey, the $\mathcal{A}_a$ action allows the vehicle to orient to the next leg of the survey. Once the survey is finished, the parking action $\mathcal{A}_p$ is initiated to approach the stationary docking station. Once the vehicle attains the pre-docking position, it finally executes the docking action $\mathcal{A}_d$ to carry out the terminal docking procedure. The plan of this survey mission is depicted in Figure 4.24. The *Minimum Altitude* task having the highest priority allows the vehicle to maintain a constant altitude above the sea floor while the *Obstacle Avoidance* ensures that the vehicle will keep out of

a collision course from the known obstacles along its path throughout the mission. While doing the parking maneuver, the *Horizontal Alignment* makes sure the vehicle does not reach an unstable configuration due to high pitch angles.

Thus the sequence of actions to carry out this mission are as follows,

$$deploy \rightarrow \mathcal{A}_p \rightarrow \mathcal{A}_{sd} \rightarrow \mathcal{A}_a \rightarrow \mathcal{A}_p \rightarrow \mathcal{A}_d \rightarrow dock$$

Chapter 5

Application to the X300 AUV: Results and Discussion

This section presents different simulations, which have been carried out to highlight the advantages in using the unified control framework based on task-priority. In particular, the capability in performing parking and homing maneuvers using the velocity field control technique, the docking maneuver in various challenging scenarios and the mission capability are studied. In the docking procedure, the guidance system should be capable of handling both the homing and docking stages. It is responsible for generating a set of feasible velocity commands for the control system in response to the situational awareness (of the operating environment and the destination target provided by the navigation system), the mission objectives, and the vehicle dynamics. The control system produces corrective control forces and moments for the vehicle with respect to the vehicle states provided by the navigation system and reference velocities generated by the guidance system to successfully complete the docking operation. The challenge is that the docking station is not necessarily moored to the seafloor but could be towed by a surface vehicle and therefore floating underwater or near the sea surface. Moreover, the docking station can be stationary or mobile. The rendezvous needs to be accomplished to enable docking of AUV to the docking station. The improvements over the previous work are also incorporated in each test and the same will be analyzed further in this section. This chapter also gives an insight on the contributions **C4** and **C5**.

This research attempted to tackle different scenarios, which include

- docking to a stationary or moving docking station
- docking to docking station with various docking approaches

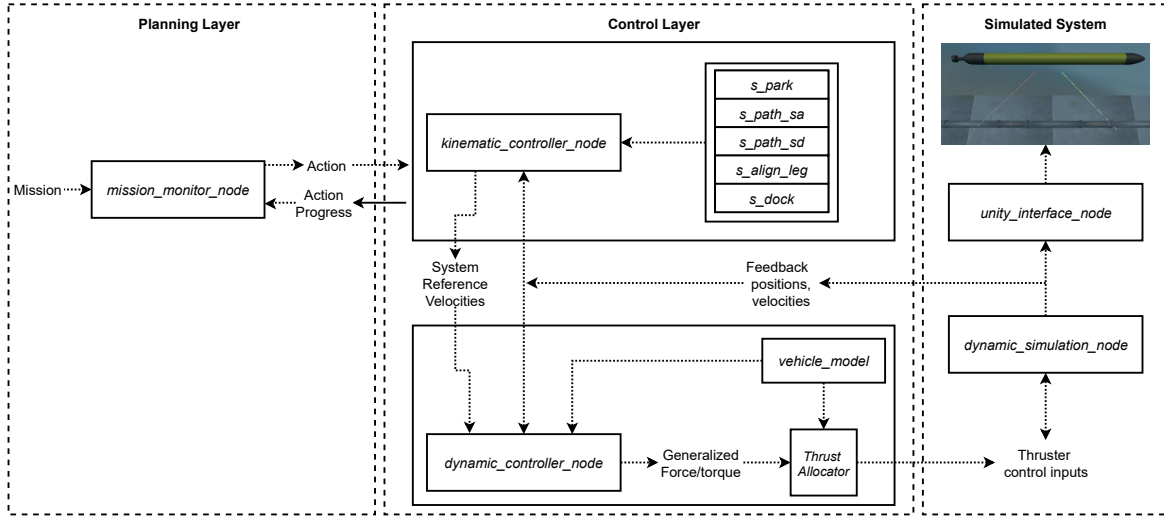


Figure 5.1 Software Program Structure.

- docking to docking station under partially enclosed or constrained situations
- deployment from or docking to a docking station near to the sea surface
- deployment from or docking to a docking station at mid-water column
- deployment from or docking to a docking station near to the sea floor
- execution of mission in the presence of known obstacles
- execution of mission near to seafloor/structures

5.1 Simulation Setup

The software program is written in C++ as it is the programming language that can be used to interface with the real vehicle. The interface to the Simulated System is the same as that of the actual vehicle, allowing for software-in-the-loop simulations of our KCL implementation as detailed in Chapter 4. Concerning the Planning Layer, a very basic state machine is implemented to handle the sequence of actions. The software program structure is given in Figure 5.1. In this work, focus is given on the control and structuring the behaviour of the vehicle. The work does not deal with the the problems of navigation or estimating the position of the docking station and are thus out of scope. Hence, the availability of sensor modules are assumed and that all simulations are done with perfect feedback.

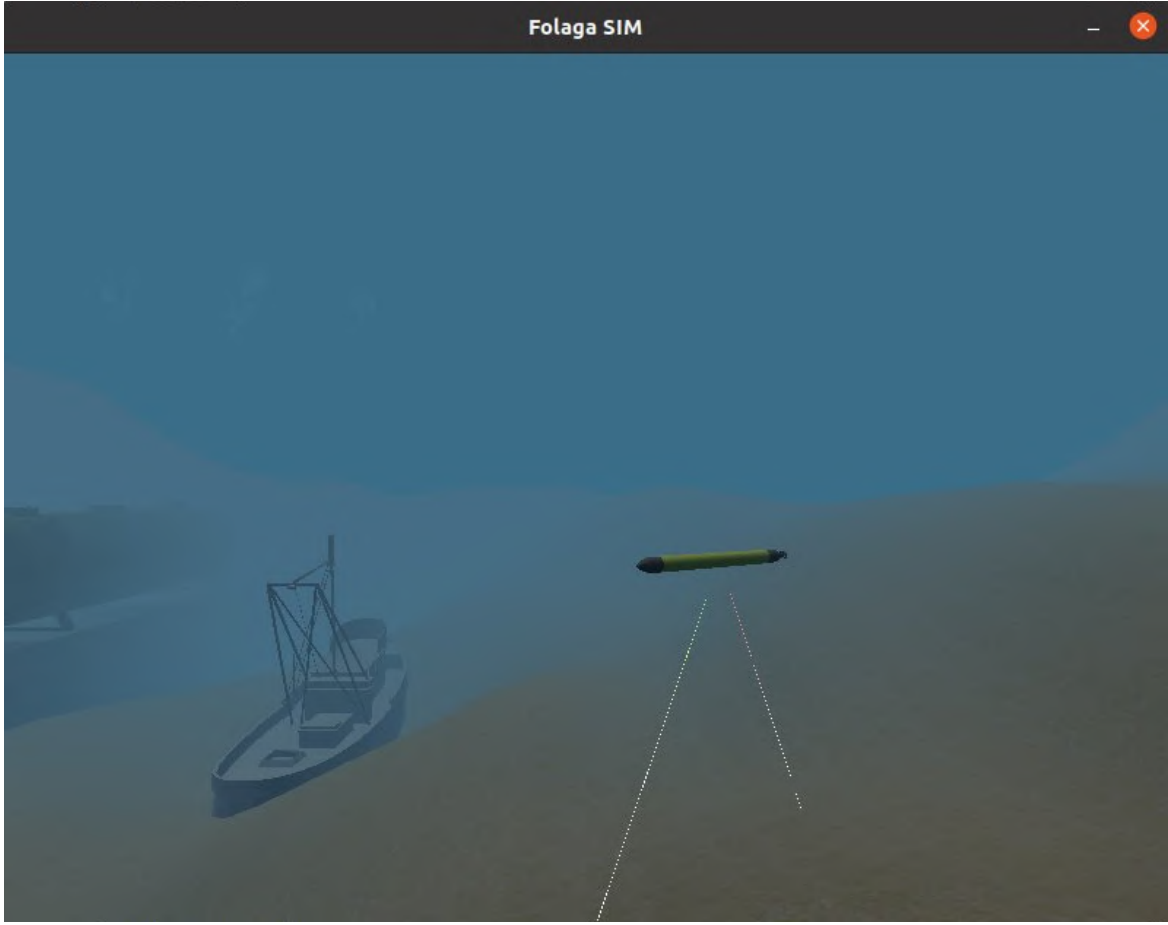


Figure 5.2 Unity GUI.

The *mission_monitor_mode* in the Planning Layer is the mission manager that oversees progress of each actions and executes the next sequence of actions to perform and complete the mission. Once the mission manager initiates an action, the message is passed to the *kinematic_controller_node* in the Control Layer. The *kinematic_controller_node* implements the unified task priority control framework and also takes advantage of the various states that are implemented through the finite state machine. The *s_park* state implements the parking action $\mathcal{A}_p$, *s_path_sa* state implements the constant altitude survey action $\mathcal{A}_{sa}$, *s_path_sd* state implements the constant depth survey action $\mathcal{A}_{sd}$, *s_align_leg* state implements the survey leg aligning action $\mathcal{A}_a$, and *s_dock* state implements the docking action $\mathcal{A}_d$. From these states, the action defining tasks are executed to produce the system reference velocities. The *dynamic_controller_node* tracks these system reference velocities using a PI controller to produce the desired wrench. The force that each thruster must exert, which is an input to the Simulated System, is found using the thruster allocation mechanism from the *vehicle_model*.

```

Time (s): 73.600          Current State: PARK

Initial Pose NED (m, deg): = -5.000 -10.000 26.000 0.000 0.000 40.000      IsDockMoving: 1      IsVirtualObs: 1
Desired Pose NED (m, deg): = -10.403 9.597 28.000 0.000 0.000 225.000      IsHoming: 1          IsParking: 1
Dock Pose NED (m, deg): = -10.407 9.593 30.000 0.000 0.000 225.000      IsDocking: 0         IsVirtualTarget: 1

Kin Position NED (m, deg)      Linear = -9.756 10.240 27.999      Angular = 359.984 1.222 224.705      Dyn Position NED (m, deg)      Linear = -9.775 10.221 27.999      Angular = 359.990 1.173 224.744
Kin Velocity BF (m/s, rad/s)   Linear = 0.333 -0.001 0.004      Angular = 0.000 -0.010 -0.003      Dyn Velocity BF (m/s, rad/s)   Linear = 0.333 -0.002 0.004      Angular = 0.001 -0.010 0.007

Dynamic Controller: PID
Dyn Thrust (%): = 3.991 -3.991 -0.840 0.840 6.842 -6.842 0.784 -0.784 2.214
PID values: = 2.214 5.416 -0.028 0.000 0.812 -1.426

AUV Altitude NED (m): 2.287      Survey Alt NED (m): 3.000
IsMission: 1      IsCnstDpth: 0      IsParking: 1      DoLOS: 1

uv PL park vel
IntAct= 1.000 1.000 1.000      TotAct= 1.000 1.000 1.000
x_dot= 0.332 -0.001 0.008      dy= 0.000 0.000 0.000 0.332 -0.001 0.032

uv PL align to velocity
IntAct= 1.000 1.000 1.000      TotAct= 1.000 1.000 1.000
x_dot= 0.000 0.012 0.003      dy= 0.000 -0.010 -0.003 0.000 0.000 0.000

uv PL align to horiz
IntAct= 0.000      TotAct= 0.000
x_dot= 0.294      dy= 0.000 0.000 0.000 0.000 0.000

uv PL terminal dock
IntAct= 1.000 1.000 1.000      TotAct= 0.000 0.000 0.000
x_dot= -0.248 -0.001 -0.500      dy= 0.000 0.000 0.000 0.000 0.000 0.000

uv PL dock attitude
IntAct= 1.000 1.000 1.000      TotAct= 0.000 0.000 0.000
x_dot= -0.000 0.011 -0.003      dy= 0.000 0.000 0.000 0.000 0.000 0.000

uv PL align to floor
IntAct= 1.000      TotAct= 0.282
x_dot= -0.003      dy= 0.000 -0.000 0.000 0.000 0.000 0.000

uv PL min altitude
IntAct= 0.282      TotAct= 0.282
x_dot= 0.357      dy= 0.000 0.000 0.000 0.001 0.000 -0.028

uv PL goto dis_lin
IntAct= 1.000 1.000      TotAct= 0.000 0.000
x_dot= -0.000 -0.023      dy= 0.000 0.000 0.000 0.000 0.000 0.000

uv PL align to surv leg
IntAct= 1.000 1.000 1.000      TotAct= 0.000 0.000 0.000
x_dot= 0.000 0.250 0.001      dy= 0.000 0.000 0.000 0.000 0.000 0.000

uv PL ctrl alt
IntAct= 1.000      TotAct= 0.000
x_dot= -0.159      dy= 0.000 0.000 0.000 0.000 0.000 0.000

uv PL obs avoidance
IntAct= 0.000      TotAct= 0.000
x_dot= -0.105      dy= 0.000 0.000 0.000 0.000 0.000 0.000

uv PL hold position
IntAct= 1.000 1.000 1.000      TotAct= 0.000 0.000 0.000
x_dot= 0.000 0.000 0.000      dy= 0.000 0.000 0.000 0.000 0.000 0.000

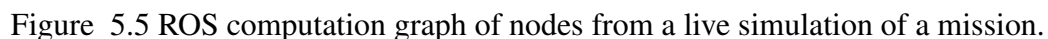
uv PL hold attitude
IntAct= 1.000 1.000 1.000      TotAct= 0.000 0.000 0.000
x_dot= 0.000 0.000 0.000      dy= 0.000 0.000 0.000 0.000 0.000 0.000

```

Figure 5.3 Live Visual Feedback Display.

It maps the desired wrench on the body to the forces that need to be generated by each of the thrusters. This thruster control input is fed to the *dynamic_simulation_node* which produces simulated positions and velocities which are fed back to the *kinematic_controller_node* to estimate the action progress. The simulated positions and velocities are further used by the *unity_interface_node* to simulate the vehicle in the Unity GUI Simulator as shown in Figure 5.2. Furthermore, various information regarding the ongoing simulation is presented in the form of live visual feedback display as shown in Figure 5.3. The mission progress is also displayed as shown in 5.4. A ROS computation graph of the nodes from a live simulation of a constant altitude survey mission is shown in Figure 5.5

The *vehicle_model* used in the simulation is the X300 AUV, which is a low-cost torpedo-shaped AUV that is capable of maneuvering with 5 degrees of freedom along the surge, sway, heave, pitch, and yaw directions. The X300 hydrodynamic model is based on the general Fossen model [32–34] for an underwater vehicle. The mass and drag parameters



of the experimental vehicle X300 have been identified prior to this research. Assuming the vehicle is symmetric in the $x-z$ plane and almost symmetric in the $x-y$ plane (presence of antennas and transducers) and $y-z$ plane (difference in nose and tail structures), the origin of vehicle frame coincides with the vehicle center of buoyancy. Various dynamic model parameter terms can thus be simplified. The vehicle is powered by 5 thrusters consisting of a

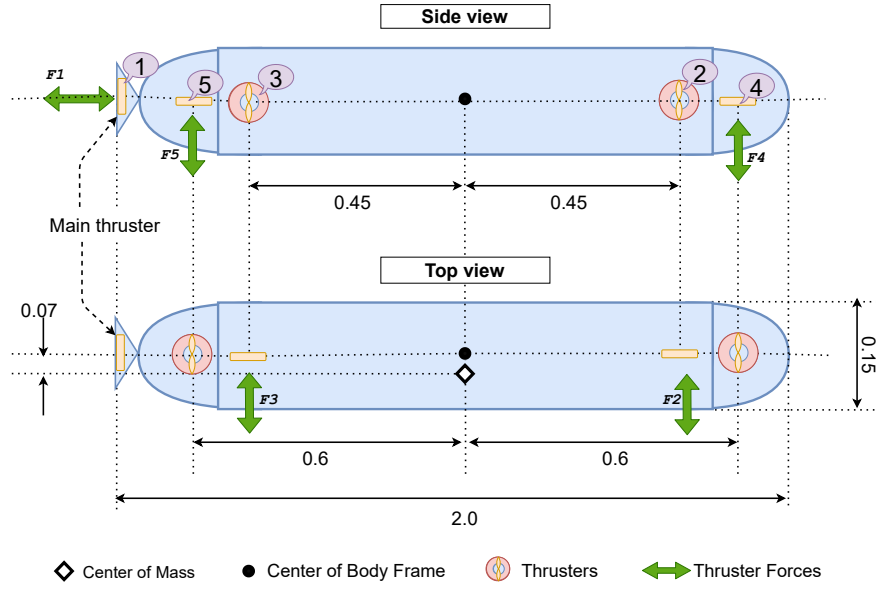


Figure 5.6 AUV: X300 model and allocation of thrusters.

main thruster at the aft, 2 lateral thrusters, and 2 vertical thrusters, and thus is not actuated in roll as depicted in Figure 5.6. The vehicle has its center of mass below its center of buoyancy, allowing the restoring forces to provide passive stability for the vehicle in roll, making roll negligible. It is also notice that the center of mass is below the line of the thrusters, thereby generating transient roll while translating. However, it should also be noted that the lateral and vertical translations are limited as much as possible.

Furthermore, the altitude sensors in the simulated vehicle are arranged in the $x-z$ plane of the vehicle as depicted in Figure 5.7 where they are fixed at 45° forward and aft w.r.t. the vehicle body frame $\langle b \rangle$. These two sensors allows an estimation of the slope of the seafloor by determining the seafloor vector. In the sensor frame $\langle s \rangle$, the rays from the two sensors make contact with seafloor at points ${}^sA$ and ${}^sB$, respectively. The floor vector $\mathbf{f}$ in $\langle b \rangle$ is given as ${}^b\mathbf{f} = {}^b\mathbf{R}_s({}^sA - {}^sB)$, where ${}^b\mathbf{R}_s$ denotes the rotation matrix from $\langle s \rangle$ to $\langle b \rangle$. The vehicle follows the seafloor terrain by aligning its body z axis in the direction of the orthogonal component $\mathbf{n}_{st}$, calculated as a normalized cross-product of the floor vector ${}^b\mathbf{f}$ and the body y axis $\mathbf{j}_b$. It is obvious that the roll could point the altitude sensors far away from the vertical projection of the vehicle on the surface, thereby estimating the slope in another point than the one just below the vehicle. The roll is assumed to be negligible as discussed in 5.1. The roll could be further limited by adding fins to make the vehicle more horizontal in terms of roll. For estimating the vehicle altitude, an absolute vertical line is considered, passing through the

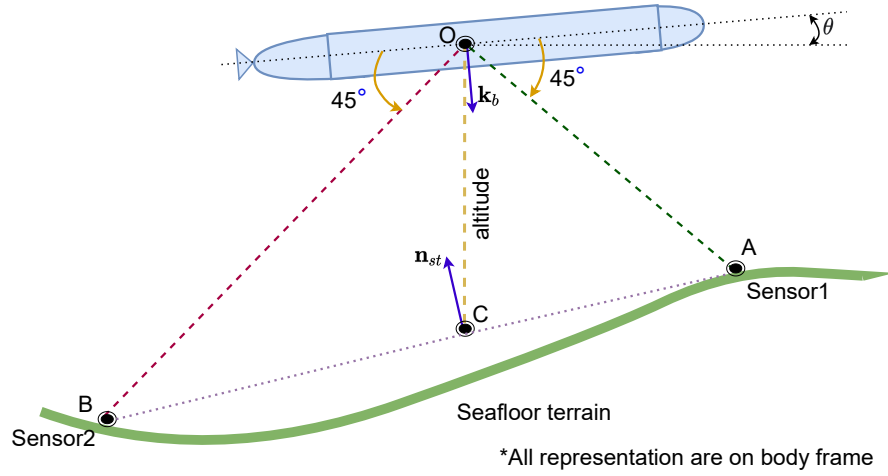


Figure 5.7 X300: Allocation of altitude sensors w.r.t. center of vehicle body frame and estimation of vehicle altitude as length of the vertical segment. The slanting rays emerging from the vehicle highlight the position of the altitude sensors w.r.t. the body of the vehicle.

System Parameter	Value	Model Parameter	Value
DOF	5	Type	Torpedo
Length (m)	2	G (m/s ²)	9.810
Width (m)	0.15	ρ (kg/m ³)	1023.6
Weight (kg)	31	CG (m)	[0.000, 0.000, 0.070]
Main Propulsion	Central thruster	CB (m)	[0.000, 0.000, 0.000]
Propellers	5	Added Mass	diag [-0.47, -22.7, -22.7, -0.1, -3.64, -3.64]
		Linear Drag	diag [1.079, 10.21, 10.21, 0.5, 1.061, 1.061]
		Quadratic Drag	diag [0.794, 102.1, 102.1, 20.48, 15.191, 15.191]
		Thruster distance (m)	[-1.0, 0.45, -0.45, 0.6, -0.6]

Table 5.1 X300: System Parameters and Dynamic Model Parameters. Thruster distance represents the longitudinal distance of each thruster from the center of body frame.

vehicle body center ${}^bO$ and touching the floor vector ${}^b\mathbf{f}$ at point ${}^bC$. The estimated altitude is the length of the vertical segment (${}^bC - {}^bO$).

Table 5.1 gives a general layout of the vehicle characteristics, including the system and dynamic model parameters.

