Task-Space Dynamic Control of Underwater Robots

Zool Hilmi Ismail

Thesis submitted for the
Degree of Doctor of Philosophy

Heriot-Watt University

Department of Electrical, Electronic

and Computer Engineering

January 2011

Abstract

This thesis is concerned with the control aspects for underwater tasks performed by marine robots. The mathematical models of an underwater vehicle and an underwater vehicle with an onboard manipulator are discussed together with their associated properties.

The task-space regulation problem for an underwater vehicle is addressed where the desired target is commonly specified as a point. A new control technique is proposed where the multiple targets are defined as sub-regions. A fuzzy technique is used to handle these multiple sub-region criteria effectively. Due to the unknown gravitational and buoyancy forces, an adaptive term is adopted in the proposed controller.

An extension to a region boundary-based control law is then proposed for an underwater vehicle to illustrate the flexibility of the region reaching concept. In this novel controller, a desired target is defined as a boundary instead of a point or region. For a mapping of the uncertain restoring forces, a least-squares estimation algorithm and the inverse Jacobian matrix are utilised in the adaptive control law.

To realise a new tracking control concept for a kinematically redundant robot, sub-region tracking control schemes with a sub-tasks objective are developed for a UVMS. In this concept, the desired objective is specified as a moving sub-region instead of a trajectory. In addition, due to the system being kinematically redundant, the controller also enables the use of self-motion of the system to perform sub-tasks (drag minimisation, obstacle avoidance, manipulability and avoidance of mechanical joint limits).

Acknowledgements

First and foremost, I would like to thank the Ministry of Higher Education Malaysia and University Teknologi Malaysia for financial support during my PhD study in Edinburgh.

I am heartily thankful to my supervisor, Dr. Matthew W. Dunnigan, whose encouragement, guidance and advice enabled me to develop an understanding of the subject and have provided a good basis for the present thesis.

A bunch of thanks is also given to my colleagues and members of the Intelligent Robotic Systems Laboratory. In particular, I wish my sincere thanks to Dr. Bach H. Dinh for all his guidance and help throughout my PhD.

I owe my loving thanks to my parents and all members of my family. A special thanks to my wife, Dr. Noor Baity Saidi and my cute adorable son, Asyraf Fikry for the happiness and joy in my life.

Table of Contents

Chapter 6: Conclusions

Appendix A: Mathematical Preliminaries

Appendix B: Proof of Theorems

Appendix C: ODIN Model Parameters and Simulation Files

List of Tables

List of Figures

List of Abbreviations

DOF	Degree of Freedom
FIS	Fuzzy Inference System
AUV	Autonomous Underwater Vehicle
multi-AUVs	multiple Autonomous Underwater Vehicles
ROV	Remotely Operated Vehicle
UUV	Unmanned Underwater Vehicle
UVMS	Underwater Vehicle-Manipulator System
MOERI	Maritime and Ocean Engineering Research Institute
RMS	Root Mean Square
VSC	Variable Structure Control
SISO	Single-Input Single-Output
PD	Proportional-Derivative
PID	Proportional-Integral-Derivative
MRAC	Model Reference Adaptive Controller
NNs	Neural Networks
cf.	see
etc.	and the rest
i.e.	that is

List of Notations

$\forall$	for all
$\exists$	there exists
$\in$	belong(s) to
$\Longrightarrow$	implies
$\Leftrightarrow$	is equivalent to, if and only if
$\rightarrow$	tends to, maps onto
$\mathbb{R}$	Real numbers
$\mathbb{R}_+$	non-negative real number
$\boldsymbol{x} \in \mathbb{R}^n$	real vector space of dimension n
$A \in \mathbb{R}^{n \times m}$	set of real matrices of dimension $n \times m$
$A \in \mathbb{R}^{n \times n}$	square matrix with dimension $n \times n$
$A = \{a_{ij}\} \in \mathbb{R}^{n \times n}$	diagonal matrix with $a_{ij} = 0$ for all $i \neq j$
$\dot{\boldsymbol{x}}$	derivative of $\boldsymbol{x}$ with respect to time
$\boldsymbol{x}_d$	desired value of the variable $\boldsymbol{x}$
$\|\boldsymbol{x}\|$	Euclidean norm of vector $\boldsymbol{x}$
$\boldsymbol{x}_i$	ith element of the vector $\boldsymbol{x}$
$\boldsymbol{x}^T (X^T)$	transpose of the vector $\boldsymbol{x}$ (matrix X)
$\tilde{\boldsymbol{e}}$	error variable
$\boldsymbol{x}_d$	desired value of the variable $\boldsymbol{x}$
$f: X \rightarrow Y$	the function f maps the set X into the set Y
$\sum_v - X_v Y_v Z_v$	body-fixed reference frame
$\sum_i - X_i Y_i Z_i$	earth-fixed or inertial reference frame
t	time (s)
$\lambda_{min}\{A\} (\lambda_{max}\{A\})$	smallest (largest) eigenvalues of matrix A
$\boldsymbol{\eta}_1 \in \mathbb{R}^3$	position vector in the inertial reference frame (m)
$\boldsymbol{v}_1 \in \mathbb{R}^3$	linear velocity vector of the body-fixed frame with respect to the origin of the inertial reference frame expressed in the body-fixed frame (m s^{-1})
$\boldsymbol{\eta}_2 \in \mathbb{R}^3$	vector of body Euler angle coordinates in the inertial reference frame (rad)

$\boldsymbol{v}_2 \in \mathbb{R}^3$	angular velocity vector of the body-fixed frame with respect to the origin of the inertial reference frame expressed in the body-fixed frame (rad s^{-1})
$\boldsymbol{\epsilon} \in \mathbb{R}^4$	unit quaternion representation
$\boldsymbol{\tau}_{RB} \in \mathbb{R}^6$	resultant external forces (N) and moments (N m) of a rigid body
$\boldsymbol{\zeta}_{v,m} \in \mathbb{R}^7$	end-effector composition vector
$V(t, \boldsymbol{x})$	Lyapunov function candidate
ρ	water density (kg m^{-3})
C_d	dimensionless drag coefficient
R_n	Reynolds number
m	mass of underwater robot (kg)
∇	volume of fluid displaced by the robot (m^3)
$I_{n\times n}$	identity matrix of dimension *n*
$0_{n\times n}$	null matrix of dimension *n*

CHAPTER 1
INTRODUCTION

1.1. Introduction

Robots have been developed to help humans in their everyday lives in several ways including increasing productivity in manufacturing areas or for safety reasons. In hazardous environments, robots are recognised as being effective and remove the need for direct human intervention. Robots can be utilised to perform tasks that would otherwise be too dangerous for humans, for example in the space, nuclear or subsea domains. It is this latter application area that has provided the motivation for the work in this thesis.

Generally, robots used for increasing productivity in the manufacturing industries exhibit little or no intelligence. They are taught exactly how to perform a task and repeat that sequence of commands a fixed number of times. Thus, in manufacturing applications, robotic work cells are designed to be structured to eliminate the occurrence of unexpected events which the robot cannot cope with. Conversely, robots that operate in hazardous environments cannot be pre-programmed as the workspace is often unstructured and subject to changes. As a result, the robot must be able to react safely to any unpredictable incidents.

In order to fulfil the requirements for operation in hazardous environments, robots are currently teleoperated where a human operator controls every aspect of the robot from a remote location [1.1]. Onboard cameras mounted at the remote worksite, or on the robot itself, allow the operator to see the surrounding area of the robot, allowing the required task to be carried out effectively. Movements of the robot are achieved using a master-slave arrangement, where the remote slave robot follows any motions that the operator commands with a suitable master input device. There are several problems associated with teleoperation such as the information delay between the robot and operator sites, objects of interest are hidden from view in certain cases and operators with great expertise are required to perform specific tasks.

For these reasons, the need for advanced underwater robot technologies has rapidly expanded in recent years which will eventually lead to underwater robotic vehicles being fully autonomous, specialised and reliable. Various studies have improved the vehicle autonomy and have reduced the demand for human operator intervention. A good review of past and future autonomous underwater robotics can be found in [1.2]. In this review, it was also stated that the aim of current research in this field is to develop self-contained, intelligent and decision-making autonomous underwater vehicles.

1.2. Underwater Vehicles with On-board Mechanical Manipulators

For several decades, extensive efforts have been devoted to the development of unmanned underwater robots, i.e. Remotely Operated Vehicles (ROVs) and Autonomous Underwater Vehicles (AUVs), with the purpose of overcoming the challenging engineering problems caused by the unstructured and hazardous ocean environment. University and research laboratories became interested in these vehicles and a complete survey of the research area was given by Yuh in [1.2, 1.3]. ROVs are underwater robots that are tethered to a surface vessel and are controlled from onboard the vessel. They have become an important tool for undersea intervention to collect samples, video images and data, as well as to perform experiments and install and maintain underwater production equipment.

AUVs are undersea robots that are not tethered to a surface vessel. An AUV is programmed at the surface and then navigates through the water collecting data under the control of its own internal computer system. When an AUV's mission is completed, it returns to the surface (or to an underwater docking station) where its data can be retrieved. Most AUVs are survey vehicles not equipped with manipulators that perform various undersea tasks ranging from environmental monitoring to scientific and military operations.

Underwater intervention missions often require mechanical manipulators which are typically mounted on the underwater robot. In this case, the system is known as an Underwater Vehicle-Manipulator System (UVMS). It is quite difficult and tiring for a human operator to operate these manipulators due to the fact the vehicle is not stationary. Therefore, the control algorithms are different from those used for stationary

industrial manipulators in factories. In general, more than one type of manipulator end-effector may be needed for multi-task operations. To change the end-effector with current underwater manipulators, the vehicle must be brought to the surface and the end-effector changed appropriately for each task. This procedure is undesirable as it is time-consuming and expensive. A flexible and dexterous design of an end-effector was developed to carry out a variety of sophisticated operations [1.4].

Most of these underwater arms have been designed for remotely operated vehicles and they are actuated by hydraulic pistons. They are also designed for teleoperation from onboard a surface ship and a tether is used to pass power, telemetry and video images. The underwater manipulators used are almost exclusively hydraulically actuated, i.e. the TITAN 4 and ORION series developed by Schilling Robotics Systems, due to their mechanical robustness and large power to weight ratio. In the case of autonomous underwater vehicles where battery power is limited, manipulators with electric drives, i.e. the Telemanipulator designed by Tecnomare, that consume less energy are preferred. Currently, some research institutions such as the University of Hawaii [1.5] and the Maritime and Ocean Engineering Research Institute (MOERI) [1.6] are developing a semi-AUV, SAUVIM, equipped with an electro-mechanical underwater manipulator.

1.3. Underwater Robot Control Tasks

In underwater robot control tasks, there are two classes for the control objectives, namely trajectory tracking and regulation control. Note that, throughout the remainder of this thesis the term underwater robots refer to any re-programmable, multi-functional unmanned vehicles such as Remotely Operated Vehicles and Autonomous Underwater Vehicles and unmanned vehicles with onboard manipulators, i.e. Underwater Vehicle-Manipulator Systems.

Trajectory tracking can be viewed as a time-varying reference trajectory specified within the joint space or task-space. In general, the robot velocities and accelerations associated with the desired trajectory should not violate the velocity and acceleration limits of the robot. Regulation control, which is also known as point-to-point control aims to move and keep the variable at the desired position in spite of external disturbances and should be independent of the initial conditions. The transient

behaviour is in general not guaranteed. In this thesis, this is referred to as position control. The task specifications of the underwater robot in terms of both the desired trajectory tracking and set-point regulation are carried out in the task-space and gives rise to operational space control, which is introduced in Chapter 2.

The fundamental components in an underwater robotic system are shown schematically in Figure 1.1. The trajectory planning function generates the reference inputs for the motion control system which ensures that the underwater robot executes the planned trajectories. Typically, a number of parameters are determined to describe the desired trajectory. Planning consists of generating a time sequence of the values from a polynomial function used for interpolating the desired trajectory. The inputs to the trajectory planning algorithm are the path description, the path constraints and the constraints imposed by the robot dynamics. The outputs, which are commonly called the reference trajectories, are given primarily in terms of coordinates in the workspace. It should be noted that the path only represents the locus of points in the operational space, relating the initial and final desired posture of the robot. Subsequently, the inverse kinematic function is used to transform a time-parameterised trajectory from the operational space to the vehicle space.