The dynamics of the vehicle is assumed to be governed by a PI-based velocity control system given by

$$\mathbf{u} = \mathbf{K}_p \tilde{\mathbf{v}} + \mathbf{K}_i \int_{t_0}^t \tilde{\mathbf{v}} dt, \quad (5.1)$$

where $\tilde{\mathbf{v}}$ is the velocity tracking error (excluding the p angular velocity), and $\mathbf{K}_p$ and $\mathbf{K}_i$ are the proportional and integral gain matrices, respectively. The controller gains are tuned based on experimental step response obtained using the simulated vehicle dynamic

Manufacturer	Sensor	Type	Application	Rate (hz)	Link
Sonardyne	Mini Ranger	USBL	Homing	3	[82]
Advanced Navigation	Subsonus	USBL	Homing	10	[83]
Matrox	Meteor-II	Frame Grabber	Docking	15	[84]
Sonardyne	Sprint-NAV	INS + DVL + AHRS	Navigation	200	[85]
Impact Subsea	ISA500	Altimeter	Navigation	500	[86]

Table 5.2 Table showing details of Commercially available AUV sensors.

model. The controller is thus initialized with gains $\mathbf{K}_p = \text{diag}([750, 780, 780, 380, 100])$ and $\mathbf{K}_i = \text{diag}([2280, 2310, 2310, 2000, 90])$.

As mentioned earlier, for the simulation, the availability of sensors is assumed (e.g., a forward looking sonar) that are able to compute distances to obstacles when in close proximity with the vehicle for providing feedback to the obstacle avoidance tasks. Moreover, it is assumed that the acoustic communications (e.g., Ultra Short Base Line (USBL) unit with integrated acoustic modem) with the docking station for long range homing is present and visual markers on the docking station are recognizable with an on-board camera for terminal docking. It is further assumed that the vehicle can measure its pose and velocity using on-board sensors (e.g., Inertial Measurement Unit (IMU) and Doppler Velocity Log (DVL)). The docking station is assumed to measure its pose and send it to AUV using proper communication channels. In case of a moving docking station, its motion is constrained to a straight line motion with a constant heading, and to allow rendezvous, the docking station will be moving with a velocity lower than that of the maximum velocity of the vehicle. Details of the commercially available sensors are depicted in Table 5.2.

In order to achieve realistic simulations, obstacles are added into the simulation environment. For simplicity, the obstacles are modeled as spheres and the paper is not concerned regarding a fine modeling of the obstacles. As is commonly done, all the area around the docking place are enveloped into sphere obstacles for repulsing the vehicle from colliding into structural elements of the station. These spheres are also used to envelope no-fly zones to represent places or areas where the vehicle should not travel for other reasons than collision risks. For example, to dock to a submarine [72] or an ASV [74], a no fly zone can be defined by placing several spheres around the propeller area of the submarine or ASV so that the vehicle avoids entering into turbulent waters generated by the propulsion system of the submarine or ASV. These envelop obstacles defined around the docking station, can also guide the vehicle by pushing it further away from the docking station and around if required, towards the required approach position in a safe manner during docking or during

deployment. Clearly when defined w.r.t. the dock location, the envelop obstacle spheres will be attached to the dock position body coordinates and would move at the same rate as that of the dock.

Regarding the common tasks in every tests, the *Minimum Altitude* Task is always active at the highest priority for safety reasons. The *Obstacle Avoidance* Task is also active at all stages except the final maneuver in Terminal Docking, allowing complete docking to the docking station. The *Seafloor Alignment* Task is active below a threshold altitude of the vehicle and the *Horizontal Alignment* Task ensures that that pitch attitude of the vehicle always stay within its safety limits. Pre-docking position and pre-docking orientation are defined if necessary outside at a safe distance from the docking station.

5.2 Simulation Results

This section highlights the capability of the system in performing various maneuvers and behaviours as part of the completion of objectives under challenging conditions. The behaviours that the system displays are taken into account while tested in free space, partially enclosed or constrained spaces and while executing various missions. There is an advantage of docking from anywhere and at any initial conditions of the vehicle, and also the ability to avoid the obstacles along the path and to avoid crashing into the dock structure which will be discussed further in this section. Concerning tasks, the activation points and threshold ranges of various parameters used in the simulations are given in Table 5.3. In all the cases, the task starts getting activated in the transition region and gets fully activated on reaching the upper threshold limit for *Minimum Altitude*, *Obstacle Avoidance* and *Seafloor Alignment* Tasks and on reaching the lower threshold limit for *Horizontal Alignment* Task. Influences of sensor data sampling rates and ocean currents are also studied in this section.

5.2.1 Parking tests in Free Space

This section highlights the capability of the system in performing free space parking maneuver (also used for homing) using the velocity field control technique implemented in the unified control framework design as discussed in Section 4.4. In these tests, the vehicle performs normal parking. The goal orientation is represented by a green arrow.

Task	Parameter	Minimum Activation	Threshold Range	Maximum Activation
<i>Minimum Altitude</i>	vehicle altitude	$< 3m$	$1m$ to $3m$	$1m$
<i>Obstacle Avoidance</i>	obstacle distance	$< 1.7m$	$0.3m$ to $1.7m$	$0.3m$
<i>Seafloor Alignment</i>	vehicle altitude	$< 3m$	$1m$ to $3m$	$1m$
<i>Horizontal Alignment</i>	vehicle pitch	$< -35^\circ$ or $> +35^\circ$	$\pm 35^\circ$ to $\pm 45^\circ$	-45° or $+45^\circ$

Table 5.3 Activation points and Threshold ranges of various parameters of the tasks used in the simulations. The threshold range represent the transition region of the task.

Influence of Gain parameters

Various trajectories of the vehicle while the parking maneuver is performed in free space with different parameters are presented in Figure 5.8. Here the control parameters; linear gain λ , angular gain γ and modulation gain σ of the task are varied. Increasing the value of λ allows the vehicle to reach the goal faster, and increasing the value of γ allows the vehicle to achieve the desired orientation sooner. Varying σ allows the vehicle to reach the final configuration faster or slower, while following the same path. The result of the free space docking simulations very well portrays how the vehicle perform the parking maneuver for various parameters to obtain the desired target position and orientation irrespective of its initial starting conditions. Action $\mathcal{A}_p$ is used for this test.

Influence of Minimum Altitude and Obstacle Avoidance Task

The influence of *Minimum Altitude* Task and *Obstacle Avoidance* Task in the parking maneuver with Action $\mathcal{A}_p$ is observed in the following test. Figure 5.9 depicts the trajectory of the vehicle in free space under the influence these tasks. The vehicle starts behind a hill area and proceeds towards the a goal (which is set on the other side of the hill). A sphere obstacle (represented by red sphere) of radius $1m$ is present behind the hill at position $[14, 10.5, 27]$ in the path of the vehicle. As the vehicle proceeds towards its goal, it encounters the obstacle behind the hill and activates the *Obstacle Avoidance* Task to avoid a collision with the obstacle. Between time $t = 18s$ and $t = 23s$, the distance of the vehicle to the obstacle breached the allowed obstacle distance safety limit of $1.7m$ as defined in Table 5.3. It is also observed that the *Minimum Altitude* Task is also active during this period because the current

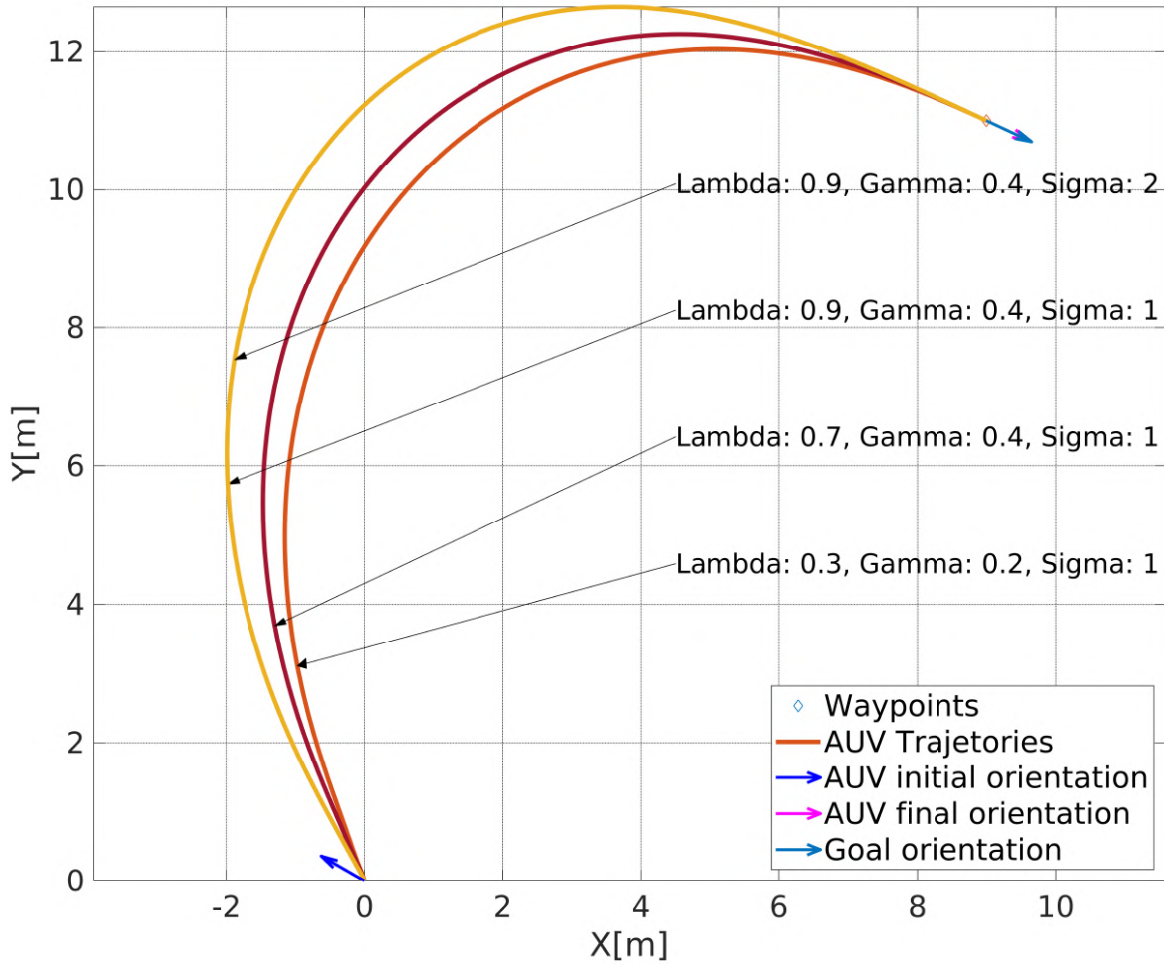


Figure 5.8 X300: Comparison between different maneuvers in free space parking by changing the linear gain, angular gain and modulation parameters.

altitude of the vehicle has breached and went below the allowed vehicle altitude safety limit of 3m. As the *Minimum Altitude* Task has higher priority over the *Obstacle Avoidance* Task (refer Table 4.2) the vehicle travels above the obstacle to avoid collision and also to maintain the minimum altitude safety limits. Thus the combination of the *Obstacle Avoidance* and *Minimum Altitude* tasks allowed the vehicle to safely navigate past the obstacle and the hill to reach the goal.

Influence of Seafloor Alignment Task

A test is done to observe the influence of *Seafloor Alignment* Task while doing the parking maneuver close to a seafloor. Figure 5.11 depicts the trajectory of the vehicle in free space under the influence of the *Seafloor Alignment* Task. The vehicle starts behind a hill area

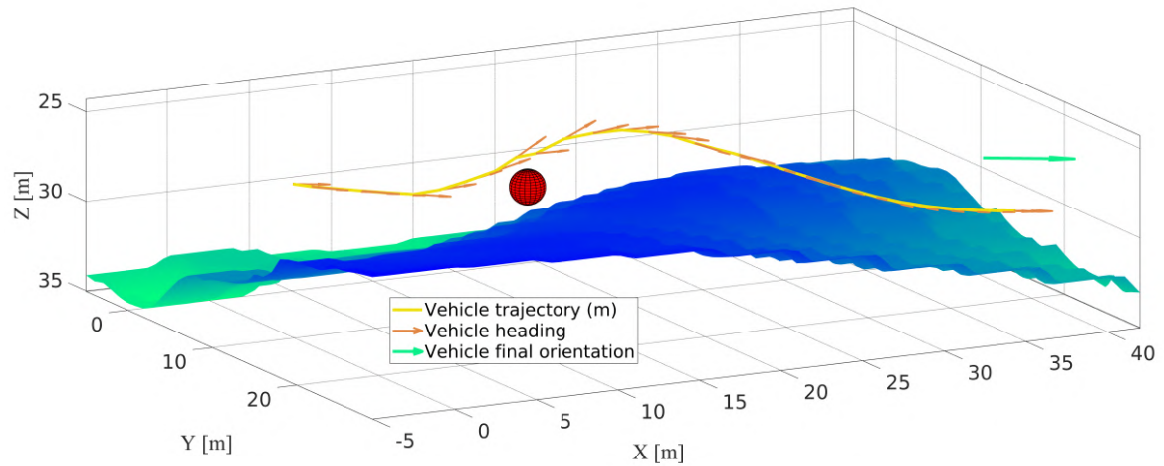


Figure 5.9 X300: Maneuver showing the vehicle performing Obstacle Avoidance Task for an obstacle (represented by red sphere) near a hill.

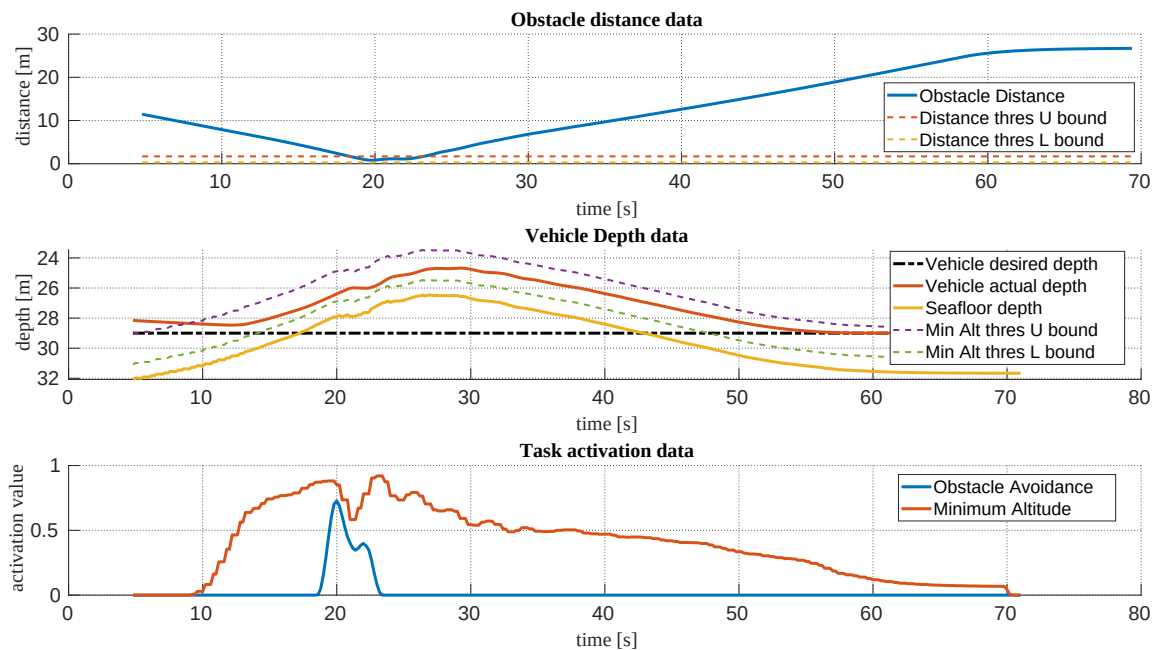


Figure 5.10 X300: Data showing the distance to object, vehicle depth and corresponding activation for Obstacle Avoidance and Minimum Altitude Tasks.

and proceeds towards a goal (which is set on the other side of the hill), close to the seafloor. As discussed in 4.3.5, *Seafloor Alignment* Task depends on an external variable, which is the current altitude of the vehicle. In this test, the *Seafloor Alignment* Task gets activated once the current altitude of the vehicle is below 3m. As the vehicle proceeds towards its goal, at altitudes below 3m, *Seafloor Alignment* Task is activated along with the *Minimum*

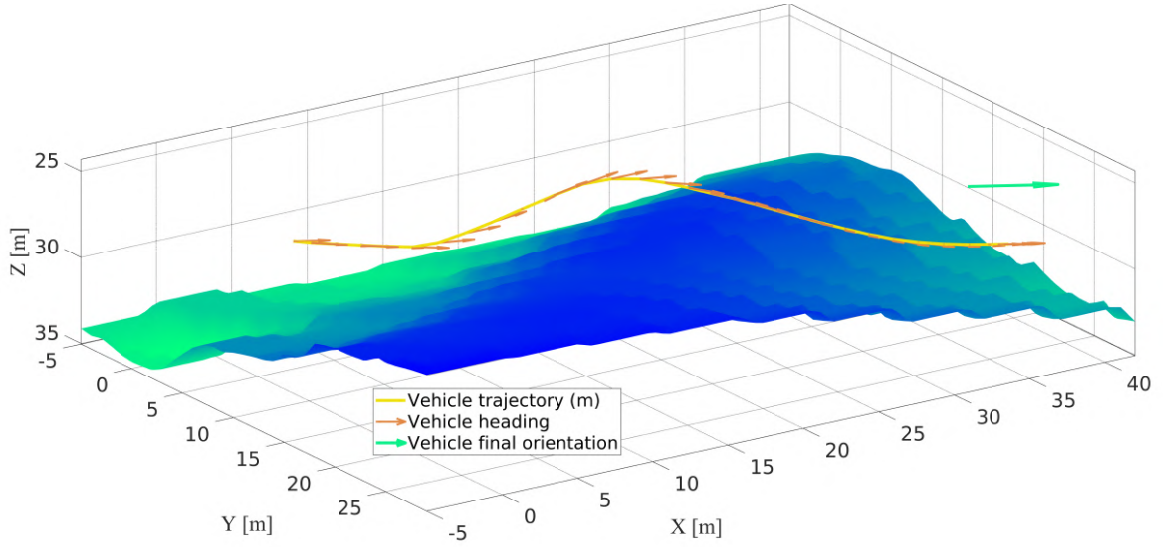


Figure 5.11 X300: Maneuver showing the vehicle performing Seafloor Alignment Task close to seafloor.

Altitude Task where both tasks work hand in hand to provide a safe approach around the hill and towards the goal near the seafloor. The *Seafloor Alignment* Task further ensures that the vehicle aligns to the slope of the seafloor in order to avoid crashing down while moving at higher speeds. Action $\mathcal{A}_p$ is used for this test.

Influence of Horizontal Alignment Task

The following test shows the influence of *Horizontal Alignment* Task in the parking maneuver with Action $\mathcal{A}_p$. Figure 5.13 depicts the trajectory of the vehicle in free space under the influence of the *Horizontal Alignment* Task. The vehicle starts from near to the sea surface and proceeds towards a goal set in the mid-water column. As the vehicle pitches down as it moves towards its goal, between time $t = 20s$ and $t = 42s$, the *Horizontal Alignment* Task gets activated. This is because the current pitch attitude of the vehicle breached the allowed vehicle pitch safety limit of $+/- 35^\circ$ as defined in Table 5.3. The *Horizontal Alignment* Task gets activated to ensure that the pitch attitude of the vehicle does not cross the maximum permissible limit of $+/- 45^\circ$.

Influence of Sensor Data Sampling Rates

The ability of the system to complete the mission for sensor data sampling rates is studied. In this test, a worst case scenario is considered where the vehicle sensors are sampled at a lower

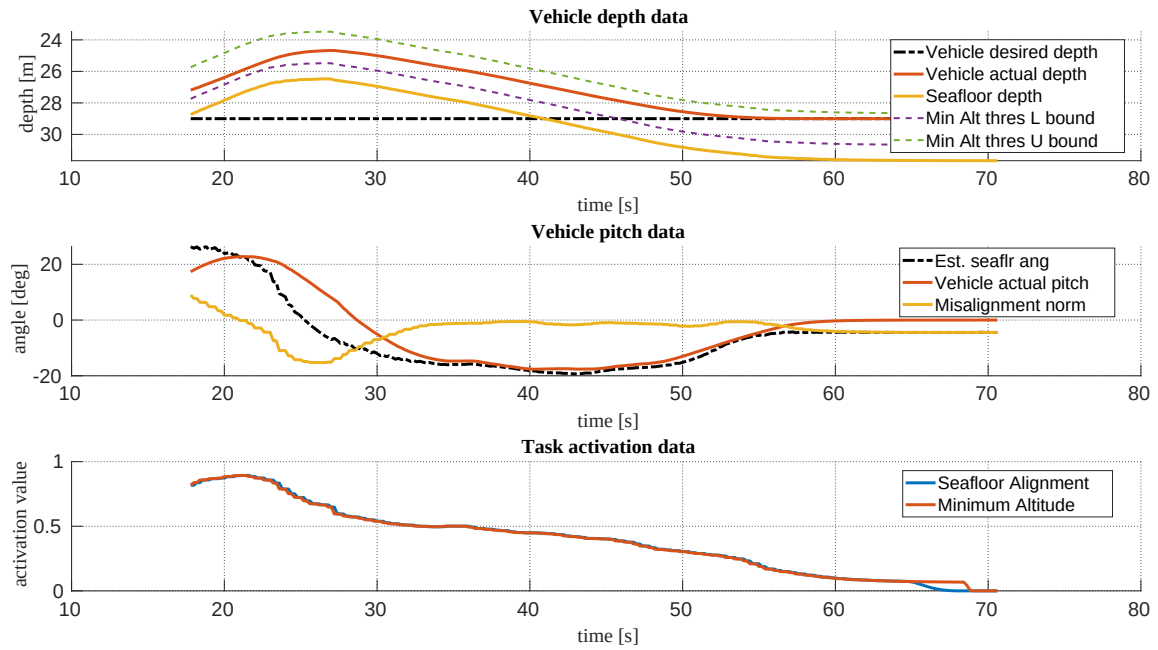


Figure 5.12 X300: Data showing the depth and pitch of the vehicle and corresponding activation for Seafloor Alignment and Minimum Altitude Tasks.

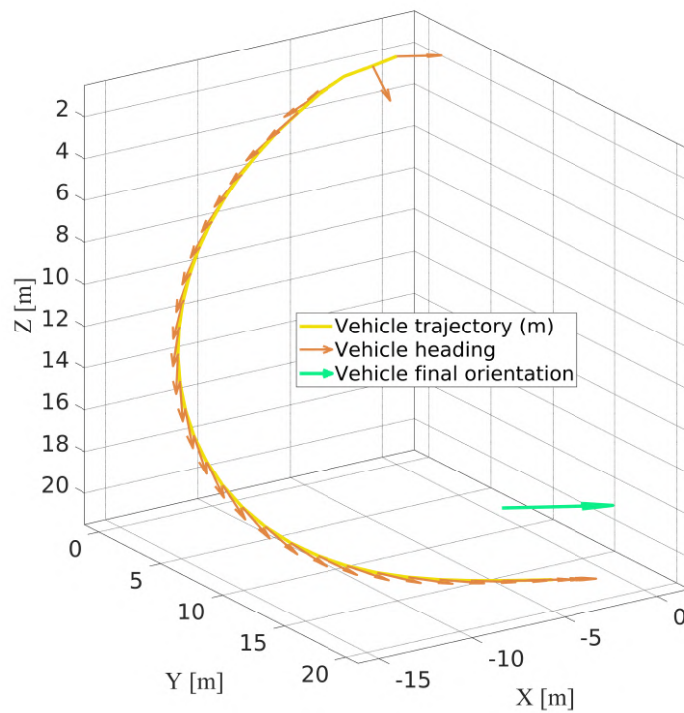


Figure 5.13 X300: Maneuver showing the vehicle performing Horizontal Alignment Task while doing a deep maneuver.

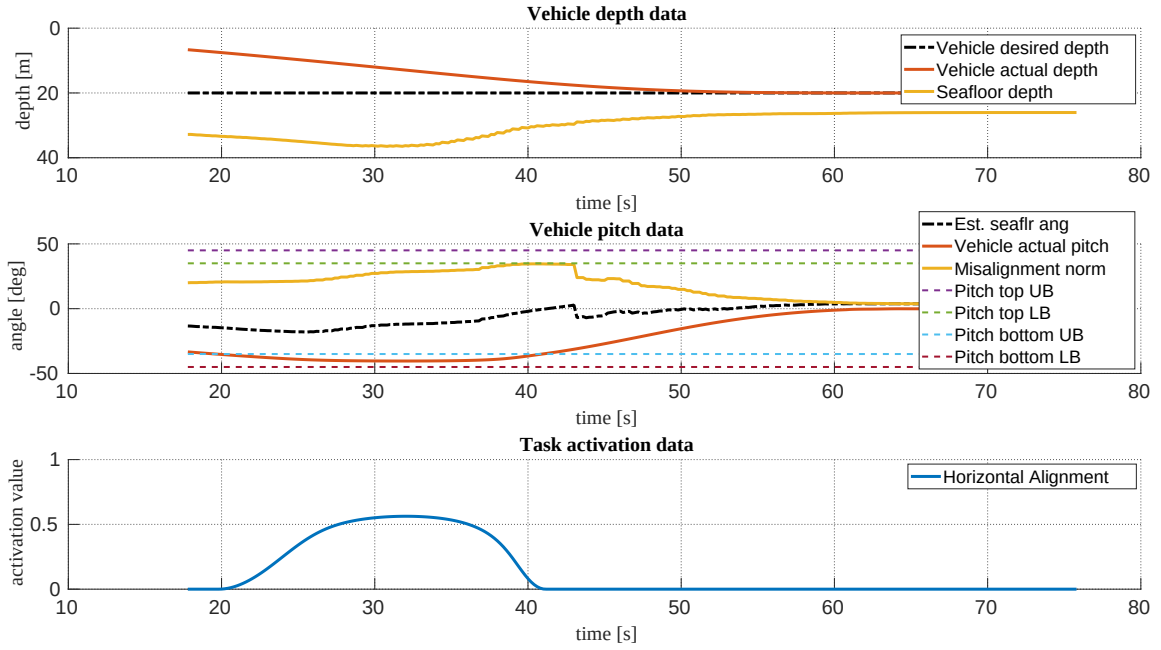


Figure 5.14 X300: Data showing the depth and pitch of the vehicle and corresponding Horizontal Alignment Task activation.

rate. The homing and docking phases of a mission are taken into account where sensors such as Navigation, USBL and optical sensors are relevant. The data sampling rates used in the test are as follows: Homing USBL sensor is sampled at 3Hz , Docking optical sensor is sampled at 5Hz and Navigation IMU sensors are sampled at 10Hz . These rates are well within the data sampling range available from sensors of commercial manufacturers (Refer Table 5.2). The system was able to achieve the final docking objective as seen from Figure 5.15.

Influence of Ocean Current Rates

The effect of intensities of ocean currents to the system response is studied in this section. The research [87] discuss the deep ocean current rates at 1000m and 3000m depth which are recorded over a period of 20 years. A similar study [88] was done for ocean current rates at 9m to 36m for varying hours. In [89], it is noted that the mean speed in the core of the current are high during periods of prolonged upwelling and favorable winds. An analysis of these literature shows that most of the time, the ocean current rate is limited between 0m/s to 0.3m/s . However, in this test, a worst case scenario is considered where the ocean current rate is assumed to be 0.6m/s . An LOS mission at constant depth is executed to study the system response. The system is found to be robust enough to compensate even constant ocean current rates of 0.6m/s as seen in Figure 5.16.

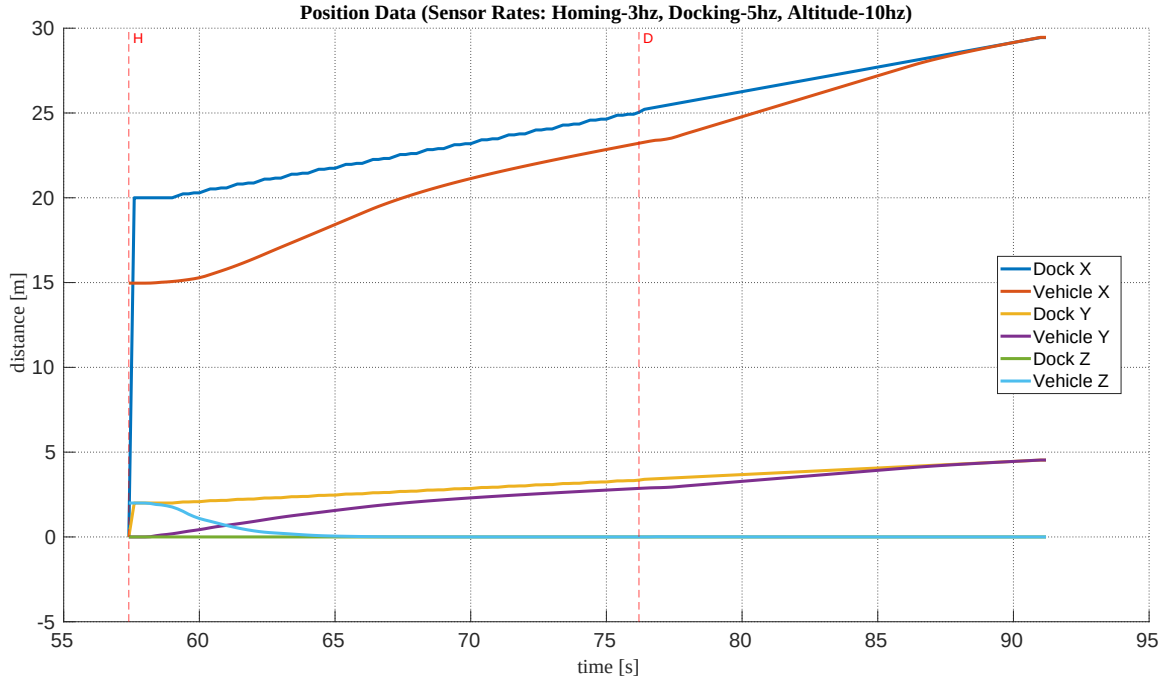


Figure 5.15 X300: Time behaviour of the docking station position data and vehicle position data with low sensor data sampling rates. The various phases of the test are denoted in the plot as follows: H - Homing, D - Docking.

5.2.2 Docking approaches

This section gives more insight into the various docking approaches proposed in Section 4.4 for various types of AUVs and docking stations. The approaches can be suitably selected depending on the constraints imposed on the vehicle or the docking station. Possible docking approaches are also suggested for non-holonomic vehicles keeping into account that they cannot translate. The envelop obstacles around the docking station are represented by the green spheres .

Docking to left or right of the docking station

Here the vehicle is allowed to dock to either the left or right side of the docking station as depicted in Figure 5.17. In case of left side docking, the pre-docking position is defined farther to the left of the docking station and in case of right side docking, the the pre-docking position is defined farther to the left of the docking station. Once the vehicle attains its pre-docking position and pre-docking orientation using Action $\mathcal{A}_p$, it can either translate to the right for docking to left side of the docking station or translate to the left for docking to the right side of the docking station using Action $\mathcal{A}_d$. Because of the lateral translation

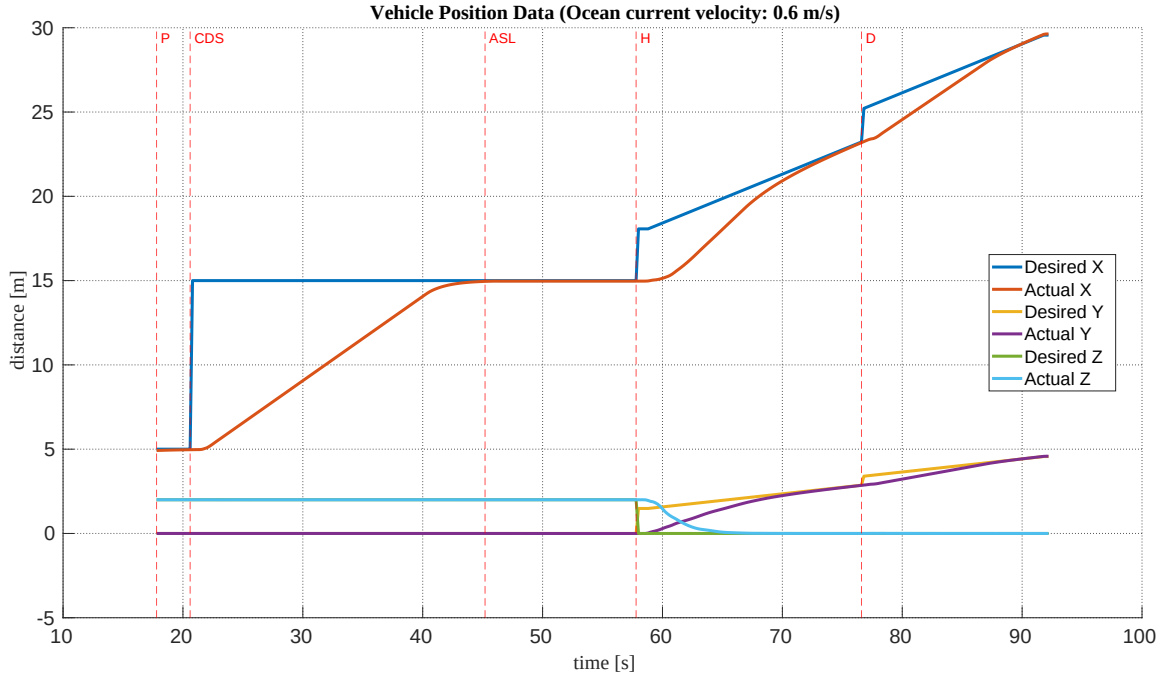


Figure 5.16 X300: Time behaviour of the vehicle position data with very high ocean current rates. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.

maneuver, this approach is well suited only for holonomic vehicles. Docking stations situated at the surface or mid-water column can safely allow this approach. The pre-docking position can be adjusted, allowing only minimum lateral translation to reduce energy expenditure and to maintain a collision free docking. Actions $\mathcal{A}_p$ and $\mathcal{A}_d$ are used for this test. The green spheres represent the envelope obstacle around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

Docking to top of the docking station

There the vehicle is allowed to dock to the top side of the docking station as depicted in Figure 5.18. The pre-docking position is defined farther above the docking station. Once the vehicle attains its pre-docking position and pre-docking orientation using Action $\mathcal{A}_p$, it can translate down towards the docking station to dock using Action $\mathcal{A}_d$. Because of the vertical translation maneuver, this approach is also well suited only for holonomic vehicles. Docking stations situated at the mid-water column and sea-floor can safely allow this approach. This approach can also be used for partially enclosed or constrained docking situations if there is no obstruction at the top side of the docking station. This will be discussed more in Section

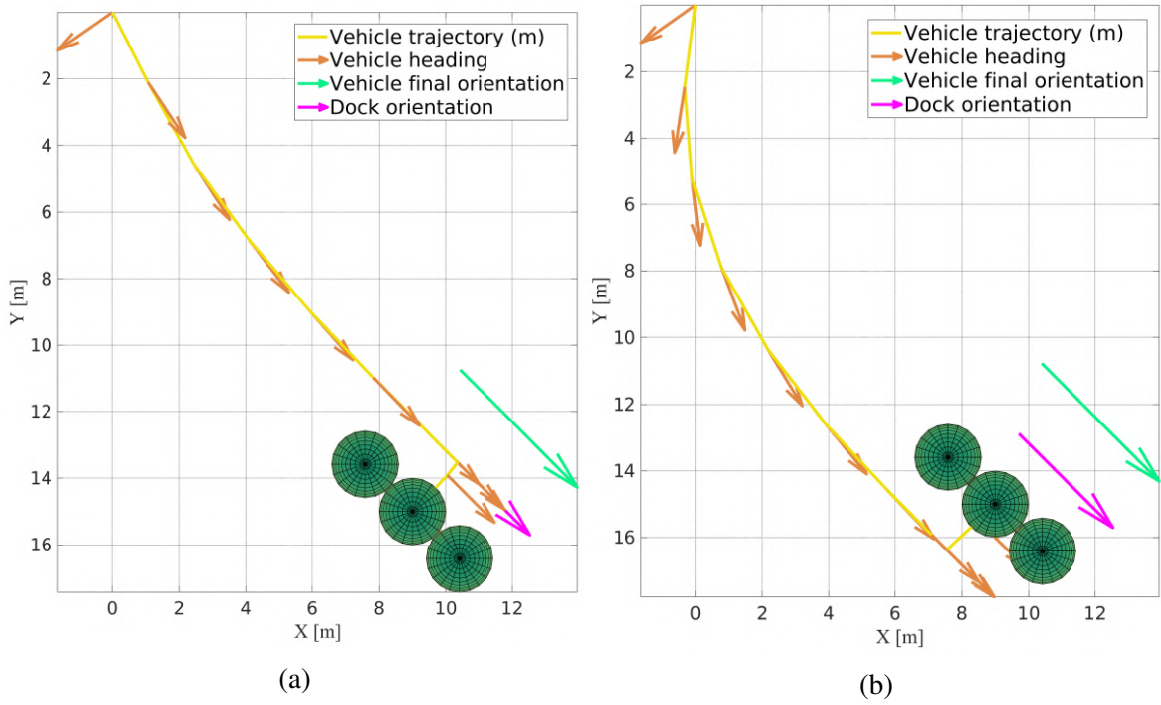


Figure 5.17 X300: Maneuver showing the vehicle approaching a dock and docking at (a) left and (b) right.

5.2.4. The pre-docking position can be adjusted, allowing only minimum vertical translation to reduce energy expenditure and to maintain a collision free docking. Actions $\mathcal{A}_p$ and $\mathcal{A}_d$ are used for this test. The green spheres represent the envelope obstacle around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

Docking to bottom of the docking station

There the vehicle is allowed to dock to the bottom side of the docking station as depicted in Figure 5.19. The pre-docking position is defined farther below the docking station. Once the vehicle attains its pre-docking position and pre-docking orientation using Action $\mathcal{A}_p$, it can translate up towards the docking station to dock using Action $\mathcal{A}_d$. Because of the vertical translation maneuver, this approach is also well suited only for holonomic vehicles. This approach can also be used for partially enclosed or constrained docking situations if there is no obstruction at the bottom side of the docking station. This will be discussed more in Section 5.2.4. Docking stations situated at the surface and mid-water column can safely allow this approach. The pre-docking position can be adjusted, allowing only minimum vertical translation to reduce energy expenditure and to maintain a collision free docking. Actions $\mathcal{A}_p$ and $\mathcal{A}_d$ are used for this test. The green spheres represent the envelope obstacle

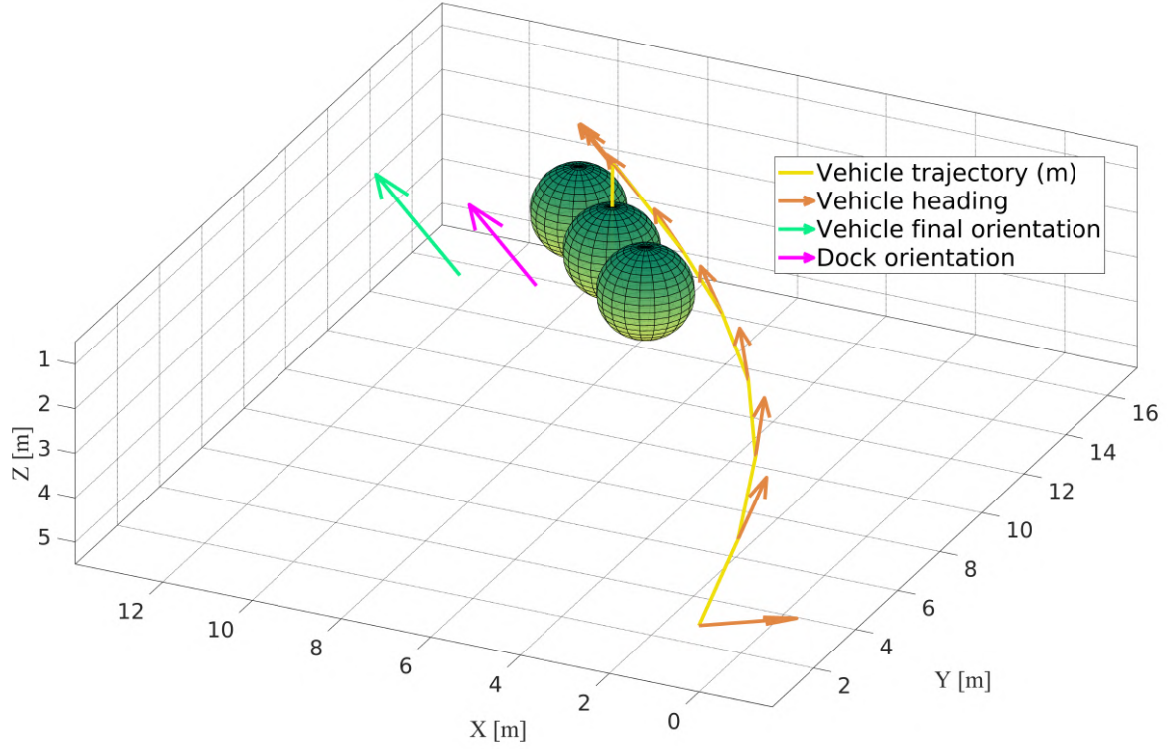


Figure 5.18 X300: Maneuver showing the vehicle approaching a dock and docking at top.

around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

Docking to rear of the docking station

There the vehicle is allowed to dock to the rear side of the docking station as depicted in Figure 5.20. The pre-docking position is defined farther behind the docking station. Once the vehicle attains its pre-docking position and pre-docking orientation using Action $\mathcal{A}_p$, it can surge towards the docking station to dock using Action $\mathcal{A}_d$. Because of the absence of lateral or vertical translation maneuver in the pre-docking or docking maneuvers, this approach is well suited for both holonomic and non-holonomic vehicles. Docking stations situated at the surface and mid-water column can safely allow this approach. This approach can also be used for partially enclosed or constrained docking situations if there is no obstruction at the rear side of the docking station. This will be discussed more in Section 5.2.3. The pre-docking position can be adjusted, allowing no lateral or vertical translation to reduce energy expenditure and to maintain a collision free docking. Actions $\mathcal{A}_p$ and $\mathcal{A}_d$ are used for this test. The green spheres represent the envelope obstacle around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