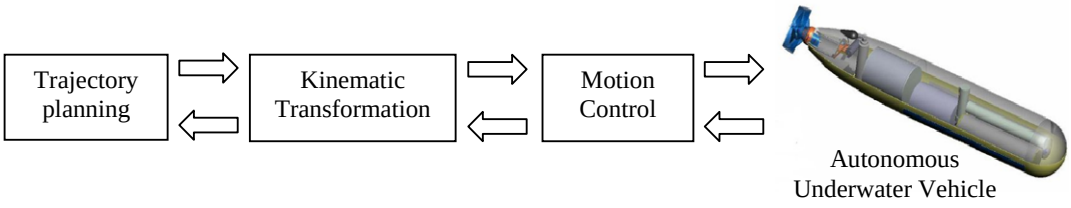


Figure 1.1: General components in the underwater robotic system

The dynamic control function realises the required robot motions to achieve the specified actions and so interfaces directly with the robot. The controlled variables can be position, velocities or interaction forces, the latter being used to perform tasks such as assembly operations. Generally, dynamic control can be categorised into two classes; operational space control schemes and joint space control schemes. In both schemes, the control structure is closed-loop to exploit the good features provided by feedback, i.e. robustness to modelling uncertainties. For a subsea robot, dynamic control is essential to provide accurate control under unknown and changing conditions i.e. unknown buoyancy and gravity forces, which enables the robot to automatically perform complex

tasks. Several useful references including Fossen [1.7] and Antonelli [1.8] have discussed a wide range of dynamic control techniques for underwater robotic systems. In addition, Antonelli reported various coordinated control laws that have been applied for multiple Autonomous Underwater Vehicles (multi-AUVs) in [1.8] including the decentralised and Lyapunov-based control technique. Recently, a distributed model predictive control was proposed by Balderud et al. [1.9] for a cluster of underwater vehicles in a towing application.

The main focus of this research is to introduce new motion controllers for typical subsea robots in the context of task-space operations and the analysis of their inherent stability in the Lyapunov sense. The path planning and trajectory generation problems are not addressed here. It should also be noted that this thesis is concerned with the operation of an underwater vehicle and its onboard manipulator. However, the problems of coupling control between the vehicle and manipulator have been previously addressed in [1.10] and [1.11]. Although the efforts presented in this thesis are focused on this specific application area, the conclusions drawn could be applied to many other robotic systems. Moreover, some control techniques such as passivity-based control, learning control and neural network based control are not covered in this thesis.

1.4. Thesis Organisation

The thesis begins with an introduction to the field of underwater robot control schemes followed by the fundamental background of Lyapunov stability theory. The mathematical formulation of underwater robots with novel regulation and tracking control schemes are subsequently presented. This also includes the stability analysis of each proposed controller. Simulation results are performed to validate the proposed control techniques. A more detailed description of the individual chapters in this thesis is presented below.

Chapter Two presents an introduction to robot control research, discussing the various approaches that have been proposed in the context of underwater applications. The concepts of inertial-fixed frame-based and body-fixed frame-based control of an underwater vehicle are briefly reviewed. In addition, the concepts of set-point regulation and trajectory tracking control are discussed along with the various control methods that are suitable to fulfil the control objectives.

The mathematical model of the underwater vehicle used in this work is presented in Chapter Three. The model covers both kinematic and dynamic aspects of the underwater vehicle, as well as the detailed mathematical analysis of the manipulator that is mounted on this particular vehicle. The related dynamic properties of the models are also presented in this chapter. These models provide an insight into the operation of the underwater robot and are used during the simulation phases.

In Chapter Four, novel control designs for the task-space regulation control problem for underwater robots are presented. These proposed control laws are then extended for the underwater vehicle with an onboard manipulator. The applicability of the proposed methods is illustrated through various simulations.

The development of tracking control laws for kinematically redundant underwater robots is presented in Chapter Five. Initially, the tracking control problem is briefly discussed followed by the proposed new trajectory tracking controllers. Several inverse kinematic solutions act as sub-task objectives for the redundant system, i.e. the joint limit constraints and manipulability. Again, simulation studies are carried out to verify the effectiveness of these controllers.

The final chapter of this thesis, Chapter Six, summarises the work presented and draws relevant conclusions. The suggestions for areas of future work are also discussed in this chapter.

1.5. Author's Contributions

The main contributions of this thesis are addressed in Chapters 4 and 5. The introduction of new task-space regulation control approaches are briefly discussed in Chapter 4. Several novel controllers are proposed that achieve global stability for autonomous underwater vehicle and underwater-vehicle manipulator systems. In particular, Theorem 4.1, Theorem 4.2, Theorem 4.3 and Theorem 4.4 are the main contribution of Chapter 4. The applicability of the presented methods is illustrated by means of simulations. This chapter is a composition of the papers:

- Z. H. Ismail and M. W. Dunnigan, “A Region Boundary-Based Control Scheme for an Autonomous Underwater Vehicle,” 2010, Submitted to *Ocean Engineering*.

- Z. H. Ismail and M. W. Dunnigan, “A Sub-Region Priority Reaching Control Scheme with a Fuzzy-Logic Algorithm for an Underwater Vehicle subject to Uncertain Restoring Forces,” in *Proceedings of Oceans '10 IEEE Sydney*, Sydney, Australia, May 24-27, 2010, pp. 1-9.

- Z. H. Ismail and M. W. Dunnigan, “An Adaptive Region Boundary-Based Control Scheme for an Autonomous Underwater Vehicle,” *11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010*, Singapore, 7-10 December 2010.

- Z. H. Ismail and M. W. Dunnigan, “A Sub-Region Boundary-Based Control Scheme with a Least-Squares Estimation Algorithm for an Underwater Robotic System,” *11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010*, Singapore, 7-10 December 2010.

- Z. H. Ismail and M. W. Dunnigan, “Geometric Formation-Based Region Boundary Control Scheme for Multiple Autonomous Underwater Vehicles,” To be submitted to 2011 IEEE Conference.

- Z. H. Ismail and M. W. Dunnigan, “Fuzzy-Based Sub-Region Priority Control Scheme of an Autonomous Underwater Vehicle with Least-Square Estimation Algorithm,” To be submitted to *International Journal of Control, Automation, and Systems*.

The main contribution of Chapter 5 is the introduction of new task-space tracking controllers for a kinematically redundant underwater robot, as described in Theorems 5.1 and 5.2. The proposed tracking controllers also enable the use of self-motion of the system to perform sub-tasks. This chapter is a composition of the papers:

- Z. H. Ismail and M. W. Dunnigan, “Tracking Control Scheme for an Underwater Vehicle-Manipulator System with Single and Multiple Sub-Regions and Sub-Task Objectives,” 2010, Accepted in *IET Control Theory & Applications*.

- Z. H. Ismail and M. W. Dunnigan, “Redundancy Resolution for Underwater Vehicle-Manipulator Systems with Congruent Gravity and Buoyancy Loading Optimization,” in *Proc. of the IEEE Int. Conference on Robotics and Biomimetic*, Guilin, China, 2009, pp. 1393-1399.

- Z. H. Ismail and M. W. Dunnigan, “A Sub-Region Tracking Control for an Underwater Vehicle-Manipulator System with a Sub-Task Objective,” in *Proc. of Oceans '10 IEEE Sydney*, Sydney, Australia, May 24-27, 2010, pp. 1-8.

- Z. H. Ismail and M. W. Dunnigan, “An Adaptive Robust Tracking Control Scheme for an Underwater Vehicle-Manipulator System Subject To Multiple Sub-Regions and Sub-Tasks Objectives,” *8th IEEE International Conference on Control & Automation*, ICCA’10, Xiamen, China, 9-11 June 2010.

- Z. H. Ismail and M. W. Dunnigan, “Adaptive Robust Tracking Control of an Underwater Vehicle-Manipulator System with Sub-Region and Self-Motion Criteria” 2010, Submitted to special issue on Networked Control and Unmanned Systems, *International Journal of Control and Intelligent Systems*.

1.6. References

[1.1] A. K. Bejczy, "Virtual Reality in Robotics," in *IEEE Conf. on Emerging Technologies and Factory Automation*, vol. 1, New York, USA, 1996, pp. 7-15.

[1.2] J. Yuh and M. West, "Underwater Robotics," *International Journal of Advanced Robotics*, vol. 15, no. 5, pp. 609–639, 2001.

[1.3] J. Yuh, "Design and Control of Autonomous Underwater Robots: A Survey," *Autonomous Robots*, vol. 8, no. 1, pp. 7-24, 2000.

[1.4] D. M. Lane, "Subsea Robotics for the Offshore Industry," in *IEEE Workshop on Robotic Technologies in Oceanic Engineering*, 1995.

[1.5] T. W. Kim and J. Yuh, "Development of a Real-Time Control Architecture for a Semi-Autonomous Underwater Vehicle for Intervention Missions," *Control Engineering Practice*, vol. 12, no. 12, pp. 1521-1530, 2004.

[1.6] J.-H. Li, B.-H. Jun, P.-M. Lee and S.-W. Hong, "A Hierarchical Real-Time Control Architecture for a Semi-Autonomous Underwater Vehicle," *Ocean Engineering*, vol. 32, no. 13, pp. 1631-1641, 2005.

[1.7] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, 1st ed. New York: John Wiley and Sons, 1994.

[1.8] G. Antonelli, *Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems*. Germany: Springer-Verlag Berlin, 2003.

[1.9] J. Balderud, L. Giovanini and M. R. Katebi , "Distributed Control of Underwater Vehicles," *Journal of Engineering for the Maritime Environment*, vol. 222, no. 2, pp. 95-107, 2008.

[1.10] M. W. Dunnigan, *An Investigation of the Dynamic Coupling between a Manipulator and an Underwater Vehicle*. Ph.D Thesis, Heriot-Watt University, 1994.

[1.11] M. W. Dunnigan and G. T. Russell, "Evaluation and Reduction of the Dynamic Coupling Between a Manipulator and an Underwater Vehicle," *IEEE Journal of Oceanic Engineering* , vol. 23, no. 3, pp. 260-273, 1998.

CHAPTER 2
REVIEW OF UNDERWATER ROBOT CONTROL STRATEGIES

2.1. Introduction

The need for accurate control of an underwater robot was addressed in Chapter 1. Some of the factors that lead to inaccurate control are the highly non-linear and time-varying dynamic behaviour of the robot; uncertainties in the hydrodynamic coefficients; the resultant high-order and redundant structure when a robot arm is attached; ocean current disturbances; and changes in the centres of the gravity and buoyancy of the system due to the arm motion which also disturbs the robot's main body. It is difficult to accurately tune the control gains during operation in water. Therefore, it is highly desirable to have a robot control system that has a self-tuning ability when the control performance degrades during operation due to changes in the dynamics of the robot and its environment.

There are various advanced control techniques that have been proposed over the past decades including sliding control, robust control, adaptive control, neural network control and fuzzy control which can accommodate the wide variations in underwater robot dynamics. However, there is still room to improve the performance of the robot's motions when subject to unknown and changing operating conditions. The motivation behind this thesis is inspired from the application of advanced task-space control schemes to improve the accuracy of these underwater robots.

This chapter starts by describing the problems associated with underwater robot control. In accordance with the adopted definition of the underwater robot's output, the control objectives related to regulation and trajectory tracking are also presented in this chapter. In particular, when the robot's output corresponds to the position and velocity, the control objectives can be referred to as "regulation control" and "tracking control" respectively. The different control schemes that have been used are then reviewed, first from the perspective of the required action of the controller, and secondly looking at the various control techniques available.

2.2. Underwater Robot Control Problems

The "point-to-point" method is recognised as the simplest way to specify the movement of a robot. Within this methodology, a series of points in the robot's workspace, which the robot is required to go through is determined beforehand. Thus, the position control problem consists in making the robot move to a specified point regardless of the trajectory followed from its initial configuration as depicted in Figure 2.1.

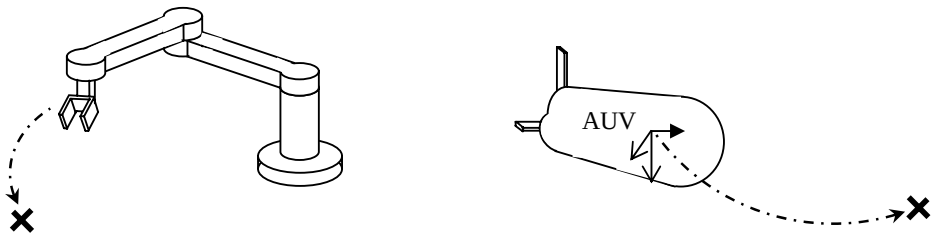


Figure 2.1: Point-to-point motion specification: (a) fixed-based manipulator, (b) underwater vehicle

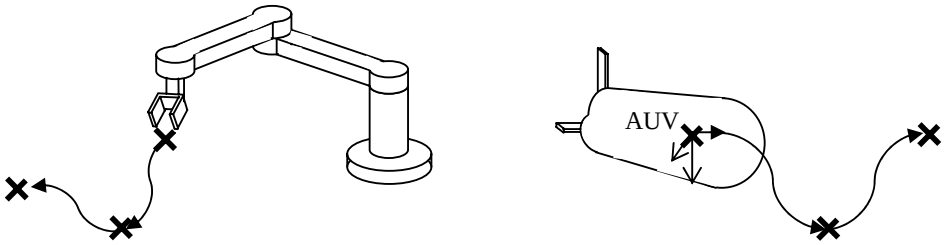


Figure 2.2: Trajectory motion specification: (a) fixed-based manipulator, (b) underwater vehicle

A more general way to specify a robot's motion is via a trajectory. In this case, a continuous path, i.e. a finite sequence of points is assigned along the path, in the state-space and parameterised in time, to achieve a desired task. Therefore, the motion control problem is achieved by ensuring the robot follows the prescribed trajectory as closely as possible. This control problem, known as trajectory tracking control, is shown in Figure 2.2.

For underwater robots, a large number of publications have been devoted to the development of advanced point-to-point control [2.1]-[2.4] and tracking control

techniques [2.5]-[2.7]. Robots used in offshore applications generally sacrifice performance and accuracy for mechanical robustness and they are crude when compared to typical “in-air” robots [2.8]. This is due to the fact that there are many difficult challenges posed for these systems, for instance, the dynamic models of underwater robots possess parameters which are highly dependent on physical quantities such as the hydrodynamic coefficients. These coefficients are typically unknown due to temperature, depth and salinity variations, which leads to poorly known values for these parameters.

2.3. Underwater Robot Control Schemes

Underwater robots can be categorised into two classes: the first class are robots which move independently in their task-space, i.e. the physical workspace is within reach of the end-effector thereby allowing movements without any contact with the environment. Welding tasks and pipeline monitoring may be performed by this type of robot. On the other hand, a robot that is capable of interacting with its environment, for instance, by applying a desired force or carrying a payload for precision assembling is included in this other class. Note that, this thesis exclusively presents control schemes for underwater robots that navigate independently in their task-space.

The underwater tasks are usually specified in the task-space in terms of a desired trajectory of the end-effector, whilst control actions are performed in the joint space to achieve the desired goals. For a floating-base robotic system such as an autonomous underwater vehicle with no serial chain manipulator attached to the vehicle, the control law is expressed either in inertial-fixed (task-space) or body-fixed reference frame. An illustration of inertial-fixed and body-fixed coordinate system representation is presented in Chapter 3. Therefore, it naturally leads to several kinds of general control methods, the so-called task-space control, joint space control and body-fixed frame control approaches. The next section discusses these types of control approaches in more detail.

2.3.1 Task-Space Control Schemes

In many robotic applications, the desired task is naturally defined in terms of end-effector motion. As a result, the desired robot trajectory is described by the desired

position and orientation of a Cartesian coordinate frame attached to the robot manipulator's end-effector with respect to the base frame, also referred to as the task-space. Besides, for robots that operate in more complex and less certain environments such as the undersea domain, their motion may be subject to online modifications in order to accommodate unexpected events or to respond to sensor inputs. There are a variety of underwater tasks where these types of control problems arise. For instance, controlling the interaction between the robot and object or specifying the desired target using an onboard camera. Here, joint space control schemes are unsuitable due to the given task being usually specified in the task-space and accurate control of the end-effector motion is highly desirable. Joint-based control has the undesirable feature of requiring the solution of the inverse kinematics to convert the desired task-space trajectory into the desired joint space trajectory. In contrast, task-space control does not require the inverse kinematics. This thesis has been motivated from this type of control approach where the control schemes are directly developed based on the dynamics expressed in the task-space.

Several parameterisations exist to describe the orientation angles, including minimum three-parameter representations (e.g., Euler angles, Rodrigues parameters, etc.) and the non-minimum four-parameter representation given by the unit quaternion. Meanwhile, the three-parameter representations always exhibit singular orientations, i.e. the orientation Jacobian matrix in the kinematic equation is singular for some orientations. Thus, the unit quaternion-based approach can be used to represent the end-effector orientation without singularities. Despite significantly complicating the control design, the unit quaternion seems to be the preferred method of formulating the end-effector orientation tracking control problem.

Some past works that deal with task-space control formulation for robot manipulators can be found in [2.9], [2.10], and [2.11]. Specifically, an experimental assessment of different end-effector orientation parameterisation for task-space robot control was provided in [2.9]. One of the first results in task-space control of robot manipulators was presented in [2.10]. Resolved-rate and resolved-acceleration task-space controllers using the quaternion parameterisation were proposed in [2.11].

Inspired from the work for a robot manipulator by Slotine and Li [2.12], an extension of task-space control scheme for a 6-DOF autonomous underwater vehicle was proposed

by Fjellstad and Fossen in [2.13] that utilised a 4-parameter unit quaternion to reach a singularity-free representation of the attitude. Yuh in [2.14] presented experimental results for a 6-DOF underwater vehicle using an adaptive task-space controller with the presence of unmodelled dynamics, sensor noise and environmental disturbances.

The objective of the task-space control is to design a feedback controller that allows execution of robot motion that tracks the desired trajectory in Cartesian space as closely as possible. The schematic diagram of task-space control methods are depicted in Figure 2.3. There are several advantages to such an approach because task-space controllers employ a feedback loop that directly minimises task errors. In other words, the computation of the inverse kinematics need not be calculated explicitly, since the control algorithm embeds the velocity-level forward kinematics. Thus, motion between points can be a straight-line segment in the task-space. Based on this concept, several novel task-space control laws are introduced in this thesis, which are specially designed for underwater robotic systems. These control designs are also inspired from the way humans execute a particular task in Cartesian space. Further explanation about the proposed task-space controllers can be found in Chapters 4 and 5.

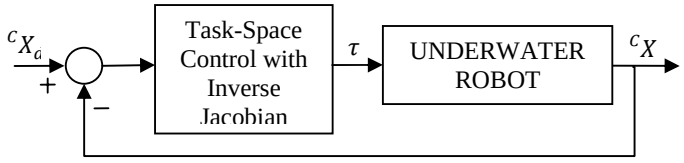


Figure 2.3: Task-space control scheme

2.3.2 Joint Space Control Schemes

The main goal of joint space control is to design a feedback controller such that the joint coordinates track the desired trajectory as closely as possible. Most of the control objectives for an underwater robot manipulator are naturally achieved by means of joint space control where the control inputs are the joint torques. However, the desired motion is essentially specified by users in terms of operational space coordinates.

Figure 2.4 depicts the basic outline of the joint space control methods. Firstly, the desired motion, which is described in terms of end-effector coordinates, $^{c}X_{d}$, is converted to a corresponding joint trajectory, q_d, using the inverse kinematics of the

robot. Then the feedback controller determines the joint torque, τ, necessary to move the robot along the desired trajectory defined in joint coordinates starting from measurement of the current joint states.

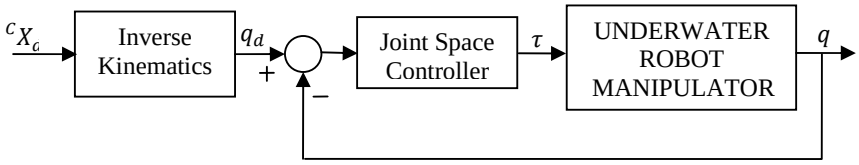


Figure 2.4: Joint space control scheme

Typically, the inverse kinematics are performed for some intermediate task points, producing the joint solutions. Although the command trajectory consists of straight-line motion in end-effector coordinates between interpolation points, the resulting joint motion consists of curvilinear segments that match the desired end-effector trajectory at the interpolation points. In fact, the joint space control techniques include simple PD control, inverse dynamic control, Lyapunov-based control and passivity-based control.

2.3.3 Body-Fixed Frame Control Schemes

The control action for an underwater vehicle, i.e. a fully actuated 6-DOF underwater vehicle, can be alternatively expressed in the body-fixed reference frame. Therefore, it is referred to as a body-fixed frame control scheme. The trajectory in task-space is projected onto the body-fixed frame using either the inverse or transpose of the Jacobian matrix. Since the error variables are defined in the body-fixed reference frame, the control law is suitable for effective compensation of the restoring moment which is generally known as a vehicle-fixed disturbance. Several controllers based on the body-fixed frame representation for autonomous underwater vehicles have been proposed over the past few years [2.15]-[2.18]. A body-fixed frame based tracking controller for a 6-DOF vehicle was developed by Fjellstad and Fossen in [2.15] and experimental results with a similar control law were reported in [2.16]. Conte and Serrani [2.17] developed a Lyapunov-based control scheme for autonomous underwater vehicles where the position errors were expressed in the body-fixed representation. Antonelli in [2.18] proposed an adaptive body-fixed and joint space controller for an underwater vehicle-manipulator system which attempted to overcome the occurrence of kinematic singularities, thus avoiding the inversion of the system Jacobian.

2.4. Motion Controllers

The motion controllers presented in this section are classified into two main classes, specifically regulation controllers and tracking controllers. The following subsections briefly explain the control problem and its generic system representation for a given control objective. Additionally, the required steps to analyse the stability of the controllers are also presented.

2.4.1 Set-Point Regulation Control

The regulation control scheme is also known as set-point control or point-to-point control. For the case of underwater robots, station keeping or regulation control tasks may be recognised as one of the most important aims in control of autonomous underwater robots. A fixed configuration in the work space is specified; the objective is to bring and keep the system at a desired position in spite of disturbances and should be independent of the initial conditions. In other words, the regulation control problem can be defined in the following terms. Given a desired constant position (set-point reference), it is required to find a control input such that the desired robot's position in Cartesian coordinates is achieved as accurately as possible.

Next, the stability analyses of a group of regulation controllers for underwater robots, which is the most essential property of a control system, are covered. The methodology to analyse the stability can be summarised in the following steps.

Firstly, the derivation of the closed-loop dynamic equation is performed by replacing the external forces and torques featured in the dynamic model of the underwater robot with the proposed regulation control action. The closed-loop equation is generally known as a non-autonomous nonlinear ordinary differential equation. The corresponding block diagram of the closed-loop system is shown in Figure 2.5. Note that, ${}^{C}\dot{X}$ represents the velocity vector in task-space and the regulation control does not rely on the acceleration input, ${}^{C}\ddot{X}$, since measurement of acceleration is typically sensitive to noise disturbances.

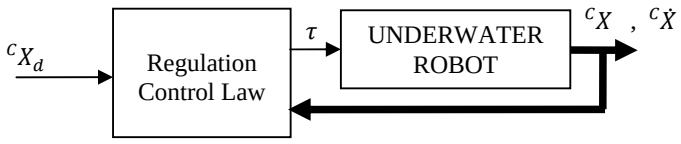


Figure 2.5: Regulation control: closed-loop system

Secondly, the existence and possible uniqueness of equilibrium for the closed-loop equation is studied whereby the closed-loop equation is rewritten in state-space form choosing the position error and the velocity to represent the states. Note that, the closed-loop system equation is autonomous since the desired position is constant. Thus, the origin is determined where it is an equilibrium and is unique.

Thirdly, a Lyapunov-candidate function is proposed to study the stability of the origin for the closed-loop equation. Note that, if all the theorems in Appendix A do not apply due to one of their conditions not being met, then other possible Lyapunov candidates should be investigated such that one of these results holds.

In fact, set-point regulation control includes simple PD control, PD control with gravity compensation, PD control with desired gravity compensation and Proportional Integral Derivative (PID) control. However, this thesis only deals with PD control with gravity compensation since it can achieve global set-point control as explained in the following section. It is interesting to note that an increased value of the derivative gain decreases overshoot, but slows down the transient response and may lead to instability due to signal noise amplification because of differentiation of the error.