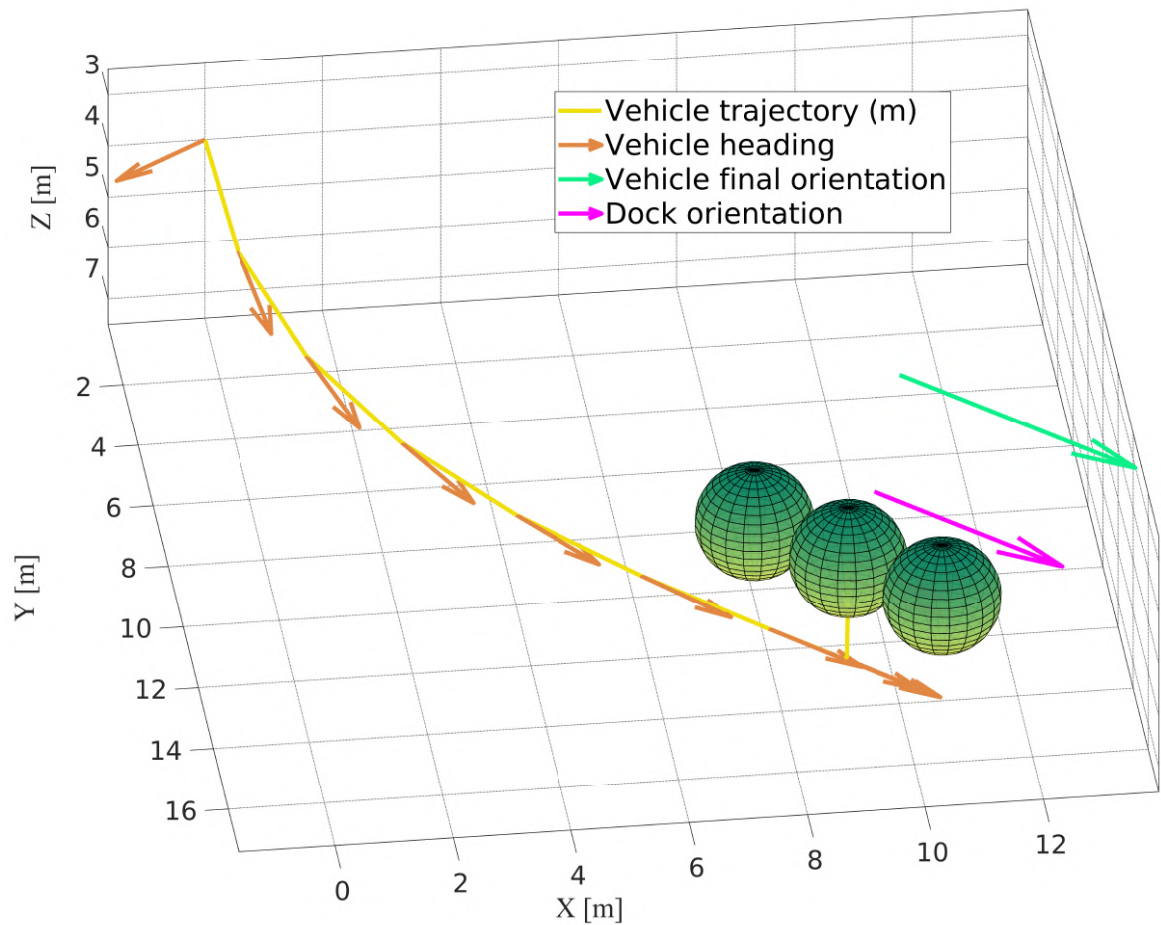


Figure 5.19 X300: Maneuver showing the vehicle approaching a dock and docking at bottom.

5.2.3 Docking tests in free space under partially enclosed or constrained situations

This section gives more information about the scenarios when the vehicle has to dock to a docking station under partially enclosed or constrained situations. It can be performed to a moving station with an initial vehicle configuration extremely close to the station itself, or to a partially enclosed or space-limited station or into a tight or constrained space. The green spheres represent the envelope obstacle around the docking station to avoid collision as the vehicle moves closer during homing and approach maneuver.

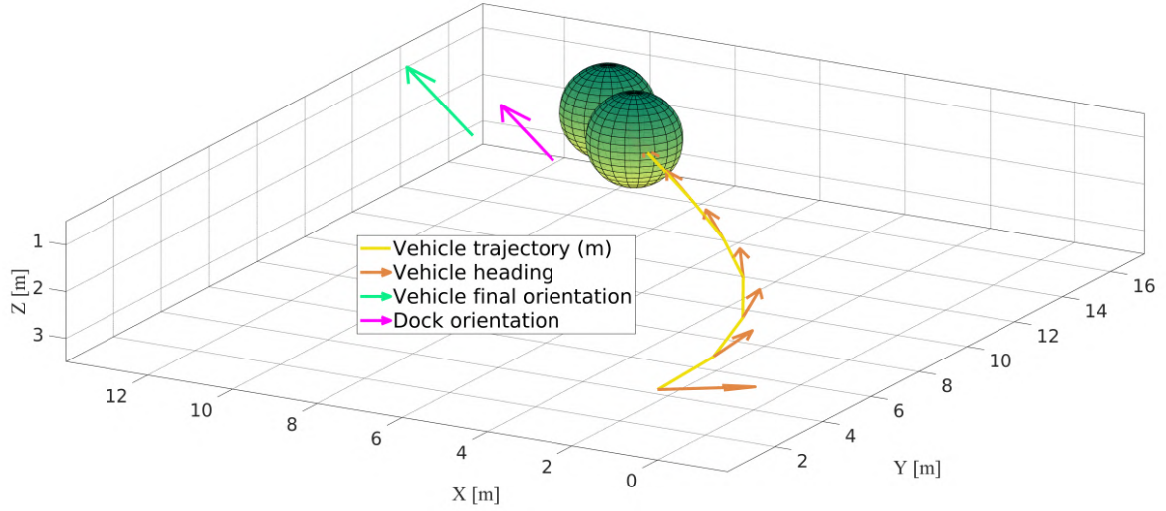


Figure 5.20 X300: Maneuver showing the vehicle approaching a dock and docking at rear.

Docking to partially enclosed moving docking station situated away from the vehicle

Figure 5.21 shows the behavior of the vehicle when it needs to dock into a tight space, such as space-limited docks or piers. The use of pre-docking position and orientation allows the vehicle to home into the required position and orientation for docking without colliding with the surrounding docking station or pier structure. To complete the docking maneuver, the vehicle finally surges towards the final dock position inside a tight space partially enclosed by the docking station. In comparison to [74], the lateral translation is not used for the terminal docking maneuver, thereby also reducing drag and energy consumption. Actions $\mathcal{A}_p$ and $\mathcal{A}_d$ are used for this test. The green spheres represent the envelope obstacle around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

Docking to partially enclosed moving docking station with initial configuration of the vehicle extremely close to the docking station

Figure 5.22 depicts the behavior of the vehicle while performing a docking maneuver around a moving docking station at close proximity, replicating the scenario presented in [71]. Here the vehicle starts at close proximity from one side of the docking station and travels to the other side of the docking station without colliding with the structure of the docking station. Thanks to the tasks *Obstacle avoidance* and *Field Velocity* that are integrated to our unified framework, the vehicle is able to move away from the close proximity of the station to go around and reach the pre-docking position defined at the rear side of the docking station. It follows a velocity field trajectory before surging forward towards the final dock

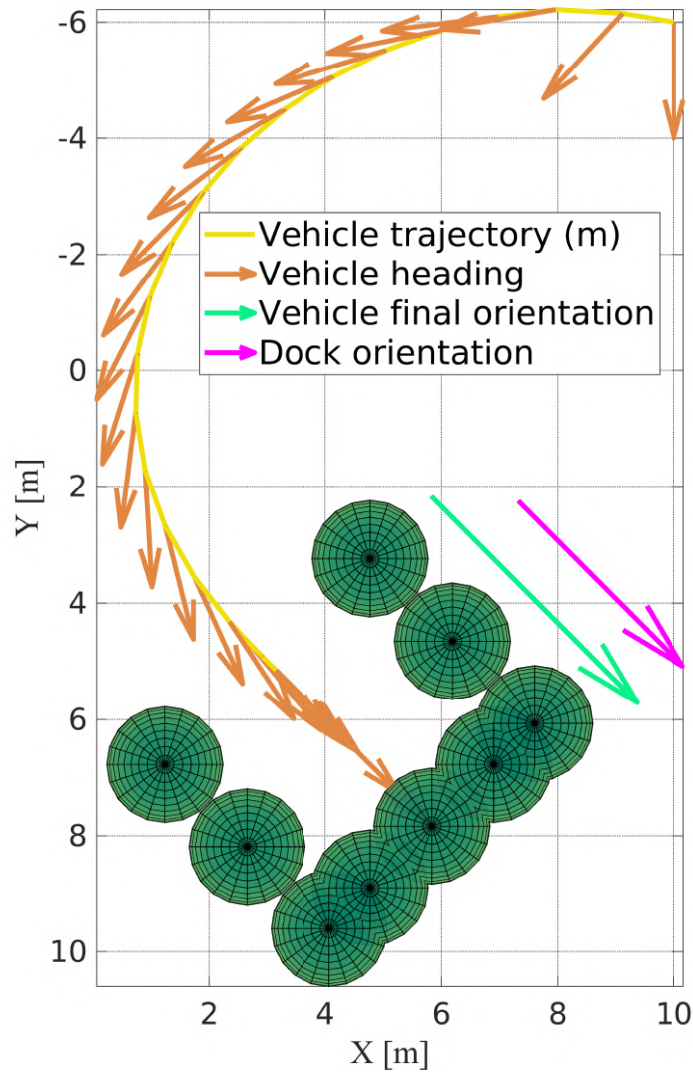


Figure 5.21 X300: Maneuver showing the vehicle performing docking into a tight space inside a moving docking station, starting from a distance.

position inside a tight space partially enclosed by the docking station, completing the docking maneuver. In comparison to [74], the lateral translation is not used for the terminal docking maneuver, thereby also reducing drag and energy consumption. Actions $\mathcal{A}_p$ and $\mathcal{A}_d$ are used for this test. The green spheres represent the envelope obstacle around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

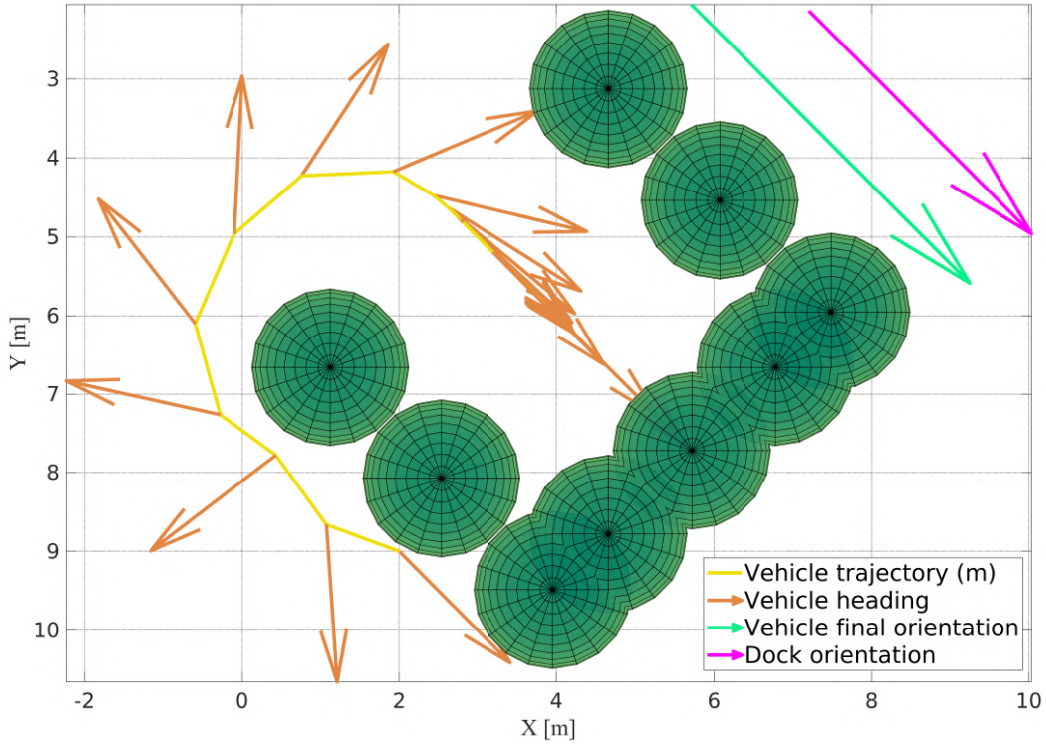


Figure 5.22 X300: Maneuver showing the vehicle performing docking into a tight space, starting from close proximity of the moving docking station and going around the docking station structure without collision.

Docking to partially enclosed moving docking station travelling in same direction of the vehicle

Figure 5.23 depicts the behavior of the vehicle while performing a docking maneuver to the rear side of a partially enclosed docking station travelling in the same direction of that of the vehicle. With the help of *Obstacle avoidance* and *Field Velocity* tasks, the vehicle follows a curvilinear velocity field trajectory away from the structure of docking station and towards the pre-docking position defined at a point in the rear side of the docking station. To complete the docking maneuver, the vehicle finally surges towards the final dock position inside a tight space partially enclosed by the docking station. In comparison to [74], the lateral translation is not used for the terminal docking maneuver, thereby also reducing drag and energy consumption. The $\mathcal{A}_p$, and $\mathcal{A}_d$ actions are exploited in these tests. The green spheres represent the envelope obstacle around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

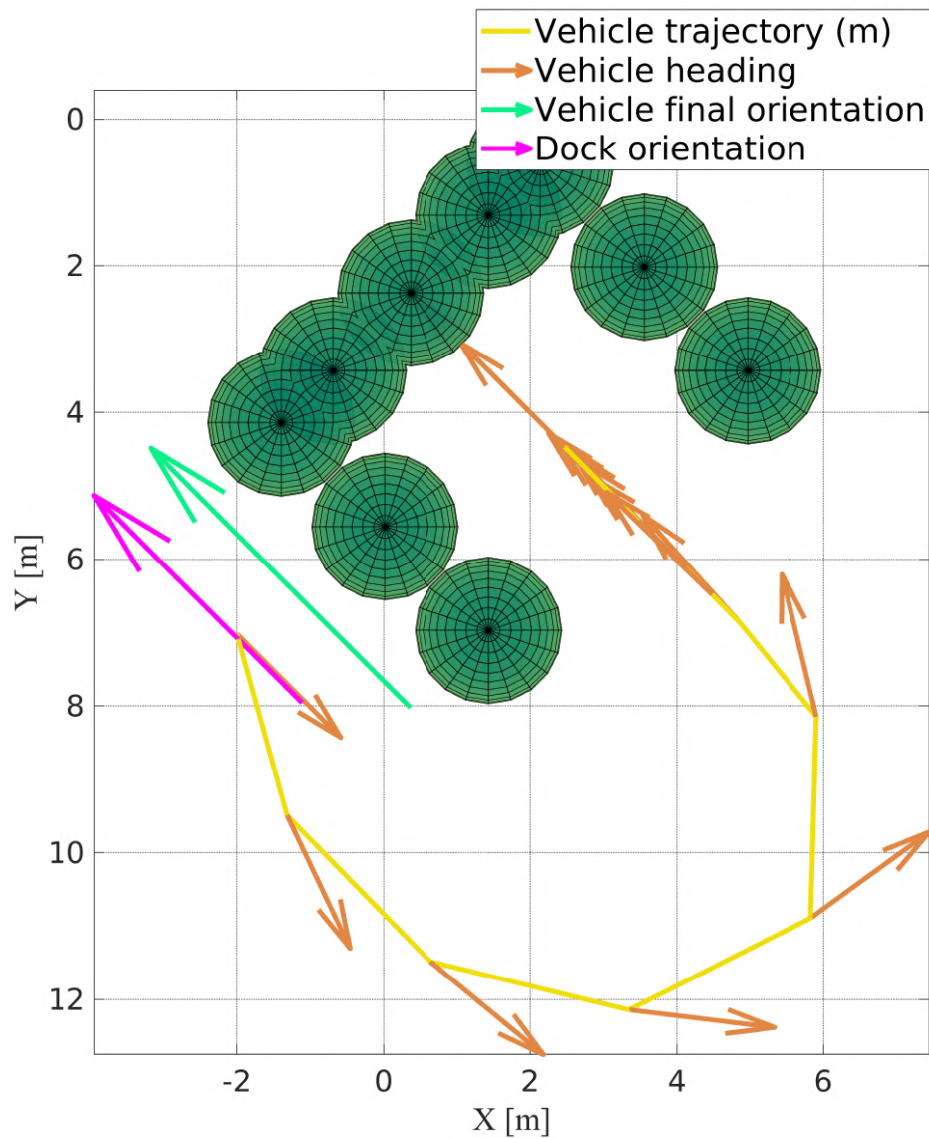


Figure 5.23 X300: Maneuver showing the vehicle performing a docking to a moving docking station travelling in same direction of the vehicle.

Docking to partially enclosed moving docking station travelling in the opposite direction of the vehicle

Figure 5.24 depicts the behavior of the vehicle while performing a docking maneuver at the rear side of a partially enclosed docking station initially travelling in front and towards the vehicle. With the help of *Field Velocity* Task, the vehicle follows a curvilinear velocity field trajectory towards the pre-docking position defined at a point in the near-vicinity of the station. To completing the docking maneuver, the vehicle finally surges towards the final

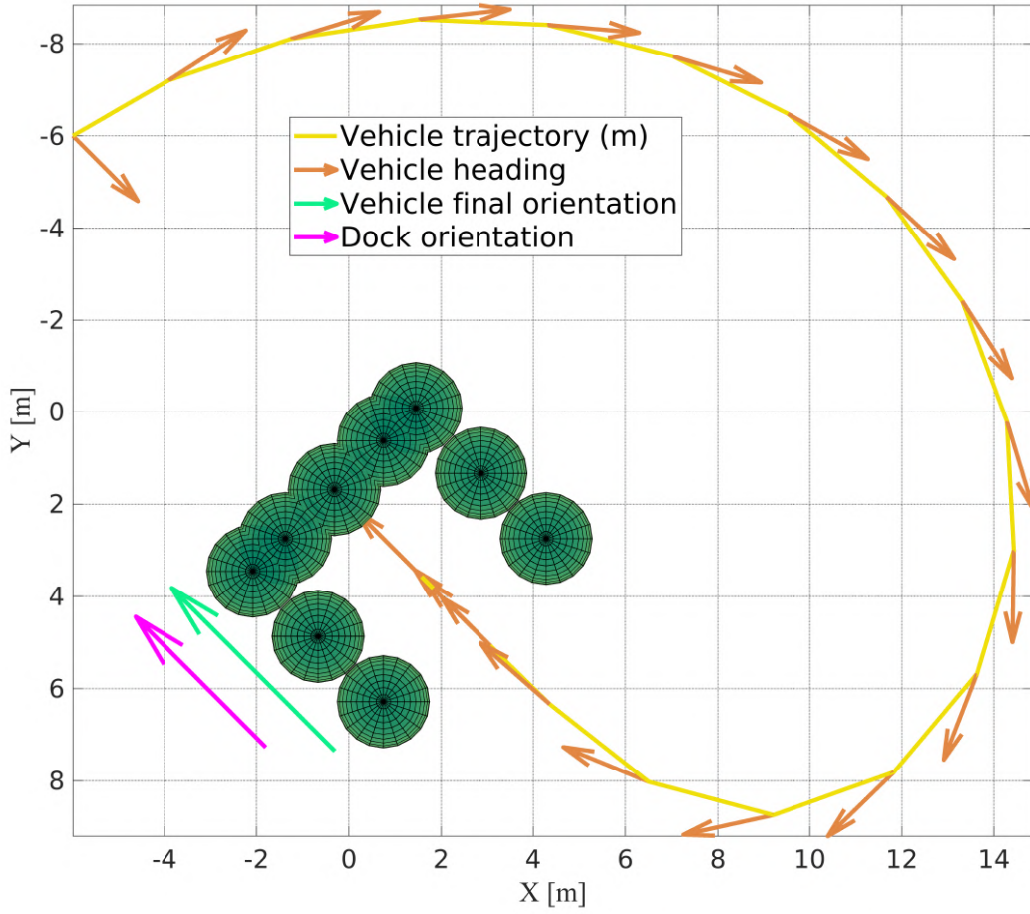


Figure 5.24 X300: Maneuver showing the vehicle performing a docking to a moving docking station travelling towards the vehicle from opposite side.

dock position inside a tight space partially enclosed by the docking station. In comparison to [74], the lateral translation is not used for the terminal docking maneuver, thereby also reducing drag and energy consumption. The $\mathcal{A}_p$, and $\mathcal{A}_d$ actions are exploited in these tests. The green spheres represent the envelope obstacle around the docking station to avoid collisions as the vehicle moves closer during homing and approach maneuver.

5.2.4 Complete Mission tests

This section highlights the simulations encompassing the behaviors of the system in various complex scenarios, using constant altitude survey, constant depth survey and resident AUV survey. They serve as representative of possible missions, exploiting the features of the unified task priority framework. These mission scenarios, which are performed and studied in the following sections, will exploit all the behaviours of the system that were discussed

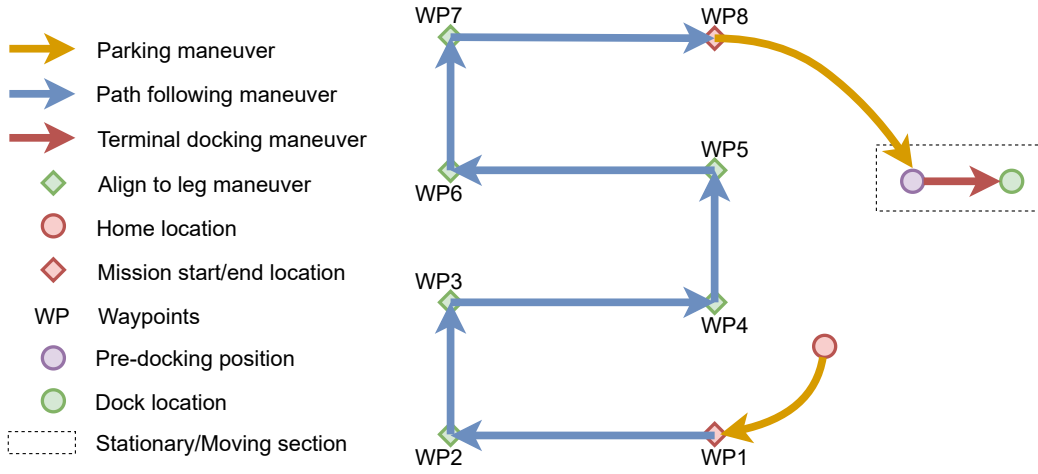


Figure 5.25 Lawn mover Profile: Representation of various locations, waypoints, and maneuvers involved for the complete survey mission.

in contributions **C3** and **C4** in Section 1.1. The following the scenarios are considered, one where the AUV executes a constant altitude survey (e.g., to acquire optical data at a prescribed distance from the seafloor for pipeline inspection purposes), one where the AUV executes a constant depth survey (e.g., to simulate the acquisition of acoustic sensor data from the seafloor for bathymetry purposes) and one where a resident AUV executes a constant depth survey (e.g., to simulate the acquisition of acoustic sensor data from oil rig structures for inspection purposes). The ability to dock onto stationary and moving docking stations are also evaluated in this study. Obstacles (represented by red spheres) and envelop obstacle (represented by green spheres around the dock) are also introduced in the following mission tests.

The missions are carried out as follows. A lawn mower profile is chosen for the survey mission exploiting the LOS path following algorithm [79] as depicted in Figure 5.25. Here this mission profile is a general representation of all missions. In real world scenario, the actual profile could vary depending on the mission requirements. Waypoints are inserted wherever the vehicle needs to make a turn during the mission. The waypoint locations, vehicle initial position and orientation, the dock velocity (if the dock is moving) and the ocean current velocity used in the simulations are reported in Table 5.4. The vehicle starts from the home (deployment) location and proceeds towards the 1st waypoint by following and aligning to the system velocities with the help of action $\mathcal{A}_p$. The vehicle transits via each waypoint using LOS based control at constant altitude using $\mathcal{A}_{sa}$ action or constant depth using $\mathcal{A}_{sd}$ action, along with the survey leg alignment $\mathcal{A}_a$ action. At every stage of the mission, the vehicle reactively avoids the obstacles along its path with the help of *Obstacle*

Parameter	Value
Vehicle initial pose / Home (m, deg)	[-5, -10, 18, 0, 0, 40]
Waypoint 1 pose (m, deg)	[0, -5, 20.0, 0, 0, 0]
Waypoint 2 pose (m, deg)	[20, -5, 20.0, 0, 0, 90]
Waypoint 3 pose (m, deg)	[20, 0, 20.0, 0, 0, 180]
Waypoint 4 pose (m, deg)	[0, 0, 20.0, 0, 0, 90]
Waypoint 5 pose (m, deg)	[0, 5, 20.0, 0, 0, 0]
Waypoint 6 pose (m, deg)	[20, 5, 20.0, 0, 0, 90]
Waypoint 7 pose (m, deg)	[20, 10, 20.0, 0, 0, 180]
Waypoint 8 pose (m, deg)	[0, 10, 20.0, 0, 0, 180]
Dock velocity (m/s) (in the dock frame)	[0.3, 0, 0]
Ocean current velocity (m/s) (in the world frame)	[0, 0.2, 0]

Table 5.4 Mission parameters. All the positions are defined w.r.t. the world frame.

Avoidance Task. A minimum altitude is always maintained using *Minimum Altitude Task* to avoid crashing into the seafloor or any bottom structures. While travelling close to seafloor, the vehicle follows the terrain using *Seafloor Alignment Task* to avoid crashing into it when maneuvering over rough terrain. Once the vehicle completes its mission, the homing stage will start using the action $\mathcal{A}_p$. The action will drive the AUV to the associated pre-docking position, defined w.r.t. the docking station as stated earlier. Thanks to the *Velocity Alignment Task* defined in action $\mathcal{A}_p$, the vehicle will align its heading to the system velocities as it proceeds towards the pre-docking position, whilst gradually aligning to the dock orientation using *Field Velocity Task*. Once the misalignment errors are within the acceptable range, the vehicle surges or translates towards the final dock position with the help of *Terminal Dock* and *Dock Attitude* tasks in the action $\mathcal{A}_d$. Some possible ways of docking approaches are depicted in Figure 4.19. In the missions docking approaches similar to the scenarios taken from [73] and [74] are presented where the vehicle docks to the moving docking stations. The classic approach is to dock from the rear of the docking station, for e.g., to dock to a towed [72] or funnel shaped [23] docking station is already presented in Section 5.2.3. For simplicity, the dock moves with a constant heading and velocity throughout the mission. Ocean current disturbances are also introduced in the following mission simulations.

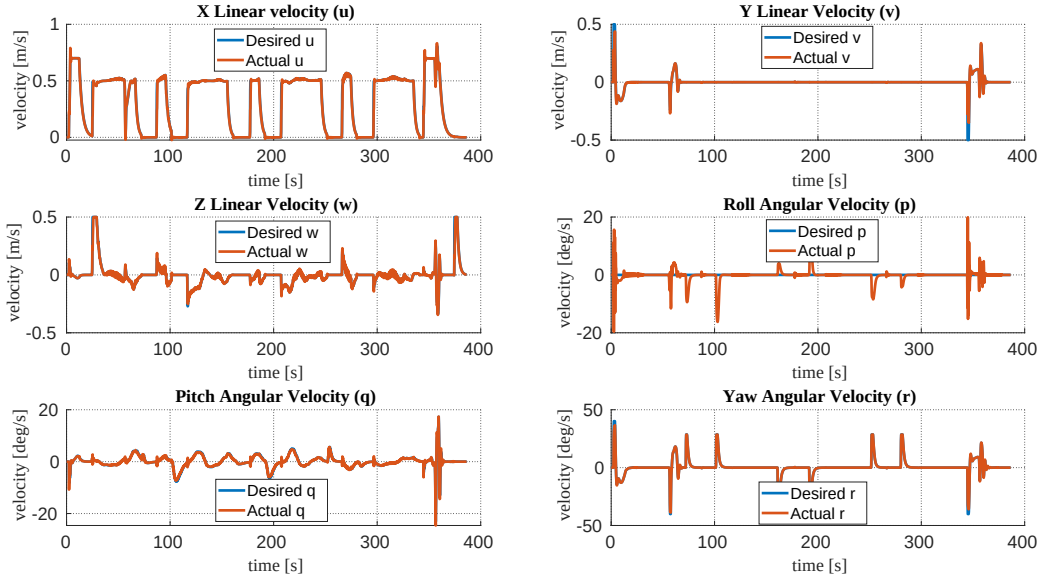


Figure 5.26 Velocity Time Behaviour: Plot depicting the time behaviour of desired velocities and the actual/system velocities of the vehicle.

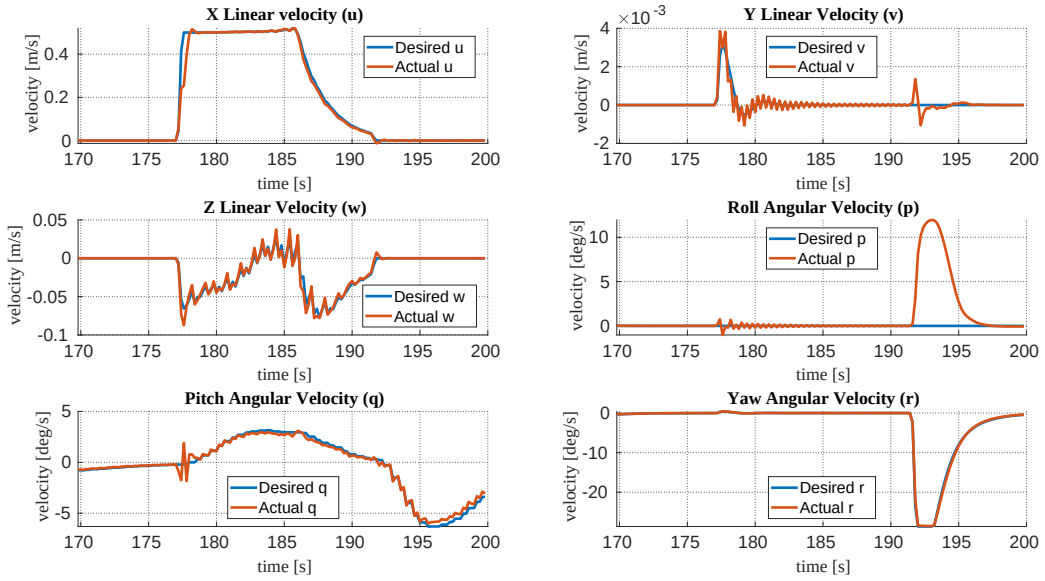


Figure 5.27 Velocity Time Behaviour: Plot depicting a magnified portion the time behaviour of desired velocities and the actual/system velocities of the vehicle.

The time behaviour of desired velocities and the actual/system velocities of the vehicle is depicted in Figure 5.26. The simulated dynamic controller is sufficiently fast, allowing the system velocities (represented in red color) to follow the desired velocities (represented in blue color) as seen from a magnified sample portion of the velocity plot in Figure 5.27.

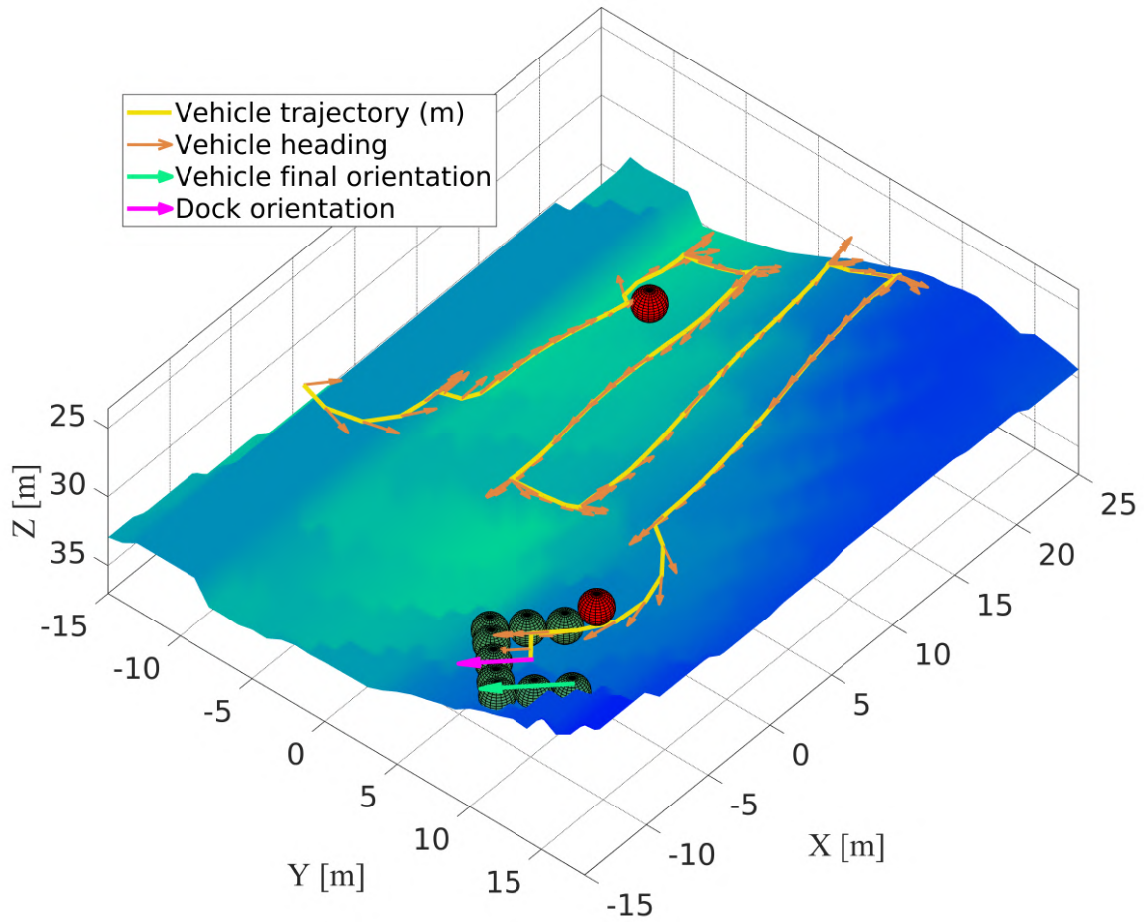


Figure 5.28 Constant altitude mission: Plot showing the mission and docking (top side) to a moving docking station situated close to the seafloor in the 3D space. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station. Notice that the vehicle avoids the 2nd red obstacle (close to the docking station) in the path, by travelling below it due to availability of free space without breaching the minimum altitude safety limit.

In the following simulations/sections, further velocity plots will be neglected as the generated/desired velocities are within the bandwidth of the controller and within reasonable times before signal changes. Also from the plots, it can be noticed how the translation on the lateral and vertical axes are mainly of a transient nature, with the exception of the terminal docking phase.

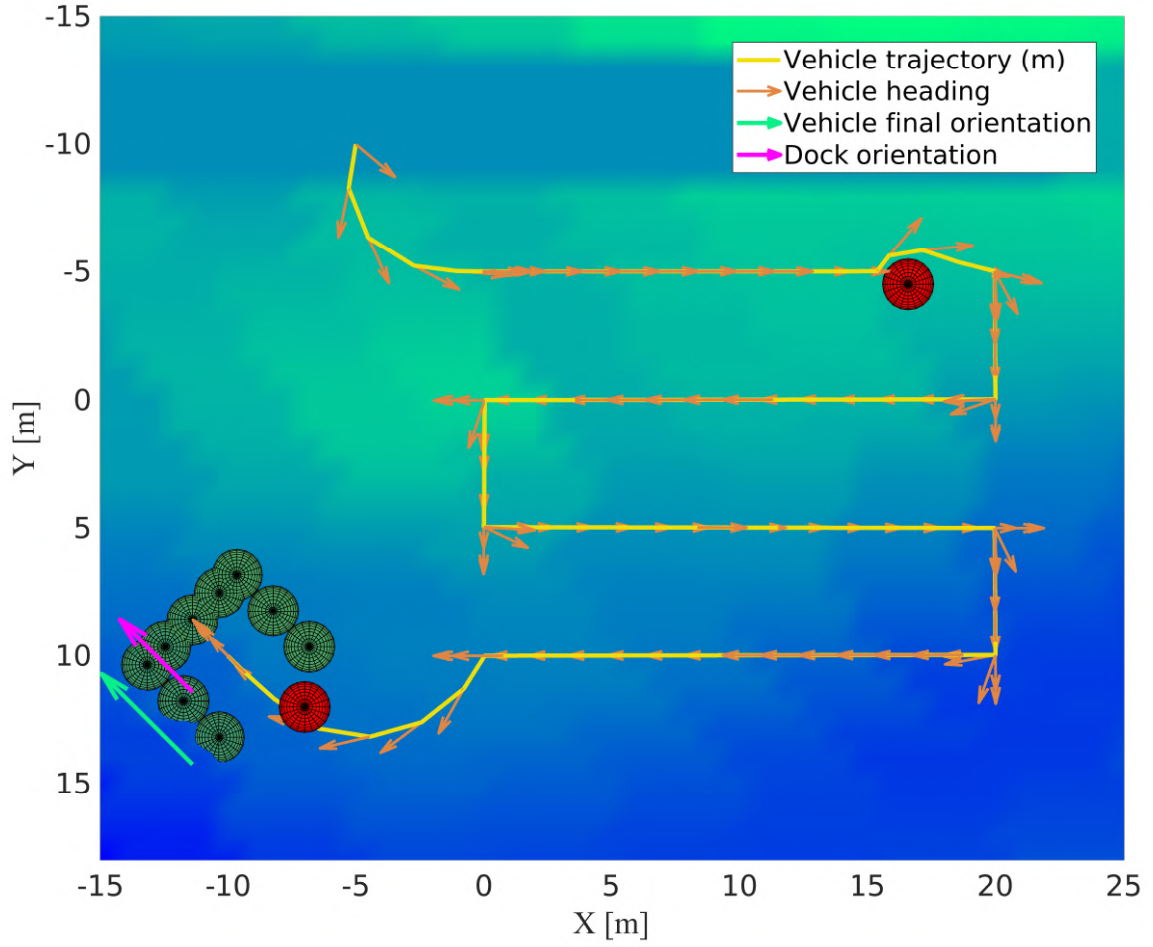


Figure 5.29 Constant altitude mission: Plot showing the mission and docking (top side) to a moving docking station situated close to the seafloor from top view. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station. Notice that the vehicle avoids the 2nd red obstacle (close to the docking station) in the path, by travelling below it due to availability of free space without breaching the minimum altitude safety limit.

Constant altitude survey mission for pipeline inspection with terrain following and docking to a moving docking station situated close to the seafloor

As discussed in Section 4.5.1, for inspection surveys, the vehicle needs to travel at a certain altitude above to the seafloor for gathering optical data for pipeline inspection and monitoring from sensors such as LIDAR or Camera. A constant altitude survey is performed in such scenarios. Figures 5.28 and 5.29 shows the complete behavior of the vehicle performing a constant altitude survey mission ($\mathcal{A}_{sa}$ and $\mathcal{A}_a$), homing ($\mathcal{A}_p$) and finally docking on the top side ($\mathcal{A}_d$) to a docking station moving at a depth close to the seafloor. After being deployed

from a submarine situated at the mid-water column, the vehicle maintains a predefined altitude from the seafloor throughout the mission. Additionally, it follows the terrain of the seafloor (an improvement over [73]) and also avoids various obstacles (represented by red spheres) if any along its path. Once the vehicle completes the mission, it enters homing phase. As the AUV is required to dock on top of the docking station, the pre-docking position is placed above the required dock position at a safety distance. The docking station is situated in front of the vehicle when the vehicle starts the homing phase. Due to the orientation error in yaw and thanks to the *Field Velocity* Task, the vehicle will make a wide turn while approaching the moving dock as seen in Figures 5.38 and 5.39. The vehicle matches its position and orientation with the pre-docking position and orientation with the help of velocity field and eventually translates down to the final dock position on top of the docking station. The spheres depicted around the docking station (represented by green spheres) are the envelope obstacle to provide a safety area around the docking station while the vehicle approaches the docking station during homing. It can also help in guiding the vehicle by pushing it further away from the docking station and around if required, towards the required approach position for a safe docking or during deployment. In this survey mission, upon reaching a waypoint, the vehicle can maintain its pitch aligned to the terrain of the seafloor even while orienting towards the next leg of the survey ($\mathcal{A}_a$). However, one can easily change this behaviour and require the AUV to maintain a horizontal attitude, while it orients towards the next leg of the survey, by simply deactivating the *Seafloor Alignment* Task in $\mathcal{A}_a$ action. This is possible only because of the flexibility of the task priority approach in introducing specific behaviours to the vehicle as described in contributions **C4** and **C5**. The state transition diagram for the Constant Altitude mission is shown in Figure 5.30.

From Figure 5.31, at around time $t = 24s$, the vehicle transitions from the parking maneuver ($\mathcal{A}_p$) to the path following using constant altitude survey maneuver ($\mathcal{A}_{sa}$ and $\mathcal{A}_a$). At time $t = 344s$, the vehicle enters homing maneuver ($\mathcal{A}_p$) and at time $t = 374s$, the docking maneuver ($\mathcal{A}_d$) is initiated. It is clear that the vehicle maintains a specific pre-defined altitude above the seafloor with the help of *Vehicle Altitude* Task during all phases of the mission. Maintaining a safe minimum altitude from the seafloor is further ensured by the *Minimum Altitude* Task during the homing phase. The vehicle pitch data shows the variation of pitch orientation of the vehicle while it follows the terrain of the seafloor with the help of *Seafloor Alignment* Task. However, due to the pitch limitation implemented via *Horizontal Alignment* Task, the vehicle pitch does not increase beyond its permissible limits. The vehicle encounters two obstacles along its path as depicted by spheres of radius 1m at positions [16.6, -4.5, 32.5] and [-7, 12, 27]. Obstacle avoidance data from Figure 5.31 shows that the vehicle

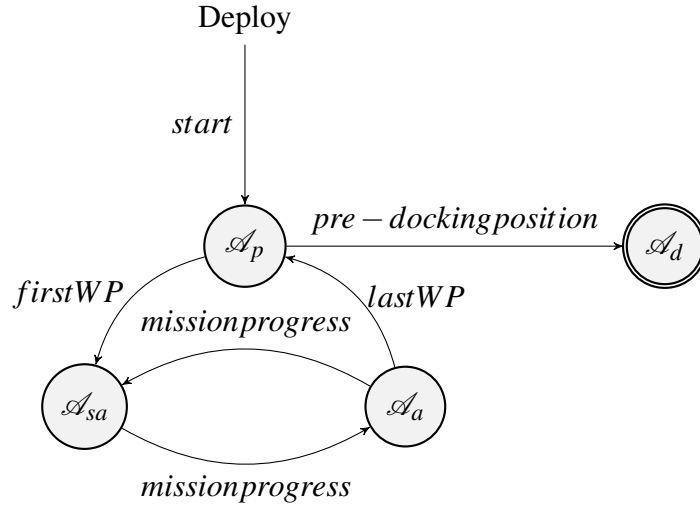


Figure 5.30 State Transition Diagram: Constant altitude survey.