2.4.2 Tracking Control

The problem of tracking control for underwater robots can be formulated in the following terms. Given a set of vector functions, desired positions ${}^{C}X_d$, velocities ${}^{C}\dot{X}_d$ and accelerations ${}^{C}\ddot{X}_d$ in Cartesian coordinates, it is required to obtain the control input such that the robot's positions in Cartesian coordinates tend to track the desired position as closely as possible. The stability analyses of a group of tracking controllers for underwater robots are also presented in this thesis. The methodology to analyse the stability can be summarised by the following steps.

Firstly, the derivation of the closed-loop dynamic equation is obtained by replacing the external forces and torques featured in the dynamic robot with the proposed tracking control action. Again, the closed-loop equation is typically known as a non-autonomous nonlinear ordinary differential equation since the desired positions are time-dependent. The corresponding block diagram of the closed-loop in its input-output representation is shown in Figure 2.6. Yet again, only the feedback inputs of position, ${}^{C}X$, and velocity, ${}^{C}\dot{X}$, are used in the control vector τ because the measurement of acceleration is practically undesirable.

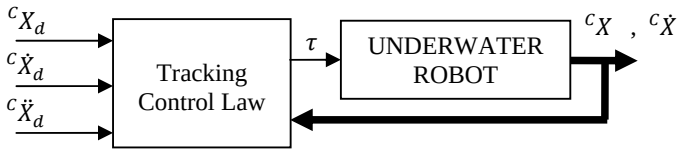


Figure 2.6: Tracking control: closed-loop system

Secondly, the existence and possible uniqueness of equilibrium for the closed-loop equation is studied whereby the closed-loop system equation is non-autonomous due to the dependence of the closed-loop function on *t*.

A Lyapunov-candidate function is then proposed to analyse the stability of any equilibrium of interest for the closed-loop equation, by using the appropriate theorems explained in Appendix A. Note that, La Salle's theorem cannot be invoked since the closed-loop system is characterised by a non-autonomous differential equation.

Typical methods of tracking control approaches include inverse dynamic control, the feedback linearisation technique and the passivity-based control method. In the following section, the computed-torque control, also known as a technique of applying feedback linearisation to nonlinear systems is briefly discussed.

2.5. Methods for Robot Control

Many research efforts have been devoted to the application of advanced control in robotics and it is the topic of several books [2.1, 2.5]. The Proportional-Derivative (PD) plus gravity compensation controller is widely employed due to its simplicity. Advanced controllers can provide a better control system, in terms of accuracy and

speed, over a wider range of operating conditions. In addition, advanced concepts such as manipulability control of a redundant robot, can be performed based on Cartesian or task-space control strategies, thus widening the range of tasks that can be automated.

Due to the computational burden and complexity, any advanced controller favours implementation on a digital computer. The digital computer offers a simple interface to other computer-based sub-systems, for instance vision and planning functions, which reduce the system's overall complexity. This section presents different control techniques that can be utilised within the robot control structure and are categorised into well known control groups.

2.5.1 PD Control for Regulation

The objective of set-point regulation control can be ideally achieved using PD control. In industrial robot manipulators, the PD controller is recognised as the simplest closed-loop controller and the application of this control strategy is common in angular position control of DC motors. In fact, the utilisation of this linear control scheme is derived from the linearisation of the system about an operating point. An example of this method is a PD controller with a gravity compensation scheme. Gravity compensation acts as a bias correction, compensating only for the forces that create overshoot and an asymmetric transient behaviour.

For underwater robotic systems, past researchers have shown an interest in the area of set-point control schemes due to the large number of underwater tasks requiring the robotic system to perform a point-to-point motion [2.1, 2.3, 2.4, 2.5, 2.19]. Within this control framework, the PD controller plus restoring forces compensation is recognised as the simplest global set-point control for underwater robots as reported in [2.1, 2.20].

2.5.2 Computed-Torque Control Techniques

As previously discussed, a proportional-derivative controller is well-known as the ideal structure to control a pure inertia, and it can effectively solve most of the set-point regulation problems. However, there are many tasks that require trajectory tracking capabilities such as high speed operations in the presence of obstacles. In this case the position of the robot at the appropriate time must be determined before the robot

executes any useful work. If local schemes (i.e. PID controllers) are employed, the system moves slowly through a number of intermediate set-points, thus considerably delaying the completion of the task. Furthermore, such fixed gain controllers are only tuned for one particular set of conditions and if these change the control action will degrade. To guarantee stability the controller is often tuned for the worst possible situation and hence the system will have a slow, sub-optimal response for most conditions. Therefore, using the robot dynamic model via a computed-torque technique should be considered in order to improve the performance of trajectory tracking. This technique is also occasionally referred to as a model-based controller.

An ideal computed-torque controller consists of the inverse of the system dynamics, used as a pre-compensator to the actual system. The control inputs required to meet the desired positions, velocities and accelerations can then be calculated directly from the inverse system model. Thus, the system is driven open-loop with perfect cancellation between the inverse dynamics and the real system. Clearly this is impractical as no real system is known perfectly, and any unmodelled effects will not be compensated for. Therefore, a feedback control scheme is used to alleviate this and can be introduced by augmenting the open-loop model based controller with a classical, usually fixed gain PID, feedback controller. These two controllers are often referred to as the primary and secondary controllers [2.21]. The primary controller can be represented as an input transformation that is used to exactly linearise the nonlinear system. Then, a secondary controller is designed to regulate the nominal linear system, producing a stable closed-loop system. The secondary controller has also been utilised to ensure set-point tracking and disturbance rejection. This approach, also known as the feedback linearising method can be depicted as in Figure 2.7. Note that the schematic diagram consists of an inner nonlinear compensation loop and an outer loop with an exogenous control signal v.

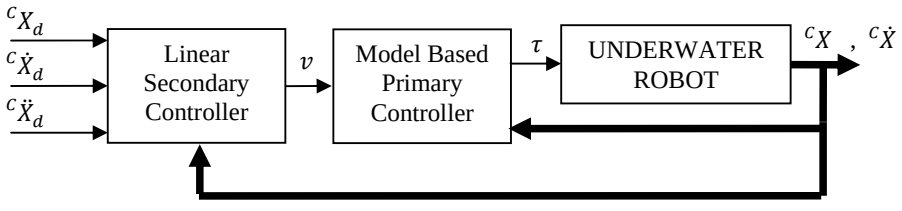


Figure 2.7: Feedback linearisation

Tarn and Yang presented a feedback linearisation method for underwater robotic manipulators in [2.22]. An extension to a full dynamic compensation controller with the derivation of the dynamic model using Kane's equation was also reported by Tarn et al. in [2.23]. Schjølberg et al. also proposed a similar control method for underwater vehicle-manipulator systems which can be found in [2.24, 2.25]. It should be noted that the assumption of exact dynamic compensation has to be made in order to yield the system's stability.

2.5.3 Robust Control Strategies

Robust control techniques were initially devised to address the problem of poorly known system dynamics and they are therefore insensitive to modelling errors and variations in the system under control. Robust controllers have been used in the secondary controller part of model based schemes to cope with the presence of uncertainties in the model based primary controller.

One nonlinear robust control technique, termed Variable Structure Control (VSC) or sliding mode control, uses a discontinuous switching function [2.26, 2.27]. This drives the system rapidly onto a switching line or sliding surface, defined in the state-space of the system. After this initial reaching phase, the system response is then governed entirely by the equation of the line, called the sliding mode. The system then remains on the sliding surface and is insensitive to disturbances and system variations, hence providing robustness. The resulting control law is a discontinuous switching function and, due to the discrete implementation, the control signal rapidly alternates between different values. This phenomenon is known as chattering and is problematic since excessive activity of the control signal can cause heating and rapid wear within the actuators. Another problem is that the high frequency content of the signal can excite unmodelled dynamics of the manipulator, such as flexibility.

The theory behind VSC is based entirely on continuous time-systems and a discrete-time implementation is an approximation of this. To ensure the stability of a discrete VSC, high sample rates are required to prevent the system moving away from the sliding surface during sample intervals. The requirement of high sample rates counteracts one of the main advantages of VSC, namely their low computational requirements.

The problem of chattering is present to an even greater degree with VSC and several approaches have been proposed to reduce this. One technique is to split the control signal into continuous and discrete components [2.27], another involves using a finite width boundary layer either side of the sliding surface [2.26]. A recently proposed VSC reduced chattering by increasing the switching frequency beyond the bandwidth of the system using appropriate hardware [2.28]. Another utilised fuzzy tuning rules to achieve the same objective [2.29].

Another difficulty with VSC is that the derivative of the error signal is required to realise a first order sliding surface and this can be problematic if signals are noisy. Further problems may arise with VSC if the initial state of the system under control is far from the sliding surface, since during the reaching phase the system dynamics are undefined and it may never reach the surface.

In spite of these problems, application of VSC to robots now forms an extensive body of work, ranging from Single-Input Single-Output (SISO) control [2.26] to multivariable robot control [2.27] and hybrid position/force control [2.28, 2.30]. A good survey of robust control techniques and application to manipulator robots is given in [2.31], which covers all of the major categories of robust control techniques mentioned above.

Since the sliding mode controllers can maintain robustness against various kinds of uncertainties such as external disturbances and measurement errors, they also have been deployed for underwater robotic systems [2.32]-[2.36]. Sliding mode controllers for an underwater vehicle and underwater gripper were proposed by Yoerger et al. in [2.32, 2.33] and Bartolini et al. in [2.34]. Xu and his colleagues have dealt with the application of a model-based sliding mode control theory to an underwater manipulator [2.35]. Experiments on a 3-DOF underwater manipulator were performed to illustrate the effectiveness of the method. They then extended their work to a fuzzy based-sliding mode controller for trajectory tracking of an underwater-manipulator system as reported in [2.36].

Antonelli and Chiaverini have designed a sliding mode controller which avoids kinematic singularities of the UVMS in [2.37]. However, their controller assumes the

perfect estimation of gravity and buoyancy forces. Actually, in practice these forces are usually difficult to estimate accurately. An adaptive sliding mode controller based on the estimation of some unknown parameters in the UVMS dynamic model was proposed in [2.18]. Nevertheless, it is rather difficult to build a dynamic model for an underwater robot that possesses so many degrees-of-freedom which increases the complexity of the adaptive controller designs. The advantages and disadvantages of the use of adaptive controllers are presented in the following sub-section.

2.5.4 Adaptive Controllers

The controllers discussed in the preceding sub-sections have several implicit assumptions; the robot model is accurately known, i.e. all the nonlinearities involved are known or negligible and the physical parameters are constant. Under these assumptions, the controllers are designed to satisfy certain stability and convergence properties even when there are variations in the system under control. In practical underwater robot control, it is difficult to estimate the mass of the payloads carried by the robot, which depends on the task to be accomplished. Therefore, it is impossible to use the typical model-based control laws since they rely on accurate knowledge of the dynamic model.

Adaptive control automatically adjusts the controller gains as the system changes. Within this design approach, the uncertainty in the dynamic system is assumed to be defined by a set of unknown constant parameters. The use of an adaptive controller stabilises the closed-loop system in response to the variations that are present in the system. Model-based adaptive schemes that are capable of satisfying the position and tracking control objective without requiring exact knowledge of the numerical values involved in the dynamic model can range from simple gravitational compensation schemes to feedback linearisation of the full manipulator dynamics. Variants of the adaptive scheme have been proposed in [2.38], known as an adaptive computed-torque control that requires the acceleration to be measurable and the bounded range of the unknown parameter is available. Another proposed scheme [2.12] which avoided the drawback of the adaptive computed-torque control scheme is called adaptive inertia-related control. In this scheme, measurement of the robot's acceleration and the inversion of the estimated inertia matrix are not required. However, these model-based adaptive controllers are generally only practical if the number of estimated parameters is restricted. The problem becomes complex if the full robot model is to be estimated.

To overcome these problems, simpler adaptive schemes have been investigated which use a low order linear approximation of the robot model. The Model Reference Adaptive Controller (MRAC) is one such scheme which utilises an adaptation algorithm to adjust the controller gains so that the output of the actual system follows the specified reference model. This method has been proposed by Dubowsky [2.39] and it has been validated experimentally [2.40]. The advantage of these controllers is that they require only a moderate number of computations and do not contain any of the complex mathematical dynamic models used in the previous methods.