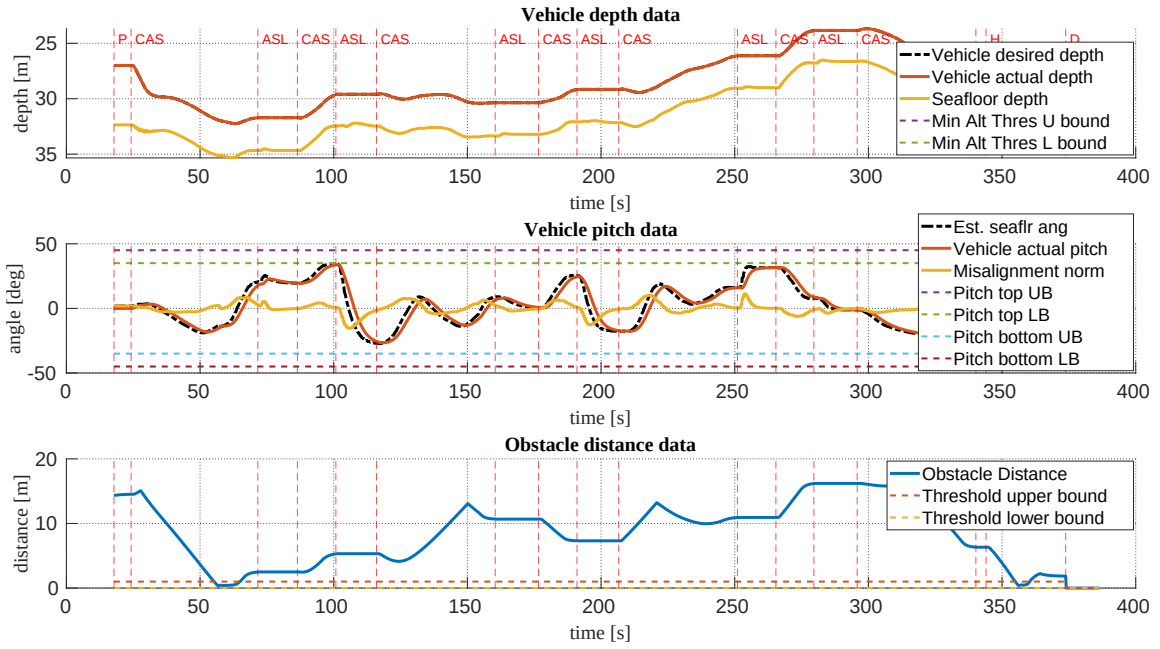


Figure 5.31 Constant altitude mission: Time behaviour of the vehicle depth, pitch and obstacle avoidance data. The various phases of the mission are denoted in the plot as follows: P - Parking, CAS - Constant Altitude Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.

is able to estimate the distance to the obstacles (only the distance to the nearest obstacle at each instant is plotted) and reactively avoids them along the path. Looking at Figures 5.28 and 5.29, notice again that the 2nd obstacle (found during the homing stage) is avoided by traveling beneath it, due to the availability of free space without conflicting with the desired

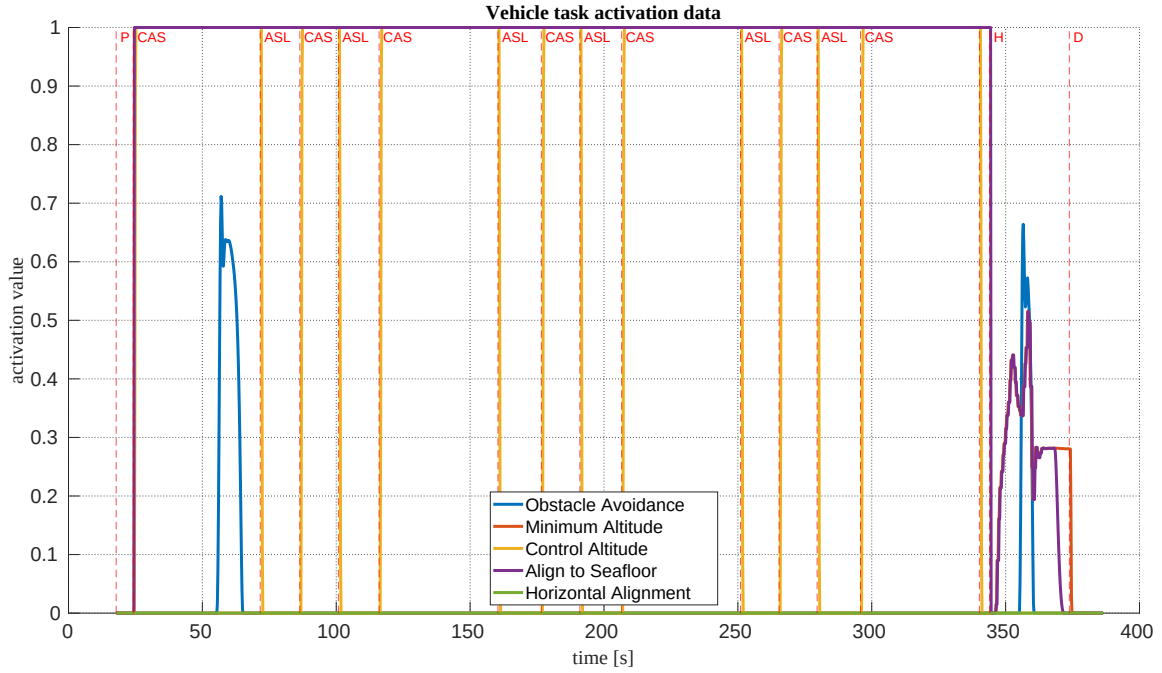


Figure 5.32 Constant altitude mission: Time behaviour of Obstacle Avoidance, Minimum Altitude, Vehicle Altitude, Seafloor Alignment and Horizontal Alignment task activation functions. The various phases of the mission are denoted in the plot as follows: P - Parking, CAS - Constant Altitude Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.

minimum altitude from the seafloor. Figure 5.32 highlights the time behavior of the tasks that are used in the mission (*Obstacle Avoidance*, *Minimum Altitude*, *Vehicle Altitude*, *Seafloor Alignment*) and homing (*Obstacle Avoidance*, *Minimum Altitude*, *Seafloor Alignment*) phases. In this mission, the roll values of the vehicle is found to vary from -4.5° to 6° , therefore confirming that the roll angles are negligible and acceptable.

Constant depth survey mission for bathymetry and docking to a moving docking station situated close to the surface

As discussed in Section 4.5.2, for bathymetry purposes, the vehicle needs to travel at a constant depth for gathering acoustic data from sensors such as SONAR or MBES. Here the vehicle performs a survey mission at constant depth. Figures 5.33 and 5.34 shows the vehicle performing the mission, homing and then docking to a moving docking station close to the surface. After being deployed from an ASV situated at the surface, the vehicle performs the constant depth survey mission ($\mathcal{A}_{sd}$ and $\mathcal{A}_a$) by maintaining a predefined depth throughout the mission. The *Seafloor Alignment* Task in the $\mathcal{A}_a$ action is disabled while performing the constant depth survey mission phase. Once the mission is completed, the vehicle enters

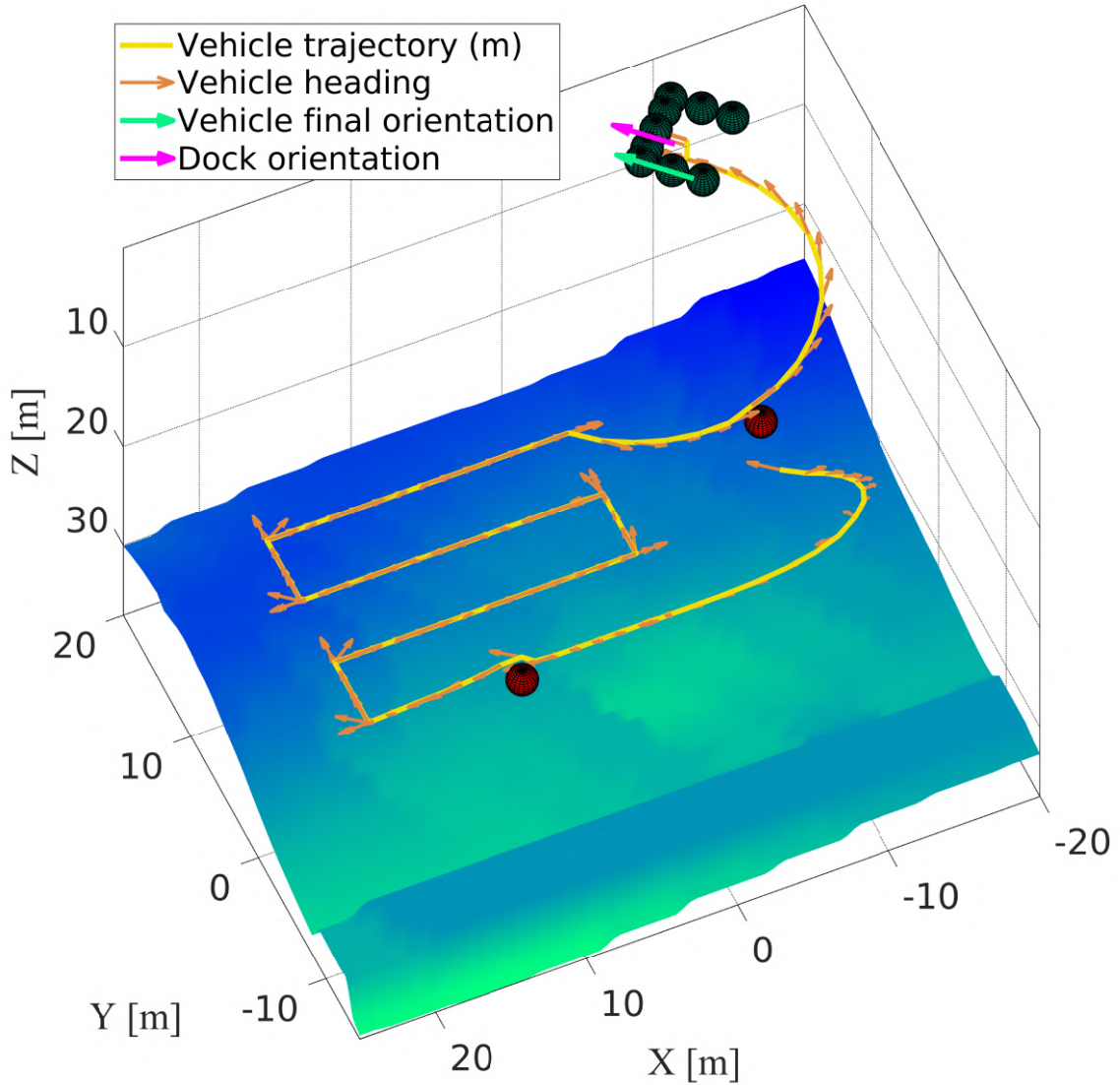


Figure 5.33 Constant depth mission: Plot showing the mission and docking (bottom side) to a moving docking station situated close to the surface in the 3D space. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station.

homing phase ($\mathcal{A}_p$) and returns to a docking station situated close to the surface. The vehicle is required to dock to the bottom side of the moving docking station and hence the pre-docking position is placed at a position to the bottom of the required dock position. Here, as the vehicle starts the homing phase, the docking station is moving in front of the vehicle in the opposite direction. Since the orientation error in yaw is large and thanks to the *Field Velocity* Task, the vehicle will make a wider turn while approaching the moving dock as seen in Figures 5.33 and 5.34. The vehicle, after matching its position and orientation with the

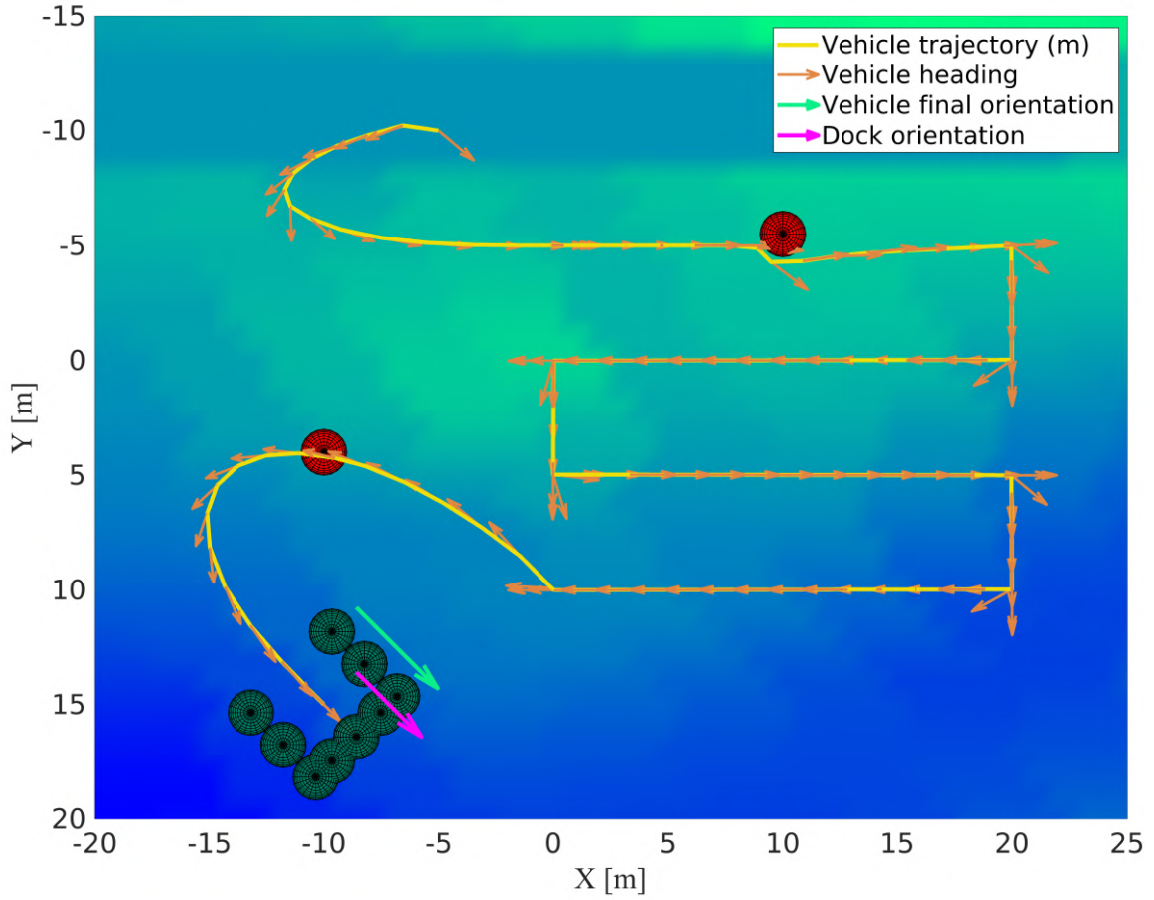


Figure 5.34 Constant depth mission: Plot showing the mission and docking (bottom side) to a moving docking station situated close to the surface from top view. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station.

pre-docking position and orientation using velocity field, eventually translate upward to the final dock position on the vehicle ($\mathcal{A}_d$). At every phase, the vehicle reactively avoids the obstacles (represented by red spheres) if any along its path. The spheres depicted around the docking station (represented by green spheres) are the envelope obstacle to provide a safety area around the docking station while the vehicle approaches the dock during homing. It can also help in guiding the vehicle by pushing it further away from the docking station and around if required, towards the required approach position for a safe docking or during deployment. The state transition diagram for the Constant Depth mission is shown in Figure 5.35.

From Figure 5.36, at around time $t = 52s$, the vehicle changes from the parking maneuver ($\mathcal{A}_p$) to the path following at constant depth survey maneuver ($\mathcal{A}_{sd}$ and $\mathcal{A}_a$). At time $t = 367s$,

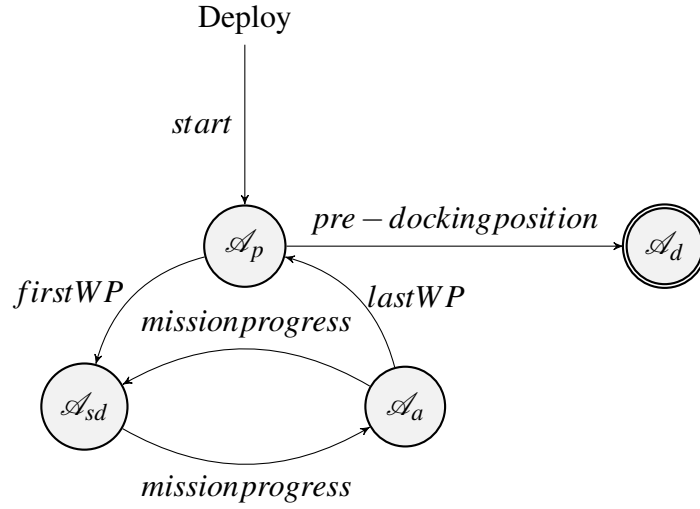


Figure 5.35 State Transition Diagram: Constant depth survey.

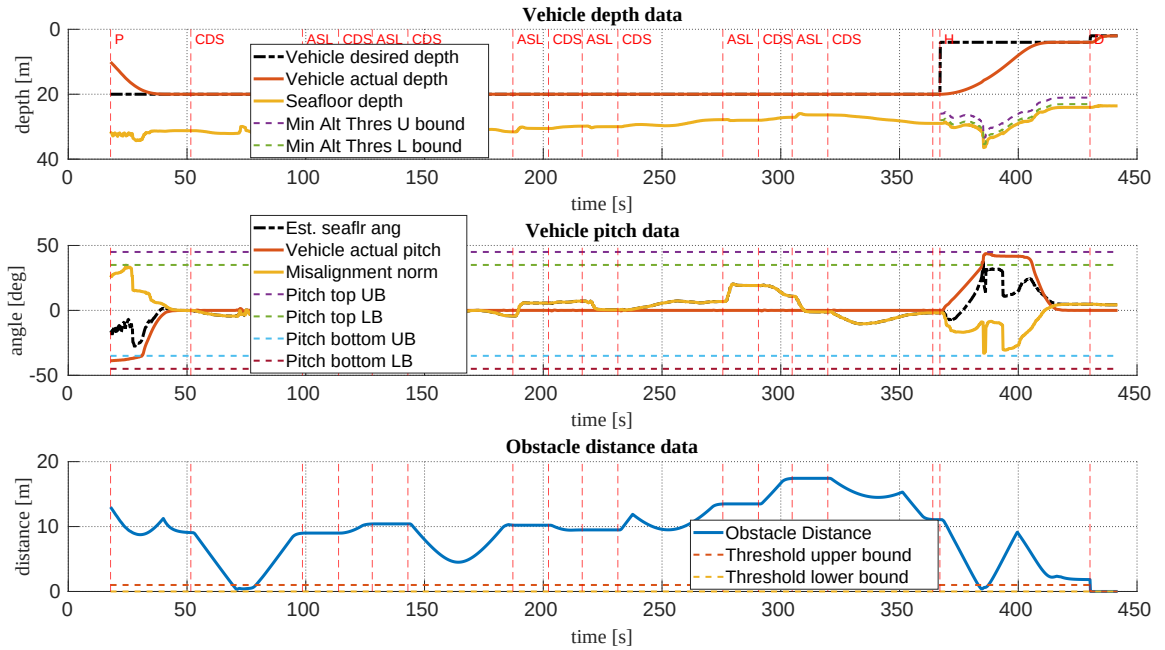


Figure 5.36 Constant depth mission: Time behaviour of the vehicle depth, pitch and obstacle avoidance data. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.

the vehicle starts the homing maneuver ($\mathcal{A}_p$) and at time $t = 430s$, the docking maneuver ($\mathcal{A}_d$) is initiated. During the mission, the depth of the vehicle is maintained at the desired level by the *Vehicle Depth Task*. However, the vehicle encounters two obstacles along

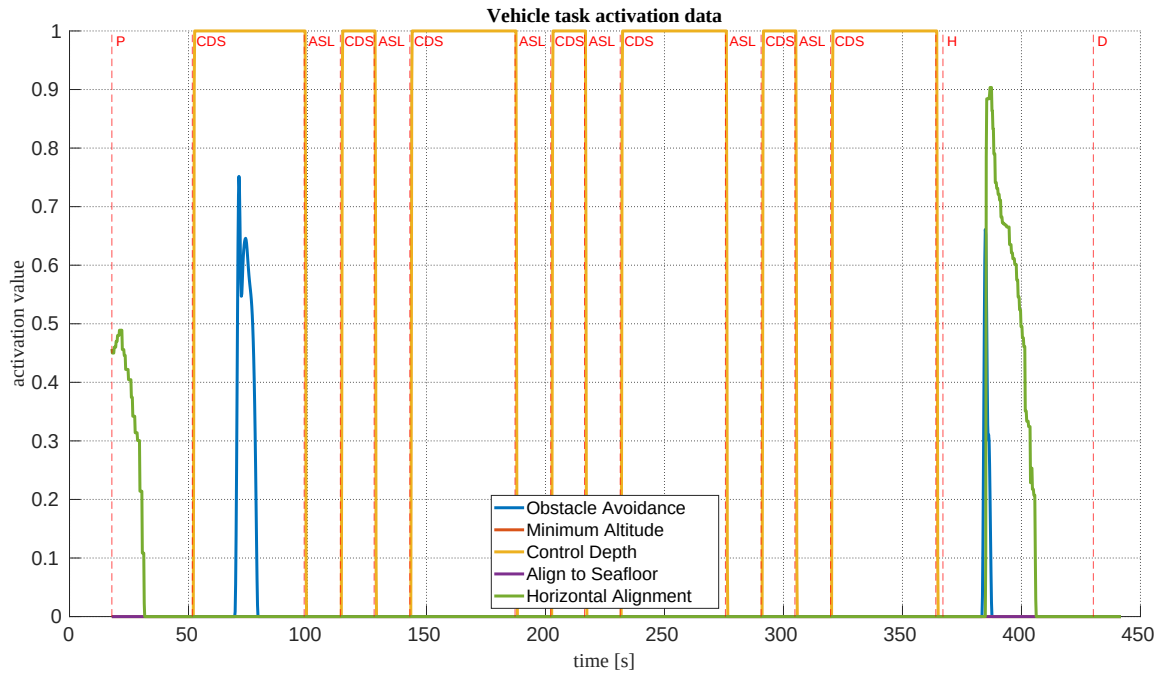


Figure 5.37 Constant depth mission: Time behaviour of Obstacle Avoidance, Minimum Altitude, Vehicle Depth, Seafloor Alignment and Horizontal Alignment task activation functions. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.

its path as depicted by spheres of radius 1m at positions [10, -5.5, 20.5] and [-10, 4, 17]. Obstacle avoidance data from Figure 5.36 shows that the vehicle is able to estimate the distance to the obstacles (only the distance to the nearest obstacle at each instant is plotted) and reactively avoids them along the path. Looking at Figures 5.33 and 5.34, notice how the obstacle found during the homing stage is avoided by traveling above it. Figure 5.37 highlights the time behavior of the tasks that are used in the mission (*Obstacle Avoidance*, *Minimum Altitude*, *Vehicle Depth*) and homing (*Obstacle Avoidance*, *Minimum Altitude*, *Seafloor Alignment*) phases. In this mission, the roll values of the vehicle vary from -4° to 4° , therefore confirming that the roll angles are negligible and acceptable.