Adaptive controllers have been applied to most underwater robot control problems [2.1, 2.41, 2.42, 2.43] and experimental results have been presented in [2.14, 2.16]. These schemes overcome the inadequacies of fixed gain controllers which cannot take into account variable operating conditions and unknown dynamics. However, these controllers require a regressor of the dynamic model which includes the inertia matrix, Coriolis and centripetal forces, hydrodynamic damping and gravity and buoyancy forces. Hence, the number of dynamic parameters to be updated by the adaptive law is significant.

2.5.5 Other Control Schemes

The use of adaptive controllers was previously discussed for robot tracking control. One of the advantages of the adaptive control law is that the control implementation does not require *a priori* knowledge of unknown constant parameters such as payload masses or hydrodynamic coefficients. However, the requirement of significant on-line calculation and the lack of robustness to additive bounded disturbances are drawbacks of the adaptive controllers.

The utilisation of robust control schemes for an underwater robot was also discussed. Robust controllers exhibits two attractive features; on-line computation is kept to a minimum and they possess inherent robustness to additive bounded disturbances. One of the disadvantages of the robust control approach is that these controllers require *a priori* known bounds on the uncertainty. In general, calculations on the uncertainty can be a tedious process since this calculation involves finding the maximum values for the constant parameters of each degree-of-freedom of the robot. Another disadvantage of

the robust control approach is that even in the absence of the additive bounded disturbances, asymptotic stability of the tracking error cannot be guaranteed.

An adaptive robust controller can be thought of as combining the best qualities of the adaptive nature of the adaptive controller and the robust controller. These have shown the ability to deliver powerful solutions to problems that have proved difficult because of parametric uncertainties and the need for disturbance rejection. This control approach has the advantages of reduced online calculations compared to adaptive control, robustness to additive bounded disturbances, no required *a priori* knowledge of the system uncertainty and asymptotic tracking error performance.

Adaptive robust control was applied to an underwater robotic system by Fossen et al. in [2.42] to compensate the uncertainties in the thruster dynamic configuration matrix. Using this proposed control method, the adaptive scheme estimates the dynamic parameters while the switching term is added to the controller to compensate for uncertainties in the thruster control input. Also, this hybrid control scheme was utilised for the undersea robot manipulator in [2.44] where the unknown added mass, added moment of inertia and drag force and friction are estimated by the direct adaptive control scheme. The drag force which is the dominant disturbance is compensated by the robust control scheme.

More advanced control approaches such as neural networks and fuzzy logic have been extensively developed. Neural networks have been applied to control the underwater robotic system where the network learns the characteristics of the robot by adjusting its own weightings. The neural network forms a nonlinear model of the robot which can then be used within any of the aforementioned computed torque control and adaptive control schemes.

An early investigation of the on-line approach of neural networks for the underwater robotic vehicle control system was performed by Yuh in [2.45] where the control technique was inspired from the feasibility of manipulator control using a neural network. Moreover, Ishii et al. [2.46] proposed an online adaptive method for a neural network controller for autonomous underwater vehicles. Their aim was to improve the time-consuming adaptation process. A good review of recent research efforts in the field of application of Neural Networks (NNs) for control of the underwater vehicles is

given in [2.47], which classified all the major neural networks approaches into a number of identified categories. The development of fuzzy logic controllers is based on heuristic and qualitative rules, rather than the algebraic and differential equations of traditional controllers. Many forms of fuzzy controllers for underwater robots have been proposed, including a fuzzy hybrid controller [2.48], robust fuzzy [2.49, 2.50] and neural fuzzy control law [2.51, 2.52]. It is worthwhile to remark that these works are predominantly restricted to simulation results.

2.6. Summary

This chapter has described the main approaches to the problem of underwater robot control. It first considered the problems associated with subsea robot control and then discussed the distinction between several control objectives, namely task-space, joint space and body-fixed frame control schemes. In addition, this chapter discussed issues regarding the selection of task-space controllers for their use in a subsea robot application. Many different control techniques that have been applied to underwater robot control were then presented. The merits and drawbacks of each particular method were discussed and instances of successful applications of each technique were highlighted.

The detail of the developed controllers and their novel features are presented in the next chapters. Due to the superiority of adaptive and robust control terms, they are also added in the control laws. To show how they complement previous work, the developed controllers are placed in the context of previously proposed underwater robot controllers.

2.7. References

[2.1] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, 1st ed. New York: John Wiley and Sons, 1994.

[2.2] S. M. Smith, G. J. S. Rae, D. T. Anderson and A. M. Shein, "Fuzzy Logic Control of an Autonomous Underwater Vehicle," *Control Engineering Practice* , vol. 2, no. 2, pp. 321-331, 1994.

[2.3] Y. C. Sun and C. C. Cheah, "Adaptive Setpoint Control for Autonomous Underwater Vehicles," in *Proc. 42nd IEEE Conference on Decision and Control*, Maui, Hawaii USA, 2003, pp. 1262 - 1267.

[2.4] P. Herman, "Decoupled PD Set-Point Controller for Underwater Vehicles," *Ocean Engineering*, vol. 36, pp. 529-534, 2009.

[2.5] G. Antonelli, *Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems*. Germany: Springer-Verlag Berlin, 2003.

[2.6] T. I. Fossen, "Adaptive Macro-Micro Control of Nonlinear Underwater Robotic Systems," in *Fifth International Conference on Advanced Robotics*, vol. 2, Pisa, Italy, Jun 1991, pp. 1569-1572.

[2.7] N. Sarkar, J. Yuh and T. K. Podder, "Adaptive Control of Underwater Vehicle-Manipulator Systems Subject to Joint Limits," in *Proc. IEEE/RJS International Conference on Intelligent Robots and Systems*, vol. 1, Kyongju, Korea, Oct. 1999, pp. 142-147.

[2.8] A. C. Clegg, *Self-tuning Position and Force Control of a Hydraulic Manipulator*.: Ph.D Thesis, Heriot-Watt University, 2000.

[2.9] F. Caccavale, C. Natale, B. Siciliano, and L. Villani, "Resolved-Acceleration Control of Robot Manipulators: A Critical Review with Experiments," *Robotica*, vol. 16, no. 5, pp. 565-573, 1998.

[2.10] O. Khatib, "Dynamic Control of Manipulators in Operational Space," in *IFTOMM Cong. Theory of Machines and Mechanisms*, New Delhi, India, 1983, pp. 1-10.

[2.11] J. S. C. Yuan, "Closed-Loop Manipulator Control Using Quaternion Feedback," *IEEE Trans. Robotics and Automation*, vol. 4, no. 4, pp. 434-440, 1988.

[2.12] J. J. Slotine and W. Li, "On the Adaptive Control of Robot Manipulators," *International Journal of Robotics Research*, vol. 6, no. 3, pp. 49-59, 1987.

[2.13] O. Fjellstad and T. I. Fossen, "Position and Attitude Tracking of AUVs: A Quaternion Feedback Approach," *Journal of Oceanic Engineering*, vol. 19, no. 4, pp. 512-518, 1994.

[2.14] J. Yuh, J. Nie, and C. S. G. Lee, "Experimental Study on Adaptive Control of Underwater Robots," in *Proc. of the IEEE International Conference on Robotics and Automation*, Detroit, Michigan, 1999, pp. 393–398.

[2.15] O. Fjellstad and T. I. Fossen, "Singularity-Free Tracking of Unmanned Underwater Vehicle in 6 DOF," , Lake Buena Vista, Florida, 1994, pp. 1128-1133.

[2.16] G. Antonelli, S. Chiaverini, N. Sarkar and M. West, "Adaptive Control of an Autonomous Underwater Vehicle: Experimental Results on ODIN," in *IEEE Transaction on Control Systems and Technology*, vol. 9, 2001, pp. 756-765.

[2.17] G. Conte and A. Serrani, "Robust Lyapunov-Based Design for Autonomous Underwater Vehicles," in *5th IFAC Symposium on Robot Control*, Nantes, France, 1997, pp. 321-326.

[2.18] G. Antonelli and S. Chiaverini, "Adaptive Tracking Control of Underwater Vehicle-Manipulator Systems," in *Proc. IEEE International Conference on Control Applications*, Trieste, Italy, 1998, pp. 1089-1093.

[2.19] F. Lizarralde, J. T. Wen and L. Hsu, "Quaternion-Based Coordinated Control of a Subsea Mobile Manipulator with only Position Measurements," in *Proc. of the 34th IEEE Conference on Decision and Control*, vol. 4, New Orleans, LA, 1995, pp. 3996-4001.

[2.20] M. Takegaki and S. Arimoto, "A New Feedback Method for Dynamic Control of Manipulators," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 102, pp. 119–125, 1981.

[2.21] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*. New York: Wiley, 1989.

[2.22] T. J. Tarn and S. P. Yang, "Modeling and Control for Underwater Robotic Manipulators - An Example," in *Proc. IEEE Int. Conf. Robotics and Automation*, Albuquerque, NM, 1997, pp. 2166-2171.

[2.23] T. J. Tarn, G. A. Shoults and S. P. Yang, "A Dynamic Model for an Underwater Vehicle with a Robotic Manipulator using Kane's Method," *Autonomous Robots*, vol. 3, pp. 269-283, 1996.

[2.24] I. Schjølberg and T. I. Fossen, "Modelling and Control of Underwater Vehicle-Manipulator Systems," in *Proc. 3rd Conf. Marine Craft Manoeuvring and Control*, Southampton, U.K., 1994, pp. 45–57.

[2.25] I. Schjølberg, *Modeling and Control of Underwater Robotic Systems*.: Ph.D Thesis for the Doctor ingeniør degree, Norwegian University of Sciences and Technology, 1996.

[2.26] J. J. Slotine, "The Robust Control of Robot Manipulators," *International Journal of Robotics Research*, vol. 4, no. 2, pp. 46-64, 1986.

[2.27] D. S. Reay, *Variable Structure Control of Industrial Robots*.: Ph.D Thesis, Cambridge University, 1988.

[2.28] A. C. Clegg, L. Cellier, P.Dauchez, D. M. Lane and M. W. Dunnigan, "Comparison of Robust and Adaptive Hybrid Position/Force Control Schemes for Hydraulic Manipulators," in *6th IARP Workshop on Underwater Robotics*, Toulon, France, 1996.

[2.29] Q. P. Ha, D. C. Rye and H. F. Durrant-Whyte, "Fuzzy Moving Sliding Mode Control with Application to Robotic Manipulators," *Automatica*, vol. 35, no. 4, pp. 607-16, 1999.

[2.30] R. R. Y. Zhen and A. A. Goldenberg, "Variable Structure Hybrid Control of Manipulators in Unconstrained and Constrained Motion," *Journal of Dynamic Systems, Measurement and Control*, vol. 118, no. 2, pp. 327-32, 1996.

[2.31] H. G. Sage, M. F. de Mathelin and E. Ostertag, "Robust Control of Robot Manipulators: A Survey," *International Journal of Control*, vol. 72, no. 16, pp. 1498-1522, 1999.

[2.32] D. N. Yoerger and J. E. Slotine, "Robust Trajectory Control of Underwater Vehicles," *IEEE J. of Oceanic Engineering*, vol. 10, no. 4, pp. 462-470, 1985.

[2.33] D. R. Yoerger, J. B. Newman and J. J. E. Slotine, "Supervisory Control System for the Jason ROV," *IEEE Journal of Oceanic Engineering*, vol. 11, pp. 392-400, 1986.

[2.34] G. Bartolini, M. Coccoli and E. Punta, "Sliding Mode Control of an Underwater Robotic Manipulator," in *Proc. IEEE Conference on Decision and Control*, Sydney, Australia, 2000, pp. 2983-2988.

[2.35] B. Xu, S. R. Pandian, M. Inoue, N. Sakagami, and S. Kawamura, "Model-Based Sliding Mode Control of Underwater Robot Manipulators," *International Journal of Offshore and Polar Engineering*, vol. 16, pp. 210-217, 2006.

[2.36] B. Xu, S.R. Pandian, and F. Petry, "A Sliding Mode Fuzzy Controller for Underwater Vehicle-Manipulator Systems," in *National Association of Fuzzy Information Processing Systems Annual Conference (NAFIPS05)*, 2005, pp. 181 - 186.