Resident AUV survey mission for oil-rig inspection with docking to a docking station situated close to the seafloor

As discussed in Section 4.5.3, for inspection of oil rig structures, the resident AUV vehicle needs to travel at a constant depth for gathering acoustic data from sensors such as SONAR. Here the vehicle performs a survey mission at constant depth to for inspection of the structures. Figures 5.38 and 5.39 shows the vehicle performing the mission, homing and then docking

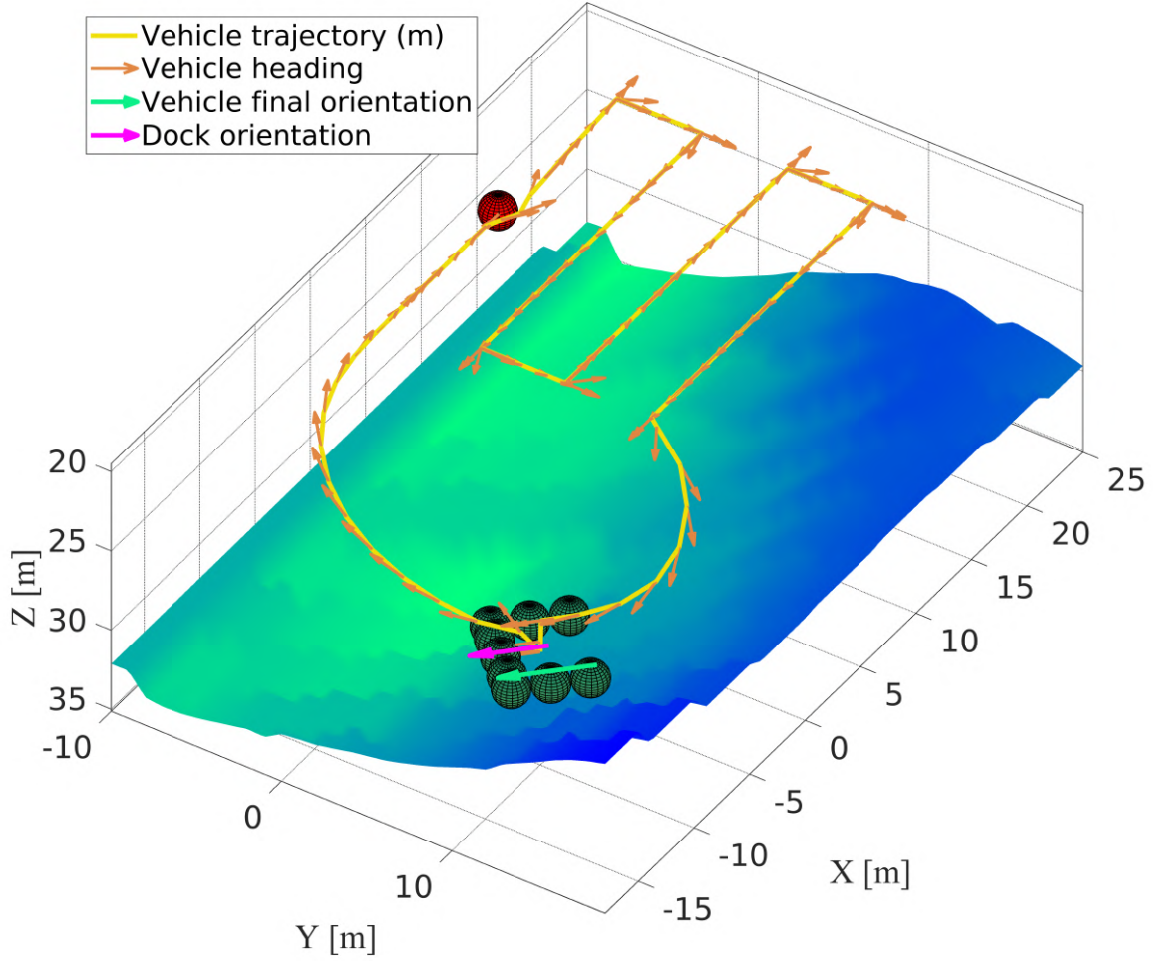


Figure 5.38 Resident AUV mission: Plot showing the mission and docking (top) to a docking station situated close to the seafloor in the 3D space. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station.

back to a the docking station situated close to the seafloor. After being deployed from the docking station situated close to the seafloor, the vehicle performs the constant depth survey mission ($\mathcal{A}_{sd}$ and $\mathcal{A}_a$) by maintaining a predefined depth throughout the mission. The *Seafloor Alignment* Task in the $\mathcal{A}_a$ action is disabled while performing the constant depth survey mission phase. Once the mission is completed, the vehicle enters homing phase ($\mathcal{A}_p$) and returns to the same docking station situated close to the seafloor. The vehicle is required to dock to the top side of the docking station and hence the pre-docking position is placed at a position to the top of the required dock position on the docking station. The docking station is situated in front of the vehicle when the vehicle starts the homing phase. Due to the orientation error in yaw and thanks to the *Field Velocity* Task, the vehicle will make a

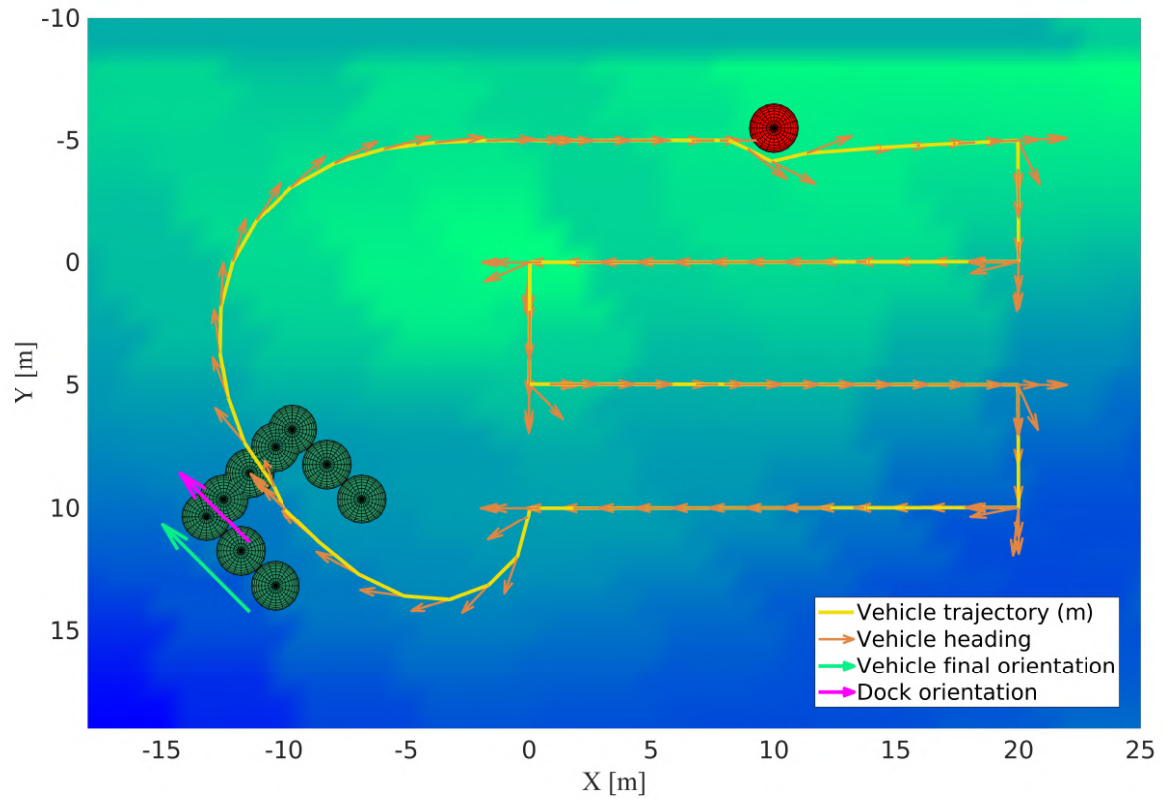


Figure 5.39 Resident AUV mission: Plot showing the mission and docking (top) to a docking station situated close to the seafloor from top view. The red spheres represent the obstacles that the vehicle encounters during the mission and green spheres represent the envelop obstacle around the docking station.

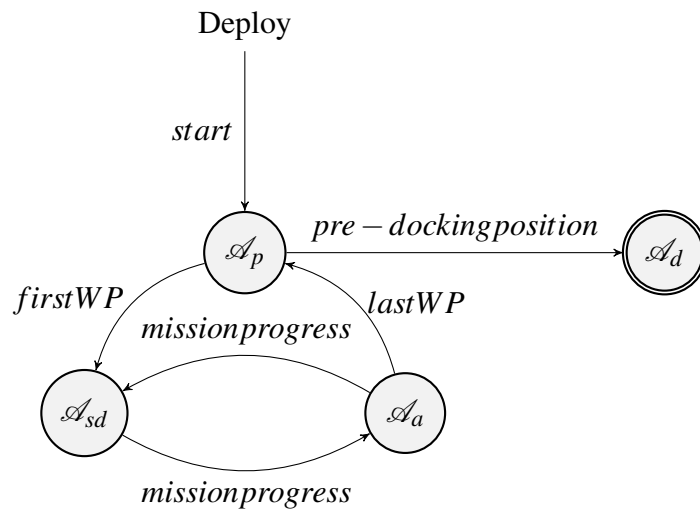


Figure 5.40 State Transition Diagram: Resident AUV survey.

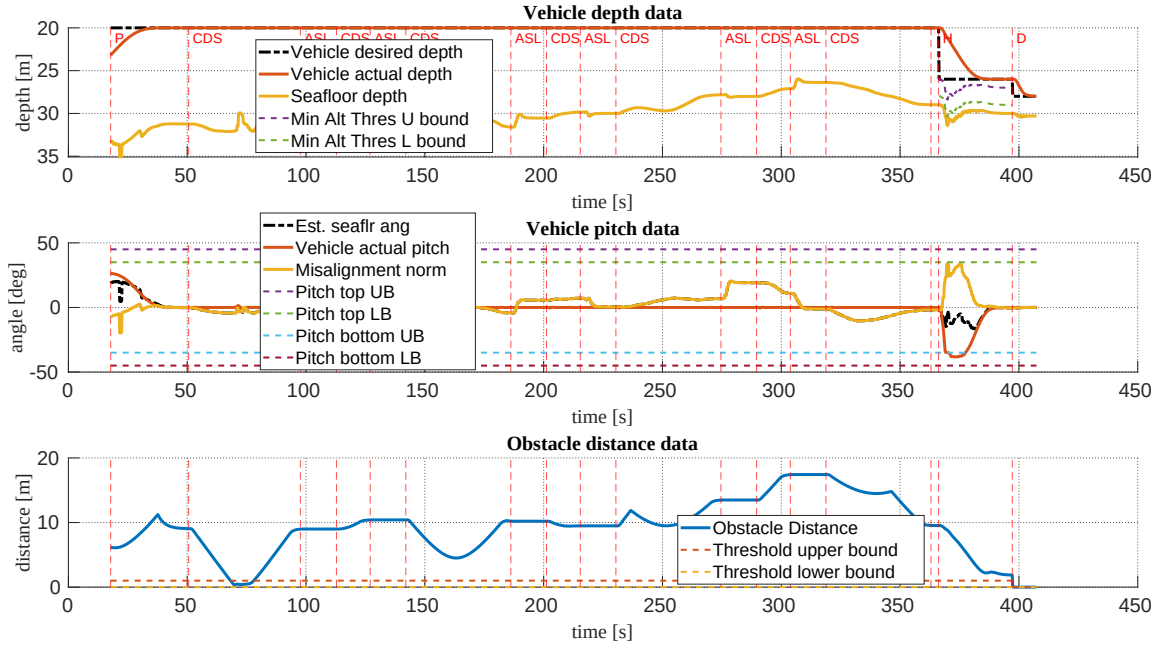


Figure 5.41 Resident AUV mission: Time behaviour of the vehicle depth, pitch and obstacle avoidance data. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.

wide turn while approaching the moving dock as seen in Figures 5.38 and 5.39. The vehicle, after matching its position and orientation with the pre-docking position and orientation using velocity field, eventually translate down to the final dock position on the vehicle ($\mathcal{A}_d$). At every phase, the vehicle reactively avoids any obstacles (represented by red spheres) if present along its path. The spheres depicted around the docking station (represented by the green spheres) is the envelope obstacle to provide a safety area around the docking station while the vehicle approaches the dock during homing. Note that as soon as the vehicle is deployed, since the docking station is a partially enclosed structure, thanks to the *Obstacle Avoidance* task, the the vehicle travels above the docking station, reactively avoiding envelop obstacle present around the docking station, Thus the envelop obstacle can help in guiding the vehicle by pushing it further away from the docking station and around if required, towards the required approach position for a safe docking or during deployment. The state transition diagram for the Resident AUV mission is shown in Figure 5.40.

From Figure 5.41, at around time $t = 51s$, the vehicle changes from the parking maneuver ($\mathcal{A}_p$) to the path following at constant depth survey maneuver ($\mathcal{A}_{sd}$ and $\mathcal{A}_a$). At time $t = 366s$, the vehicle starts the homing maneuver ($\mathcal{A}_p$) and at time $t = 397s$, the docking maneuver

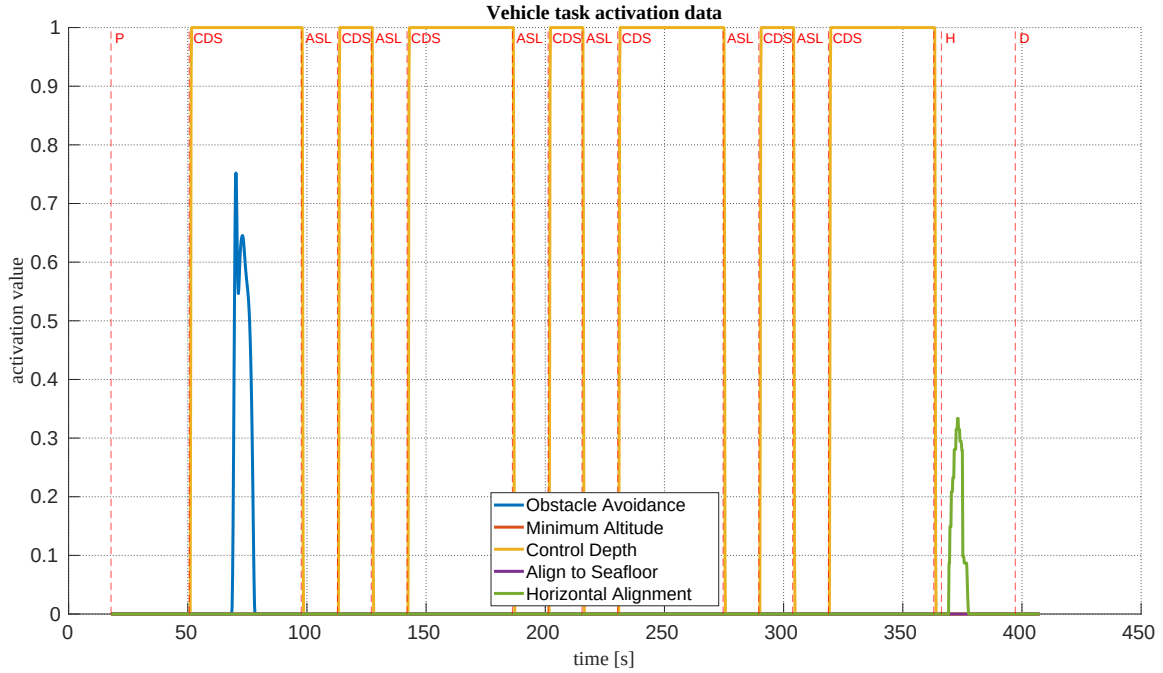


Figure 5.42 Resident AUV mission: Time behaviour of Obstacle Avoidance, Minimum Altitude, Vehicle Depth, Seafloor Alignment and Horizontal Alignment task activation functions. The various phases of the mission are denoted in the plot as follows: P - Parking, CDS - Constant Depth Survey, ASL - Align to Survey Leg, H - Homing, D - Docking.

($\mathcal{A}_d$) is initiated. During the mission, the depth of the vehicle is maintained at the desired level by the *Vehicle Depth* Task. However, the vehicle encounters an obstacle along its path as depicted by sphere of radius 1m at positions [10, -5.5, 20.5]. Obstacle avoidance data from Figure 5.41 shows that the vehicle is able to estimate the distance to the obstacles (only the distance to the nearest obstacle at each instant is plotted) and reactively avoids them along the path. It is also noted that the vehicle is deployed and docked to the same location due to its resident behaviour. Figure 5.42 highlights the time behavior of the tasks that are used in the mission (*Obstacle Avoidance*, *Minimum Altitude*, *Vehicle Depth*) and homing (*Obstacle Avoidance*, *Minimum Altitude*, *Seafloor Alignment*) phases. In this mission, the roll values of the vehicle vary from -4° to 4° , therefore confirming that the roll angles are negligible and acceptable.

Chapter 6

Conclusions

This chapter summarizes and evaluates the results of the previous chapters and the research objectives set out before. It focused on the problems arising when AUVs are required to execute a mission and autonomously home and dock to stationary or moving docking stations. The research showed how to tackle them using a unified framework approach. It presented a general control strategy incorporating various vehicle behaviors into the task priority framework, targeting a system capable of staying and operating underwater, perform autonomous missions, homing, docking, obstacle avoidance, terrain following, minimum altitude and pitch control. The tasks and actions with proper priorities defined to achieve the various vehicle behaviours are presented. In particular, it shows how to integrate the *Field Velocity* Task for long range parking maneuver, allowing the AUV to reach the close proximity of the docking station oriented well in advance and then taking advantage of the *Terminal Dock* Task to successfully complete the short range parking maneuver. Furthermore, it shows how the adoption of the task-priority framework allows to integrate tasks such as *Minimum Altitude*, *Obstacle Avoidance*, *Seafloor Alignment*, *Horizontal Alignment* and exploit them during the execution of the regular survey mission, and also while doing the homing and docking maneuvers. Hence these maneuvers are tackled in more challenging conditions w.r.t. previous works. The contributions have been shown, thanks to different numerical simulations based on the hydrodynamic model of the X300 AUV, demonstrating the successful integrated execution of surveying, homing and terminal docking maneuvers. The results very well prove that the proposed framework is flexible. Hence additional capabilities to the system can be incorporated very easily.