[2.37] G. Antonelli and S. Chiaverini, "Singularity-Free Regulation of Underwater Vehicle-Manipulator Systems," in *Proceedings of the American Control Conference 1998*, vol. 1, 1998, pp. 399-403.

[2.38] J. J. Craig, P. Hsu and S. S. Sastry, "Adaptive Control of Mechanical Manipulators," *International Journal of Robotics Research*, vol. 6, no. 2, pp. 16-28, June 1987.

[2.39] S. Dubowsky and D. T. DesForges, "The Application of Model Referenced Adaptive Control to Robotic Manipulators," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 101, pp. 193-200, 1979.

[2.40] F. G. de Almeida, *Model Reference Adaptive Control of a 2 Axes Hydraulic Manipulator*.: Ph.D Thesis, Bath University, 1993.

[2.41] S. Zhao and J. Yuh, "Adaptive DOB Control of Underwater Robots," in *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, 2003, pp. 571-576.

[2.42] T. I. Fossen and S. I. Sagatun, "Adaptive Control of Nonlinear Systems: A Case Study of Underwater Robotic Systems," *Journal of Robotic Systems*, vol. 8, pp. 393-412, 1991.

[2.43] J. Yuh, "Modeling and Control of Underwater Robotic Vehicles," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 20, no. 6, pp. 1475–1483, 1990.

[2.44] Y.-S. Kim and H.-S. Choi, "An Adaptive and Robust Controller for the Undersea Robot Manipulator," *International Journal of Precision Engineering and Manufacturing*, vol. 4, pp. 13-22, 2003.

[2.45] J. Yuh, "A Neural Net Controller for Underwater Robotic Vehicles," *IEEE Journal of Oceanic Engineering*, vol. 15, no. 3, pp. 161-166, 1990.

[2.46] K. Ishii, T. Fujii and T. Ura, "An On-line Adaptation Method in a Neural Network Based Control System for AUVs," *IEEE Journal of Oceanic Engineering*, vol. 20, no. 3, pp. 221 - 228, 1995.

[2.47] P. W. J. v. de Ven, C. Flanagan and D. Toal, "Neural Network Control of Underwater Vehicles," *Engineering Applications of Artificial Intelligence*, vol. 18, no. 5, pp. 533-547, 2005.

[2.48] C. H. dos Santos, G. Bittencourt and R. Guenther, "Motion Coordination for Underwater Vehicle-Manipulator Systems using a Fuzzy Hybrid Strategy," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp.

3018 - 3023.

[2.49] H. S. Choi, "Robust Control of Robot Manipulators with Torque Saturation using Fuzzy Logic," *Robotica*, vol. 19, no. 6, pp. 631 - 639, 2001.

[2.50] W. M. Bessa, M. S. Dutra and E. Kreuzer, "An Adaptive Fuzzy Sliding Mode Controller for Remotely Operated Underwater Vehicles," *Robotics and Autonomous Systems*, vol. 58, no. 1, pp. 16-26, 2010.

[2.51] J. S. Wang and C. S. G. Lee, "Efficient Neuro-Fuzzy Control Systems for Autonomous Underwater Vehicle Control," in *Proc. IEEE International Conference on Robotics and Automation*, 2001, pp. 2986 - 2991.

[2.52] J. S. Wang and C. S. G. Lee, "Self-Adaptive Recurrent Neuro-Fuzzy Control of an Autonomous Underwater Vehicle," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 2, pp. 283-295, 2003.

CHAPTER 3
KINEMATIC AND DYNAMIC MODELLING OF UNDERWATER ROBOTS

3.1. Introduction

In this chapter, a mathematical model for underwater robots is presented. The kinematic and dynamic modelling of an underwater robotic system is presented as it is an essential requirement for accurate control of the system. Over the past decades, many research works have been carried out to derive the kinematic and dynamic models for rigid bodies that are moving in a fluid environment. Some of the previous literature [3.1, 3.2] presented the mathematical models for 6-DOF underwater robotic vehicles. A complete survey of the dynamic models for underwater vehicles was also reported by Yuh in [3.3, 3.4].

The modelling becomes more complicated when one or more manipulators are mounted on the vehicle. In this case, the effect of the hydrodynamic forces on each link of the manipulator on vehicle motion has to be considered when modelling the vehicle and manipulator [3.5, 3.6]. For a more complex model, for instance, two UVMSs holding the same rigid object [3.7], the modelling needs to be fully established in simulation phases before the design process can be implemented in actual field-testing.

Various modelling approaches for an underwater robotics system have been proposed by previous researchers [3.5, 3.8, 3.9, 3.10, 3.11]. Manesh et al. [3.5] derived a discrete-time model for the underwater system to be used with a discrete-time adaptive controller. In [3.8], the authors used the classical Newton-Euler formalism to compute a recursive formulation for the model of an underwater vehicle-manipulator system. A recursive algorithm was proposed by Janocha and Papadimitriou in [3.9] to generate a dynamic model of an underwater robot. Simulation results were presented to validate their proposed method. McMillan et al. developed an efficient dynamic simulation algorithm in [3.10] which included a mobile base and the hydrodynamic forces. Meanwhile, the closed-form model of an underwater vehicle-manipulator system was derived by Schjølberg and Fossen [3.11] using the iterative Newton-Euler algorithm.

3.2. Kinematics of a Rigid-Body

The motion of a 6-DOF rigid body is best described in a moving coordinate frame which is called the body-fixed reference frame, $\sum_v - X_v Y_v Z_v$. The origin of the body-fixed frame is typically chosen to coincide with the centre of gravity (CG) when the CG is in the principle plane of symmetry. Note that, the position and orientation of the rigid body should be expressed relative to the earth-fixed or inertial reference frame $\sum_i - X_i Y_i Z_i$, while the linear and angular velocities of the rigid body should be described in the body-fixed coordinate system. Define the position vector in the inertial reference frame $\boldsymbol{\eta}_1 \in \mathbb{R}^3$ as

$$\boldsymbol{\eta}_1 = [x \quad y \quad z]^T \tag{3.1}$$

The time derivative of the position vector is given by $\dot{\boldsymbol{\eta}}_1 \in \mathbb{R}^3$, where it is expressed in the inertial reference frame. The linear velocity vector of the body-fixed frame with respect to the origin of the inertial reference frame expressed in the body-fixed frame is represented as

$$\boldsymbol{v}_1 = [u \quad v \quad w]^T \tag{3.2}$$

Next, the attitude representations, namely Euler angles and Euler parameters or the quaternion are presented. Note that, the use of Euler angles exhibits singularities which leads to the use of the quaternion in several control strategies [3.12, 3.13, 3.14].

Table 3.1 Common notation for marine vehicles

		Forces and moments	$\boldsymbol{v}_1, \boldsymbol{v}_2$	$\boldsymbol{\eta}_1, \boldsymbol{\eta}_2$
Motion in the x-direction surge	surge	X	u	x
Motion in the y-direction surge	sway	Y	v	y
Motion in the z-direction surge	heave	Z	w	z
Rotation about the x-axis	roll	K	p	ϕ
Rotation about the y-axis	pitch	M	q	θ
Rotation about the z-axis	yaw	N	r	ψ

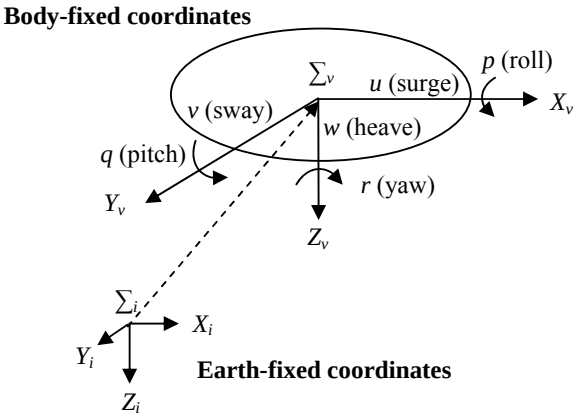


Figure 3.1: Coordinate system representation of an underwater robotic system

Table 3.1 summarises the common notation used for marine robots according to the SNAME (The Society of Naval Architects and Marine Engineers) notation [3.15] while Figure 3.1 illustrates the coordinate frames.

3.2.1 Euler Angles

The relation between the defined linear velocities is given by

$$\dot{\boldsymbol{\eta}}_1 = J_1(\boldsymbol{\eta}_2)\boldsymbol{v}_1 \tag{3.3}$$

where $\boldsymbol{\eta}_2 = [\phi \quad \theta \quad \psi]^T \in \mathbb{R}^3$ is the vector of body Euler angle coordinates in the inertial reference frame and $J_1(\boldsymbol{\eta}_2)$ is a transformation matrix. The inverse velocity transformation is written as

$$\boldsymbol{v}_1 = J_1^{-1}(\boldsymbol{\eta}_2)\dot{\boldsymbol{\eta}}_1 = J_1^T(\boldsymbol{\eta}_2)\dot{\boldsymbol{\eta}}_1 \tag{3.4}$$

The transformation matrix $J_1(\boldsymbol{\eta}_2)$ is expressed in terms of the Euler angles by

$$J_1(\boldsymbol{\eta}_2) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{3.5}$$

where $\mathrm{s}\cdot = \sin(\cdot)$ and $\mathrm{c}\cdot = \cos(\cdot)$.

The Euler rate vector $\dot{\boldsymbol{\eta}}_2$ is related through a transformation matrix $J_2(\boldsymbol{\eta}_2)$ so that

$$\dot{\boldsymbol{\eta}}_2 = J_2(\boldsymbol{\eta}_2)\boldsymbol{v}_2 \tag{3.6}$$

where $\boldsymbol{v}_2 = [p \quad q \quad r]^T$ is the body-fixed angular velocity vector. It should be noted that the vector $\boldsymbol{v}_2$ cannot be integrated directly to obtain actual angular coordinates because the integration of $\boldsymbol{v}_2$ does not have a physical interpretation. The transformation matrix $J_2(\boldsymbol{\eta}_2)$ is written in Euler angles as

$$J_2(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \tag{3.7}$$

where $t\cdot = \tan(\cdot)$ and (3.7) is singular for $\theta = \pm(\pi/2)$ rad.

3.2.2 Unit Quaternion Representation

In 3D underwater operational space, a 3-parameter representation such as roll, pitch and yaw forms only a local parameterisation and exhibits singularities [3.1]. Alternatively, the unit quaternion (or Euler parameters) can be used to represent attitude without singularities with one constraint equation. Moreover, the quaternion provides a global nonsingular parameterisation with desirable computational properties [3.16]. Consider two orthonormal right-handed coordinate frames: the inertial reference frame, $\sum_i$ and body-fixed frame, $\sum_v$. Define the matrix R, as a 3×3 rotational matrix from the body-fixed frame to the inertial-fixed frame. The unit quaternion representation of the rotational matrix, R, can be defined by

$$\boldsymbol{\epsilon} = [\epsilon_0 \quad \epsilon_1 \quad \epsilon_2 \quad \epsilon_3]^T = [\epsilon_0 \quad \boldsymbol{\epsilon}_\varepsilon^T]^T \in \mathbb{R}^4 \tag{3.8}$$

with

$$\epsilon_0 = \cos(\vartheta/2); \boldsymbol{\epsilon}_\varepsilon \triangleq k \sin(\vartheta/2) \tag{3.9}$$

where ϑ is the angle and $k(t) \in \mathbb{R}^3$ are the Euler angle/axis parameters subject to the constraint $\epsilon^T\epsilon = 1$.