6.1 Summary of main achievements

With respect to the research objectives presented in Chapter 1, the thesis identifies the below achievements:

- Chapter 1 introduced the need of AUVs and its various applications. It also provides a general thesis outline and outlines the main thesis contributions.
- Chapter 2 introduces the general notations used in the thesis and also discusses the mathematical models for the AUV. The kinematic and dynamic model of the AUV was then used in the simulations.
- Chapter 3 gave insight into the task priority framework that this thesis bases on. It also provides information regarding the tasks, task hierarchy, combining multiple tasks to form action, action sequencing using priority of the tasks and structuring the required vehicle behaviour. Last, but not least, it discusses the task priority inverse kinematics.
- Chapter 4 discussed the development of a unified control system based on the flexible task priority framework. It incorporates different kind of tasks, thereby achieving flexibility in the control system itself (tasks). It introduces the various tasks in detail and also gives information on the framework design based on actions. Various examples are also provided where a full mission is modelled based on the required behaviour of the vehicle. This includes the contributions **C1**, **C2** and **C3** which are as follows:
 - C1** Developed a unified control system based on the flexible task priority framework incorporating different kind of tasks, thereby achieving flexibility in the control system itself (tasks).
 - C2** Generated different behaviours to handle various phases of the mission within the same framework, focusing on a general control strategy (action sequencing).
 - C3** Developed various homing and docking configurations that allows the vehicle to dock onto both stationary and moving docking stations situated or moving at seafloor, mid-water or at surface (docking from sides, top, bottom, rear, etc).
- Chapter 5 encompass the simulation results of various tests that were carried out to prove the working and application of the proposed unified framework model based on task priority. It includes free space tests as well as mission based tests in various challenging conditions. This includes the contributions **C4** and **C5** which are as follows:

- C4** Integrated the homing and docking maneuver into the flexible task priority framework for performing missions such as Constant altitude survey, Constant depth survey, Resident AUV survey etc. It also tackles various challenging scenarios such as avoiding obstacles, moving parallel to seafloor, maintaining a minimum altitude, maintain the pitch within safety limits etc., taking into account all the constraints and objectives (Missions).
- C5** Exploited the flexibility allowed by the task priority framework, allowing a holonomic vehicle behave like a non-holonomic vehicle to improve performance by reducing the power consumption and drag (Align task).

With respect to the state of the art, this research have improved the homing and docking approaches as detailed below.

- In [71], if the vehicle starts too close to the structures, there is a chance of collision as the vehicle might perform some tight maneuvers. Here, a similar scenario in Section 5.2.3 is replicated with a moving docking station, where the vehicle starts at close proximity from one side of the docking station and travels to the other side of the docking station to perform docking. The vehicle is able to move away from the close proximity of the docking station by going around due to the *Obstacle Avoidance Task*. The vehicle reaches the pre-docking position defined at the rear side of the docking station, by following a velocity field trajectory.
- In the scenario discussed in Section 5.2.4, the vehicle after being deployed from a docking station, maintains a predefined altitude from the seafloor throughout the mission and follows the terrain of the seafloor at lower altitudes (an improvement over [73]). The vehicle also avoids various obstacles along its path.
- While docking, in comparison to [74], lateral or vertical translations were used (if required) only for the terminal docking maneuver (refer 5.2.4), thereby reducing drag and energy consumption.

As discussed in 5.2.2, based on this study, the research study recommend various docking approaches by taking into account of the constraints imposed on the vehicle as well as the docking station. Table 6.1 summarizes the recommendations.

Docking approach	Vehicle Type	Docking Station Type
Left	Holonomic	Surface, Mid water
Right	Holonomic	Surface, Mid water
Top	Holonomic	Mid water, Sea floor, Partially enclosed
Bottom	Holonomic	Surface, Mid water, Partially enclosed
Rear	Holonomic, Non Holonomic	Surface, Mid water, Partially enclosed

Table 6.1 Table depicting the recommendations for the docking approach based on the vehicle type and docking station type.

6.2 Future Scope

In this work, the research focused on the control and structuring the behaviour of the vehicle. As this work does not deal with the the problems of navigation or estimating the position of the docking station, they are hence out of scope. The availability of sensor modules were assumed and that all simulations are done with perfect feedback. This work could be further improved by relaxing these assumptions set in Section 5.1. Specifically, the uncertainties in navigation could be improved by estimating the states of the vehicle. The docking maneuver could be improved by estimating the position of the docking station using visual markers attached to the docking station. These visual markers can be recognized with the help of an on-board camera on the vehicle. To analyze various behaviours and features in a more realistic scenario, Hardware-In-the-Loop simulator or a real vehicle could be used in the setup. To eliminate the problem of local minima discussed in Section 3.4, a global path planning solution can be devised into our unified unified framework. Lastly, this unified framework architecture is ready to be expanded to manipulation tasks in the future [63, 66].

References

- [1] R. B. Wynn et al. “Autonomous Underwater Vehicles (AUVs): Their past, present and future contributions to the advancement of marine geoscience”. In: *Marine Geology* 352 (2014), pp. 451–468.
- [2] J. Nicholson and A. Healey. “The present state of autonomous underwater vehicle (AUV) applications and technologies”. In: *Marine Technology Society Journal* 42.1 (2008), pp. 44–51.
- [3] S. B. Joye. “Deepwater Horizon, 5 years on”. In: *Science* 349.6248 (2015), pp. 592–593.
- [4] B. Anderson and J. Crowell. *Workhorse AUV—a cost-sensible new autonomous underwater vehicle for surveys/soundings, search & rescue, and research*. IEEE, 2005.
- [5] G. Marani, S. K. Choi, and J. Yuh. “Underwater autonomous manipulation for intervention missions AUVs”. In: *Ocean Engineering* 36.1 (2009), pp. 15–23.
- [6] N. Palomeras et al. “AUV homing and docking for remote operations”. In: *Ocean Engineering* 154 (2018), pp. 106–120.
- [7] Z. Song, A. Marburg, and D. Manalang. “Resident Subsea Robotic Systems: A Review”. In: *Marine Technology Society Journal* 54.5 (2020), pp. 21–31.
- [8] A. Y. Martin. “Unmanned maritime vehicles: Technology evolution and implications”. In: *Marine Technology Society Journal* 47.5 (2013), pp. 72–83.
- [9] T. Newell. “Technical Building Blocks for a Resident Subsea Vehicle”. In: *Offshore Technology Conference*. OnePetro. 2018.
- [10] T. Curtin and D. Manalang. “Resident Autonomous Underwater Vehicles: Docking Revisited”. In: *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*. IEEE. 2018, pp. 1–6.
- [11] D. Manalang et al. “Resident auv workshop 2018: Applications and a path forward”. In: *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*. IEEE. 2018, pp. 1–6.
- [12] T. Jacobs. “Robotic Roustabouts for Tomorrow’s Subsea Fields”. In: *Journal of Petroleum Technology* 66.01 (2014), pp. 50–55.
- [13] B. Gilmour, G. Niccum, and T. O’Donnell. “Field resident AUV systems—Chevron’s long-term goal for AUV development”. In: *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*. IEEE. 2012, pp. 1–5.
- [14] R. Zagatti et al. “FlatFish Resident AUV: leading the autonomy era for subsea oil and gas operations”. In: *Offshore Technology Conference*. OnePetro. 2018.

- [15] T. Matsuda et al. "Resident autonomous underwater vehicle: Underwater system for prolonged and continuous monitoring based at a seafloor station". In: *Robotics and Autonomous Systems* 120 (2019), p. 103231.
- [16] M. Furuholmen et al. "Resident Autonomous Underwater Vehicle Systems—A Review of Drivers, Applications, and Integration Options for the Subsea Oil and Gas Market". In: *Offshore Mediterranean Conference and Exhibition*. OnePetro. 2013.
- [17] E. I. Sarda and M. R. Dhanak. "A USV-Based automated launch and recovery system for AUVs". In: *IEEE journal of oceanic engineering* 42.1 (2016), pp. 37–55.
- [18] S. Ahmad, R. Sutton, and R. Burns. "Retrieval of an autonomous underwater vehicle: An interception approach". In: *Underwater Technology* 25.4 (2003), pp. 185–197.
- [19] T. Hardy and G. Barlow. "Unmanned Underwater Vehicle (UUV) deployment and retrieval considerations for submarines". In: *International Naval Engineering Conference and Exhibition 2008*. 2008.
- [20] M. Renilson. "A simplified concept for recovering a UUV to a submarine". In: *Underwater Technology* 32.3 (2014), pp. 193–197.
- [21] H. Stuart et al. "The ocean one hands: An adaptive design for robust marine manipulation". In: *The International Journal of Robotics Research* 36.2 (2017), pp. 150–166.
- [22] E. Simetti. "Autonomous Underwater Intervention". In: *Current Robotics Reports* (2020), pp. 1–6.
- [23] R. Stokey et al. "A docking system for REMUS, an autonomous underwater vehicle". In: *Oceans' 97. MTS/IEEE Conference Proceedings*. Vol. 2. IEEE. 1997, pp. 1132–1136.
- [24] M. T. Vu et al. "Docking assessment algorithm for autonomous underwater vehicles". In: *Applied Ocean Research* 100 (2020), p. 102180.
- [25] F. Maire et al. "A vision based target detection system for docking of an autonomous underwater vehicle". In: *Proceedings of the 2009 Australasian Conference on Robotics and Automation*. Australian Robotics & Automation Association. 2009, pp. 1–7.
- [26] S. Cowen, S. Briest, and J. Dombrowski. "Underwater docking of autonomous undersea vehicles using optical terminal guidance". In: *Oceans' 97. MTS/IEEE Conference Proceedings*. Vol. 2. IEEE. 1997, pp. 1143–1147.
- [27] P. Trslić et al. "Vision-based localization system suited to resident underwater vehicles". In: *Sensors* 20.2 (2020), p. 529.
- [28] L. Brignone, M. Perrier, and C. Viala. "A fully autonomous docking strategy for intervention AUVs". In: *OCEANS 2007-Europe*. IEEE. 2007, pp. 1–6.
- [29] B. N. J. Vandavasi et al. "Concept and testing of an electromagnetic homing guidance system for autonomous underwater vehicles". In: *Applied Ocean Research* 73 (2018), pp. 149–159.
- [30] C. Thomas, E. Simetti, and G. Casalino. "A Unifying Task Priority Approach for Autonomous Underwater Vehicles Integrating Homing and Docking Maneuvers". In: *Journal of Marine Science and Engineering* 9.2 (2021), p. 162.
- [31] SNAME. "Nomenclature for treating the motion of a submerged body through a fluid". In: *Technical and Research Bulletin No. 1-5 - New York* (Jan. 1950), pp. 1–15.

- [32] T. I. Fossen. "Guidance and control of ocean vehicles". In: *University of Trondheim, Norway, Printed by John Wiley & Sons, Chichester, England, ISBN: 0 471 94113 1, Doctors Thesis* (1999).
- [33] T. I. Fossen and O.-E. Fjellstad. "Nonlinear modelling of marine vehicles in 6 degrees of freedom". In: *Mathematical Modelling of Systems* 1.1 (1995), pp. 17–27.
- [34] T. I. Fossen. *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.
- [35] Y. Nakamura, H. Hanafusa, and T. Yoshikawa. "Task-priority based redundancy control of robot manipulators". In: *The International Journal of Robotics Research* 6.2 (1987), pp. 3–15.
- [36] B. Siciliano and J.-J. Slotine. "A general framework for managing multiple tasks in highly redundant robotic systems". In: *Fifth International Conference on Advanced Robotics' Robots in Unstructured Environments*. IEEE. 1991, pp. 1211–1216.
- [37] J. Lee, N. Mansard, and J. Park. "Intermediate desired value approach for task transition of robots in kinematic control". In: *IEEE Transactions on Robotics* 28.6 (2012), pp. 1260–1277.
- [38] A. Santamaria-Navarro, V. Lippiello, and J. Andrade-Cetto. "Task priority control for aerial manipulation". In: *2014 IEEE International Symposium on Safety, Security, and Rescue Robotics (2014)*. IEEE. 2014, pp. 1–6.
- [39] Y. Hu, B. Huang, and G.-Z. Yang. "Task-priority redundancy resolution for co-operative control under task conflicts and joint constraints". In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2015, pp. 2398–2405.
- [40] L. R. Buonocore, J. Cacace, and V. Lippiello. "Hybrid visual servoing for aerial grasping with hierarchical task-priority control". In: *2015 23rd Mediterranean Conference on Control and Automation (MED)*. IEEE. 2015, pp. 617–623.
- [41] Y. Nakamura and H. Hanafusa. "Inverse kinematic solutions with singularity robustness for robot manipulator control". In: *Journal of Dynamic Systems, Measurement, and Control* (1986).
- [42] O. Khatib. "A unified approach for motion and force control of robot manipulators: The operational space formulation". In: *IEEE Journal on Robotics and Automation* 3.1 (1987), pp. 43–53.
- [43] A. A. Maciejewski and C. A. Klein. "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments". In: *The international journal of robotics research* 4.3 (1985), pp. 109–117.
- [44] T. Yoshikawa. "Analysis and control of robot manipulators with redundancy. Robotic Research". In: *The First International Symposium, 1984* (1984), pp. 735–747.
- [45] Y. Nakamura. *Advanced robotics: redundancy and optimization*. Addison-Wesley Longman Publishing Co., Inc., 1990.
- [46] S. Chiaverini. "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators". In: *IEEE Transactions on Robotics and Automation* 13.3 (1997), pp. 398–410.

- [47] G. Antonelli, F. Arrichiello, and S. Chiaverini. “The null-space-based behavioral control for autonomous robotic systems”. In: *Intelligent Service Robotics* 1.1 (2008), pp. 27–39.
- [48] G. Antonelli and S. Chiaverini. “Task-priority redundancy resolution for underwater vehicle-manipulator systems”. In: *Proceedings. 1998 IEEE international conference on robotics and automation (cat. No. 98CH36146)*. Vol. 1. IEEE. 1998, pp. 768–773.
- [49] G. Antonelli and S. Chiaverini. “Fuzzy redundancy resolution and motion coordination for underwater vehicle-manipulator systems”. In: *IEEE Transactions on Fuzzy Systems* 11.1 (2003), pp. 109–120.
- [50] G. Casalino et al. “A task & subsystem priority based control strategy for underwater floating manipulators”. In: *IFAC Proceedings Volumes* 45.5 (2012), pp. 170–177.
- [51] E. Zereik et al. “Space robotics supporting exploration missions: vision, force control and coordination strategy for crew assistants”. In: *Intelligent Service Robotics* 4.1 (2011), pp. 39–60.
- [52] G. Casalino et al. “Self-organizing control of reconfigurable manipulators: a distributed dynamic programming based approach”. In: *2009 ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots*. IEEE. 2009, pp. 632–640.
- [53] N. Mansard, A. Remazeilles, and F. Chaumette. “Continuity of varying-feature-set control laws”. In: *IEEE Transactions on Automatic Control* 54.11 (2009), pp. 2493–2505.
- [54] N. Mansard, O. Khatib, and A. Kheddar. “A unified approach to integrate unilateral constraints in the stack of tasks”. In: *IEEE Transactions on Robotics* 25.3 (2009), pp. 670–685.
- [55] O. Kanoun, F. Lamiraux, and P.-B. Wieber. “Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task”. In: *IEEE Transactions on Robotics* 27.4 (2011), pp. 785–792.
- [56] W. Decré et al. “Extending iTaSC to support inequality constraints and non-instantaneous task specification”. In: *2009 IEEE International Conference on Robotics and Automation*. IEEE. 2009, pp. 964–971.
- [57] G. Antonelli, S. Moe, and K. Y. Pettersen. “Incorporating set-based control within the singularity-robust multiple task-priority inverse kinematics”. In: *2015 23rd Mediterranean Conference on Control and Automation (MED)*. IEEE. 2015, pp. 1092–1097.
- [58] E. Simetti et al. “Floating underwater manipulation: Developed control methodology and experimental validation within the TRIDENT project”. In: *Journal of Field Robotics* 31.3 (2014), pp. 364–385.
- [59] E. Simetti and G. Casalino. “A novel practical technique to integrate inequality control objectives and task transitions in priority based control”. In: *Journal of Intelligent & Robotic Systems* 84.1 (2016), pp. 877–902.
- [60] E. Simetti et al. “Autonomous underwater intervention: Experimental results of the MARIS project”. In: *IEEE Journal of Oceanic Engineering* 43.3 (2017), pp. 620–639.
- [61] E. Simetti and G. Casalino. “Manipulation and transportation with cooperative underwater vehicle manipulator systems”. In: *IEEE Journal of oceanic engineering* 42.4 (2016), pp. 782–799.

- [62] S. Moe et al. “Set-based tasks within the singularity-robust multiple task-priority inverse kinematics framework: General formulation, stability analysis, and experimental results”. In: *Frontiers in Robotics and AI* 3 (2016), p. 16.
- [63] E. Simetti et al. “Task priority control of underwater intervention systems: Theory and applications”. In: *Ocean Engineering* 164 (2018), pp. 40–54.
- [64] P. Cieřlak et al. “Practical formulation of obstacle avoidance in the Task-Priority framework for use in robotic inspection and intervention scenarios”. In: *Robotics and Autonomous Systems* 124 (2020), p. 103396.
- [65] S. Aldhaheri et al. “Underwater Robot Manipulation: Advances, Challenges and Prospective Ventures”. In: *arXiv preprint arXiv:2201.02954* (2022).
- [66] P. Di Lillo et al. “Underwater intervention with remote supervision via satellite communication: Developed control architecture and experimental results within the dexrov project”. In: *IEEE Transactions on Control Systems Technology* 29.1 (2020), pp. 108–123.
- [67] D. Di Vito et al. “Experimental validation of the modeling and control of a multibody underwater vehicle manipulator system for sea mining exploration”. In: *Journal of Field Robotics* 38.2 (2021), pp. 171–191.
- [68] E. Simetti et al. “Sea Mining Exploration with an UVMS: Experimental Validation of the Control and Perception Framework”. In: *IEEE/ASME Transactions on Mechatronics* (2020).
- [69] E. Simetti et al. “A task priority approach to cooperative mobile manipulation: Theory and experiments”. In: *Robotics and Autonomous Systems* 122 (2019), p. 103287.
- [70] G. Antonelli, G. Indiveri, and S. Chiaverini. “Prioritized closed-loop inverse kinematic algorithms for redundant robotic systems with velocity saturations”. In: *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2009, pp. 5892–5897.
- [71] J. Woo and N. Kim. “Vector field based guidance method for docking of an unmanned surface vehicle”. In: *The Twelfth ISOPE Pacific/Asia Offshore Mechanics Symposium*. OnePetro. 2016.
- [72] P. Sujit, A. Healey, and J. B. Sousa. “AUV docking on a moving submarine using a KR navigation function”. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2011, pp. 3154–3159.
- [73] Z. Yan et al. “Modeling, strategy and control of UUV for autonomous underwater docking recovery to moving platform”. In: *2017 36th Chinese Control Conference (CCC)*. IEEE. 2017, pp. 4807–4812.
- [74] M. Breivik and J.-E. Loberg. “A virtual target-based underway docking procedure for unmanned surface vehicles”. In: *IFAC Proceedings Volumes* 44.1 (2011), pp. 13630–13635.
- [75] P. Encarnacao, A. Pascoal, and M. Arcak. “Path following for autonomous marine craft”. In: *IFAC Proceedings Volumes* 33.21 (2000), pp. 117–122.
- [76] P. Encarnacao and A. Pascoal. “3D path following for autonomous underwater vehicle”. In: *Proceedings of the 39th IEEE conference on decision and control (cat. No. 00CH37187)*. Vol. 3. IEEE. 2000, pp. 2977–2982.

- [77] E. Peymani and T. I. Fossen. “2d path following for marine craft: A least-square approach”. In: *IFAC Proceedings Volumes* 46.23 (2013), pp. 98–103.
- [78] X. Xu et al. “A novel non-collision trajectory planning algorithm based on velocity potential field for robotic manipulator”. In: *International Journal of Advanced Robotic Systems* 15.4 (2018), p. 1729881418787075.
- [79] A. M. Lekkas and T. I. Fossen. “Line-of-sight guidance for path following of marine vehicles”. In: *Advanced in marine robotics* (2013), pp. 63–92.
- [80] M. Aicardi et al. “Guidance of 3D underwater non-holonomic vehicle via projection on holonomic solutions”. In: *Symposium on Underwater Robotic Technology, SURT 2000*. 2000.
- [81] M. Aicardi, G. Casalino, and G. Indiveri. “Closed loop control of 3D underactuated vehicles via velocity field tracking”. In: *2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics. Proceedings (Cat. No. 01TH8556)*. Vol. 1. IEEE. 2001, pp. 355–360.
- [82] *Mini-Ranger 2 USBL - Sonardyne*. [Online; accessed 7. Apr. 2022]. URL: <https://www.sonardyne.com/products/underwater-positioning-mini-ranger>.
- [83] *Subsonus | Advanced Navigation*. [Online; accessed 7. Apr. 2022]. URL: <https://www.advancednavigation.com/solutions/subsonus>.
- [84] *Matrox Meteor-II*. [Online; accessed 7. Apr. 2022]. URL: <https://www.matrox.com/sites/default/files/matrox-meteor-ii.pdf>.
- [85] *SPRINT-Nav Mini - Sonardyne*. [Online; accessed 7. Apr. 2022]. URL: <https://www.sonardyne.com/products/sprint-nav-mini-hybrid-acoustic-inertial-navigator>.
- [86] *ISA500 - AUV & ROV Altimeter, Echosounder & Underwater Distance Measurement*. [Online; accessed 7. Apr. 2022]. URL: <https://www.impactsubsea.co.uk/isa500>.
- [87] L. Cabrera-Brito et al. “Fractal analysis of deep ocean current speed time series”. In: *Journal of Atmospheric and Oceanic Technology* 34.4 (2017), pp. 817–827.
- [88] P. C. Liu and G. S. Miller. “Wavelet transforms and ocean current data analysis”. In: *Journal of atmospheric and oceanic technology* 13.5 (1996), pp. 1090–1099.
- [89] R. E. Thomson and D. M. Ware. “A current velocity index of ocean variability”. In: *Journal of Geophysical Research: Oceans* 101.C6 (1996), pp. 14297–14310.