The rotational matrix can be determined through

$$R = (\epsilon_0{}^2 - \boldsymbol{\epsilon}_\varepsilon{}^T \boldsymbol{\epsilon}_\varepsilon) I_3 + 2\boldsymbol{\epsilon}_\varepsilon \boldsymbol{\epsilon}_\varepsilon{}^T - 2\epsilon_0 \boldsymbol{\epsilon}_\varepsilon{}^\times \tag{3.10}$$

where for any vector $\boldsymbol{a} = [a_1 \quad a_2 \quad a_3]^T$, the notation $\boldsymbol{a}^\times$ denotes the skew-symmetric matrix of the form

$$\boldsymbol{a}^\times \triangleq \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{3.11}$$

where the product $\boldsymbol{a}^T \boldsymbol{a}^\times$ satisfies the following property

$$\boldsymbol{a}^T \boldsymbol{a}^\times = [0 \quad 0 \quad 0]^T \tag{3.12}$$

The transformation matrix relating the linear velocity vector in the inertial reference frame $\dot{\boldsymbol{\eta}}_1$ to the velocity in the body-fixed reference frame $\boldsymbol{v}_1$ can be determined in quaternion form as follows

$$E_1(\boldsymbol{\epsilon}) = \begin{bmatrix} 1 - 2(\epsilon_2^2 + \epsilon_3^2) & 2(\epsilon_1\epsilon_2 - \epsilon_3\epsilon_0) & 2(\epsilon_1\epsilon_3 + \epsilon_2\epsilon_0) \\ 2(\epsilon_1\epsilon_2 + \epsilon_3\epsilon_0) & 1 - 2(\epsilon_1^2 + \epsilon_3^2) & 2(\epsilon_2\epsilon_3 - \epsilon_1\epsilon_0) \\ 2(\epsilon_1\epsilon_3 - \epsilon_2\epsilon_0) & 2(\epsilon_2\epsilon_3 + \epsilon_1\epsilon_0) & 1 - 2(\epsilon_1^2 + \epsilon_2^2) \end{bmatrix} \tag{3.13}$$

Generally, the relationship between unit quaternion and angular velocity is in the body-fixed frame, $\boldsymbol{\omega}$ can be obtained using the quaternion propagation equation

$$\dot{\boldsymbol{\epsilon}} = \frac{1}{2} E_2(\boldsymbol{\epsilon}) \boldsymbol{\omega} \tag{3.14}$$

with

$$E_2(\boldsymbol{\epsilon}) = \begin{bmatrix} -\boldsymbol{\epsilon}_\varepsilon^T \\ \epsilon_0 I_3 + \boldsymbol{\epsilon}_\varepsilon{}^\times \end{bmatrix} \tag{3.15}$$

where the Jacobian $E(\boldsymbol{\epsilon})$ satisfies the following important properties

$$E_2^T(\boldsymbol{\epsilon}) E_2(\boldsymbol{\epsilon}) = I_{3\times 3} \quad ; \quad E_2^T(\boldsymbol{\epsilon}) \boldsymbol{\epsilon} = 0 \tag{3.16}$$

Consequently, from (3.15) and (3.16), the inverse kinematics can be computed as

$$\boldsymbol{\omega} = 2E_2^T(\boldsymbol{\epsilon})\dot{\boldsymbol{\epsilon}} \tag{3.17}$$

3.2.3 Attitude Error Representation

To quantify mismatch between the actual and desired attitudes, the rotation matrix $\tilde{R}$ can be represented as follows [3.16]

$$\tilde{R} \triangleq RR_d^T = \left(\tilde{e}_0{}^2 - \tilde{\boldsymbol{e}}_\varepsilon{}^T\tilde{\boldsymbol{e}}_\varepsilon\right)I_3 + 2\tilde{\boldsymbol{e}}_\varepsilon\tilde{\boldsymbol{e}}_\varepsilon{}^T - 2\tilde{\boldsymbol{e}}_0\tilde{\boldsymbol{e}}_\varepsilon{}^\times \tag{3.18}$$

where R is defined in (3.10) and R_d is the rotational matrix of R expressing the desired orientation which is also described by the quaternion $\epsilon_d \triangleq [\epsilon_{0d} \quad \epsilon_{\varepsilon d}^T]^T$. The corresponding attitude error representation, $\tilde{\boldsymbol{e}}_\epsilon \triangleq [\tilde{e}_0 \quad \tilde{\boldsymbol{e}}_\varepsilon^T]^T \in \mathbb{R}^4$ is defined as

$$\tilde{e}_0 = \epsilon_0\epsilon_{0d} + \boldsymbol{\epsilon}_\varepsilon^T\boldsymbol{\epsilon}_{\varepsilon d} \tag{3.19}$$

$$\tilde{\boldsymbol{e}}_\varepsilon = \epsilon_{0d}\boldsymbol{\epsilon}_\varepsilon - \epsilon_0\boldsymbol{\epsilon}_{\varepsilon d} + \boldsymbol{\epsilon}_\varepsilon^\times\boldsymbol{\epsilon}_{\varepsilon d} \tag{3.20}$$

Thus, the quaternion propagation equation can be considered as

$$\begin{aligned} \dot{\tilde{e}}_0 &= -\frac{1}{2}\tilde{\boldsymbol{e}}_\varepsilon^T\tilde{\boldsymbol{\omega}} \\ \dot{\tilde{\boldsymbol{e}}}_\varepsilon &= \frac{1}{2}(\tilde{e}_0 I_3 + \tilde{\boldsymbol{e}}_\varepsilon{}^\times)\tilde{\boldsymbol{\omega}} \end{aligned} \tag{3.21}$$

that is,

$$\dot{\tilde{\boldsymbol{e}}}_\epsilon = \frac{1}{2}\begin{bmatrix} -\tilde{\boldsymbol{e}}_\varepsilon^T \\ (\tilde{e}_0 I_3 + \tilde{\boldsymbol{e}}_\varepsilon{}^\times) \end{bmatrix}\tilde{\boldsymbol{\omega}} = E_2(\tilde{\boldsymbol{e}}_\epsilon)\tilde{\boldsymbol{\omega}} \tag{3.22}$$

where $\tilde{\boldsymbol{\omega}}(t) = \boldsymbol{\omega} - \boldsymbol{\omega}_d$.

Remark 3.1. The relations given in (3.19) and (3.20) can be explicitly calculated via quaternion algebra by noticing that the quaternion equivalent of (3.18) is the quaternion product [3.17]

$$\tilde{\boldsymbol{e}} = \boldsymbol{\epsilon}_d^* \boldsymbol{\epsilon} \tag{3.23}$$

where $\boldsymbol{\epsilon}_d^*(t) = [\boldsymbol{\epsilon}_{0d}(t),\ -\boldsymbol{\epsilon}_{vd}(t)] \in \mathbb{R}^4$ is the unit quaternion representing the rotation matrix $R_d^T(\boldsymbol{\epsilon}_d)$.

Transformation between Euler Angles and Euler Parameters

The transformation of the Euler parameters from the Euler angles can be realised using the following algorithm (Quaternion from Rotational Matrix) [3.1, 3.18].
Given the general transformation matrix J_1 from (3.3):

1. The trace of J_1 is computed according to :

$$J_{44} = \mathrm{tr}(J_1) = \sum_{j=1}^{3} J_{jj}$$

2. Let $1 \leq i \leq 4$ be the index corresponding to :

$$J_{ii} = \max(J_{11}, J_{22}, J_{33}, J_{44})$$

3. Define the scalar c_i as:

$$|c_i| = \sqrt{1 + 2J_{ii} - J_{44}}$$

where the sign can be chosen as either plus or minus.

4. The other three values of c can be computed from :

$$c_4 c_1 = J_{32} - J_{23}$$
$$c_4 c_2 = J_{13} - J_{31}$$
$$c_4 c_3 = J_{21} - J_{12}$$
$$c_2 c_3 = J_{32} + J_{23}$$
$$c_3 c_1 = J_{13} + J_{31}$$
$$c_1 c_2 = J_{21} + J_{12}$$

by simply dividing the three equations containing the components c_i with c_i on both sides. For example, if c_1 is chosen as the scalar in step 3, then the three equations involving c_1 in step 4 are divided by c_1 so that c_2, c_3 and c_4 can be obtained.

5. Compute the Euler parameters ϵ

$$\epsilon = [\epsilon_0 \quad \epsilon_1 \quad \epsilon_2 \quad \epsilon_3]^T = \frac{1}{2}[c_1 \quad c_2 \quad c_3 \quad c_4]^T$$

Transformation between Euler Parameters and Euler Angles

The Euler angles can be obtained from the Euler parameters based on the following relationship

$$J_1(\phi, \theta, \psi) \triangleq E_1(\boldsymbol{\epsilon}) \tag{3.24}$$

Let the element of E_1 be represented by E_{ij} where the subscripts i and j denote the i-th row and j-th column of E_1. Rewriting expression (3.24) with three unknowns (ϕ, θ and ψ) leads to

$$\theta = -\operatorname{asin}(E_{31}); \qquad \theta \neq \pm 90° \tag{3.25}$$

$$\phi = \operatorname{atan2}(E_{32}, E_{33}) \tag{3.26}$$

$$\psi = \operatorname{atan2}(E_{21}/E_{11}) \tag{3.27}$$

where $\operatorname{atan2}(y, x)$ is the four quadrant arctangent of the real elements of x and y, defined as

$$\alpha = \operatorname{atan2}(y, x) = \begin{cases} 2\pi - \operatorname{acos}(x) & \text{if } y \leq 0 \\ \operatorname{acos}(x) & \text{if } y > 0 \end{cases} \tag{3.28}$$

where $-\pi \leq \alpha \leq \pi$. Note that there are computational errors in the vicinity of $\theta = \pm 90°$.

3.2.4 6-DOF Rigid-Body Kinematics

Let the two vectors be defined as $\boldsymbol{v} = [\boldsymbol{v}_1^T \quad \boldsymbol{v}_2^T]^T \in \mathbb{R}^6$ and $\boldsymbol{\eta} = [\boldsymbol{\eta}_1^T \quad \boldsymbol{\eta}_2^T]^T \in \mathbb{R}^6$, thus the forward kinematic equation of a rigid-body in a 6-dimensional matrix form can be represented as follows

$$\dot{\boldsymbol{\eta}} = \begin{bmatrix} J_1(\boldsymbol{\eta}_2) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & J_2(\boldsymbol{\eta}_2) \end{bmatrix} \boldsymbol{v} \quad \Leftrightarrow \quad \dot{\boldsymbol{\eta}} = J_{v_a}(\boldsymbol{\eta})\boldsymbol{v} \tag{3.29}$$

where $J_1(\boldsymbol{\eta}_2)$ and $J_2(\boldsymbol{\eta}_2)$ are given in (3.5) and (3.7) respectively.

The inverse kinematic version of (3.29) can be written in the following form

$$\boldsymbol{v} = \begin{bmatrix} J_1^{-1}(\boldsymbol{\eta}_2) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & J_2^{-1}(\boldsymbol{\eta}_2) \end{bmatrix} \dot{\boldsymbol{\eta}} \quad \Leftrightarrow \quad \boldsymbol{v} = J_{v_a}^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}} \tag{3.30}$$

where $J_1^{-1}(\boldsymbol{\eta}_2)$ is defined in (3.4) and the inverse of $J_2^{-1}(\boldsymbol{\eta}_2)$ is given by

$$J_2^{-1}(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & c\theta s\phi \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \tag{3.31}$$

For the Euler parameters representation, the forward kinematics can be calculated as follows

$$\begin{bmatrix} \dot{\boldsymbol{\eta}}_1 \\ \dot{\boldsymbol{\epsilon}} \end{bmatrix} = \begin{bmatrix} E_1(\boldsymbol{\epsilon}) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{4\times3} & \frac{1}{2}E_2(\boldsymbol{\epsilon}) \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix} \quad \Leftrightarrow \quad \dot{\boldsymbol{\eta}}_e = J_{v_e}(\boldsymbol{\eta}_e)\boldsymbol{v} \tag{3.32}$$

where $\boldsymbol{v}_2 = \boldsymbol{\omega} \in \mathbb{R}^3$. $E_1(\boldsymbol{\epsilon})$ and $E_2(\boldsymbol{\epsilon})$ are given in (3.13) and (3.15) respectively. The inverse mapping can be obtained as

$$\begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix} = \begin{bmatrix} E_1^T(\boldsymbol{\epsilon}) & \mathbf{0}_{3\times4} \\ \mathbf{0}_{3\times3} & 2E_2^T(\boldsymbol{\epsilon}) \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\eta}}_1 \\ \dot{\boldsymbol{\epsilon}} \end{bmatrix} \quad \Leftrightarrow \quad \boldsymbol{v} = J_{v_e}^{\dagger}(\boldsymbol{\eta}_e)\dot{\boldsymbol{\eta}}_e \tag{3.33}$$

where the inverse transformation of matrix $E_1(\boldsymbol{\epsilon})$ satisfies $E_1^{-1}(\boldsymbol{\epsilon}) = E_1^T(\boldsymbol{\epsilon})$ while $E_2^T(\boldsymbol{\epsilon})$ is given in (3.17). The matrix $J_{v_e}^{\dagger} = \left(J_{v_e}^T J_{v_e}\right)^{-1} J_{v_e}^T \in \mathbb{R}^{6\times7}$ is also known as the generalised left inverse of J_{v_e} which satisfies $J_{v_e}^{\dagger} J_{v_e} = I_{6\times6}$.

3.3. Dynamic Model of a Rigid-Body

In this section, the Newton-Euler formulation that is widely used to derive the rigid-body equations of motion is briefly summarised. Other approaches are feasible, i.e. Lagrangian formulation and Quasi-Lagrange formulation [3.1]. It is interesting to note that Newton's laws are formulated in a body-fixed reference frame since the

hydrodynamic and kinematic forces and moments are given in the body-fixed reference frame.

Defining the mass of the body as

$$m = \int_{V_B} \rho dV \tag{3.34}$$

where ρ is the mass density of the body and V_B is the body volume. Let $\boldsymbol{r}$ be the distance from the origin of the body-fixed frame to an element of a rigid body, the centre of mass can be given as

$$\boldsymbol{r}_G = \frac{1}{m}\int_{V_B} \boldsymbol{r}\rho dV \tag{3.35}$$

Let $\boldsymbol{f}_c$ be the external forces and $\boldsymbol{r}_c = \boldsymbol{r}_0 + \boldsymbol{r}_G$ be the distance from the origin of the inertial-fixed frame to the centre of gravity. If a system has a constant mass, then Newton's law for the translational motion can be computed as

$$\boldsymbol{f}_c = m\frac{d}{dt}\dot{\boldsymbol{r}}_c \tag{3.36}$$

where (3.36) can also be rewritten as

$$\boldsymbol{f}_c = m\ddot{\boldsymbol{r}}_c \tag{3.37}$$

Next, the body's inertia tensor referred to as an arbitrary body-fixed coordinate system with the origin in the body-fixed frame is defined

$$I_0 = -\int_{V_B} (\boldsymbol{r}^T\boldsymbol{r}I_{3\times3} - \boldsymbol{r}\boldsymbol{r}^T)\rho dV \tag{3.38}$$

where (3.38) is symmetric and positive definite. The positive diagonal elements are the inertia moments with respect to the three coordinate axes of the reference frame while the off-diagonal elements are the products of the inertia. $I_{3\times3}$ is a 3×3 identity matrix.

For rotational motion, the absolute angular momentum about the origin can be defined in the following form

$$\boldsymbol{h}_0 = \int_{V_B} \boldsymbol{r} \times \boldsymbol{v}\, \rho dV \tag{3.39}$$

where $\boldsymbol{v} = \dot{\boldsymbol{r}} + \dot{\boldsymbol{r}}_0$. Taking into account the definition of the inertia tensor and centre of gravity, this expression can be simplified as

$$\boldsymbol{h}_0 = I_0\boldsymbol{\omega} + m\boldsymbol{r}_G \times \boldsymbol{v}_0 \tag{3.40}$$

where $\boldsymbol{v}_0$ and $\boldsymbol{\omega}$ are the linear and angular velocities of the body-fixed frame with respect to the inertial-fixed frame. If the origin of the body-fixed frame is chosen to be different from the vehicle's centre of gravity, then the resultant moment $\boldsymbol{m}_0$ is given by

$$\boldsymbol{m}_0 = I_0\acute{\boldsymbol{\omega}} + \boldsymbol{\omega} \times I_0\boldsymbol{\omega} + m\boldsymbol{r}_G \times (\acute{\boldsymbol{v}}_0 + \boldsymbol{\omega} \times \boldsymbol{v}_0) \tag{3.41}$$

where the acute accent represents the time derivative in the body-fixed frame. It is interesting to note that the translational and rotational equations of motion (3.37) and (3.41) are also referred to as the Newton and Euler equations. The details of derivation for both equations of motion can be found in [3.1].

3.3.1 6-DOF Rigid-Body Equation of Motion

By using the previous equations (3.37) and (3.41), the Newton-Euler equation of motion of a rigid body moving in space can be expressed in a compact form as follows

$$M_{RB}\dot{\boldsymbol{v}} + C_{RB}(\boldsymbol{v})\dot{\boldsymbol{v}} = \boldsymbol{\tau}_{RB} \tag{3.42}$$

where $\boldsymbol{v} = [u \quad v \quad w \quad p \quad q \quad r]^T$ is the body-fixed linear and angular velocity and $\boldsymbol{\tau}_{RB} = [X \quad Y \quad Z \quad K \quad M \quad N]^T$ is the corresponding resultant external forces and moments.

Property 3.1: The parameterisation of the rigid-body inertia matrix $M_{RB} \in \mathbb{R}^{6\times6}$ is unique and it satisfies $M_{RB} = M_{RB}^T > 0$ with its time derivative $\dot{M}_{RB} = 0$. The matrix M_{RB} is given by

$$M_{RB} = \begin{bmatrix} mI_{3\times3} & -m(r_G)^\times \\ m(r_G)^\times & I_0 \end{bmatrix}$$

where $I_{3\times3}$ is the 3×3 identity matrix and $I_0 = I_0^T > 0$ is the inertia tensor with respect to the body-fixed frame.

Property 3.2: The rigid-body Coriolis and centripetal matrix $C_{RB}(\boldsymbol{v})$ can always be parameterised such that $C_{RB}(v)$ is skew-symmetric, that is $C_{RB}(\boldsymbol{v}) = -C_{RB}(\boldsymbol{v}) \ \ \forall \ \ \boldsymbol{v} \in \mathbb{R}^6$.

3.4. Hydrodynamic Forces and Moments

The general expression of the 6-DOF dynamic equation of motion is represented in the following form [3.1]

$$(M_{RB} + M_A)\dot{\boldsymbol{v}} + \big(C_{RB}(\boldsymbol{v}) + C_A(\boldsymbol{v})\big)\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \boldsymbol{\tau}_e \qquad (3.43)$$

where subscript A represents the added mass, $D(\boldsymbol{v}) \in \mathbb{R}^{6\times6}$ is the total hydrodynamic damping matrix and $\boldsymbol{g}(\boldsymbol{\eta}) \in \mathbb{R}^6$ are the restoring forces. The external forces and moments on the right hand side of (3.43) are part of $\boldsymbol{\tau}_{RB}$; $\boldsymbol{\tau}_e \in \mathbb{R}^6$ are the environmental forces and moments, i.e. ocean current effects while $\boldsymbol{\tau} \in \mathbb{R}^6$ are the propulsion forces and moments. In the next section, the hydrodynamic effects that appear in (3.43) and their related properties are presented.

3.4.1 Added Mass and Inertia

As the robot moves underwater, additional force and moment coefficients are added to account for the effective mass of the fluid that surrounds the robot which must be accelerated with the robot. These coefficients are referred to as added (virtual) mass and include added moments of inertia and cross coupling terms such as force coefficients due to the linear and angular accelerations. The concept of fluid kinetic energy can be used to derive the added mass terms.

The expression for the fluid kinetic energy T_A can be given as follows [3.19]

$$T_A = \frac{1}{2}\boldsymbol{v}^T M_A \boldsymbol{v} \tag{3.44}$$

where $M_A \in \mathbb{R}^{6\times 6}$ is an added inertia matrix. If the rigid body is fully submerged, then the matrix M_A is always strictly positive, that is $M_A > 0$. Moreover, the added mass matrix also satisfies the following property [3.20]:

Property 3.3: For a stationary rigid body under the assumption of an ideal fluid, no incident waves, no sea currents and frequency independence, the added inertia matrix is positive definite, that is $M_A = M_A^T > 0$.

The added Coriolis and centripetal term $C_A(\boldsymbol{v})$ also contributes to the added mass where it satisfies:

Property 3.4: For a rigid body moving through an ideal fluid the hydrodynamic Coriolis and centripetal matrix $C_A(\boldsymbol{v})$ can always be parameterised such that $C_A(\boldsymbol{v})$ is a skew-symmetrical matrix, that is $C_A(\boldsymbol{v}) = -C_A^T(\boldsymbol{v}) \; \forall \;\; \boldsymbol{v} \in \mathbb{R}^6$.
Proof: See Fossen [3.1].

3.4.2 Hydrodynamic Damping Effects

The total hydrodynamic damping $D(\boldsymbol{v})$ in (3.43) consists of various damping components, i.e. potential damping, wave drift damping and damping due to vortex shedding and satisfies the following property:

Property 3.5: The hydrodynamic damping for a fully submerged body moving through an ideal fluid is a real, non-symmetrical and strictly positive matrix, that is $D(\boldsymbol{v}) > 0 \; \forall \;\; \boldsymbol{v} \in \mathbb{R}^6$.
Proof: See Fossen [3.1].

Generally, the damping of an underwater robot moving at high speed will be highly nonlinear and coupled. Assuming that a fully submerged body is performing a non-

coupled motion where the terms that are higher than second order are negligible leads to a diagonal structure expression for $D(\boldsymbol{\nu})$. Using the notation of SNAME [3.15], the diagonal damping matrix can be written as

$$D(\boldsymbol{\nu}) = -diag\{X_u, Y_v, Z_w, K_p, M_q, N_r\} \\ -diag\{X_{u|u|}|u|, Y_{v|v|}|v|, Z_{w|w|}|w|, K_{p|p|}|p|, M_{q|q|}|q|, N_{r|r|}|r|\} \quad (3.45)$$

where only linear quadratic damping terms are in the diagonal matrix and the coefficients are considered to be constant. The references [3.21, 3.22] present a detailed analysis of the dissipative forces. The rest of this section only presents the general formulation of these forces.

The two forces on a submerged body that arise due to the viscous effects are known as the drag and the lift forces. The drag forces act in parallel to the relative velocity between the submerged body and the fluid, while the lift forces are normal to it. It is difficult to exactly compute these forces because their coefficients are not known and are always determined experimentally.

As in [3.23], the drag force for a spherical body moving in a fluid can be defined as

$$f_{drag} = \frac{1}{2}\rho \boldsymbol{U}^2 S C_d(R_n) \quad (3.46)$$

where ρ is the fluid density, $\boldsymbol{U}$ is the velocity of the sphere, S is the frontal area of the sphere, C_d is the dimensionless drag coefficient and R_n is the Reynolds number. Normally, S is specified as the projection of the frontal area along the flow direction of a body. The drag coefficient $C_d(R_n)$ depends on the Reynolds number

$$R_n = \frac{\rho|\boldsymbol{U}|D}{\mu} \quad (3.47)$$

where D is the characteristic length of the body perpendicular to the direction $\boldsymbol{U}$ and μ is the dynamic viscosity of the fluid.

As reported in [3.23], the lift forces are perpendicular to the flow direction and for a wing-like structure, the lift forces can be formulated as

$$f_{lift} = \frac{1}{2}\rho \boldsymbol{U}^2 S C_l(R_n, \alpha) \tag{3.48}$$

where C_l and α denote the dimensionless lift coefficient and the desired angle of attack, respectively. Note that (3.48) is basically the same as (3.46), only the lift/drag coefficient is different. The drag and lift coefficients as a function of the Reynolds number for a cylinder have been presented by Fossen in [3.1].

3.4.3 Ocean Current Effects

The ocean current, due to tidal movement or heat exchange at the sea surface, is known as one of the environmental disturbances for autonomous underwater vehicles. Other environmental forces such as wind and waves are not discussed in this thesis. Assume that the ocean current, expressed in the inertial-fixed frame, ${}^I\boldsymbol{v}_c$, is constant and unidirectional, i.e. ${}^I\boldsymbol{v}_c = [v_{c_x} \quad v_{c_y} \quad v_{c_z} \quad 0 \quad 0 \quad 0]^T$ with ${}^I\dot{\boldsymbol{v}}_c = \boldsymbol{0}$. Its effects can be taken into account in the dynamic model of the rigid body moving in a fluid by simply considering in vehicle-fixed coordinates the relative velocity between the vehicle and the current $\boldsymbol{v}_r = \boldsymbol{v} - J_{v_a}^{-1}(\boldsymbol{\eta}_2)\ {}^I\boldsymbol{v}_c$, instead of $\boldsymbol{v}$ in the derivation of the Coriolis, centripetal and damping terms. If the ocean current is assumed unidirectional and constant in the inertial-fixed frame, then the body-fixed current disturbance $\boldsymbol{\tau}_e$ can be obtained by projecting the constant inertial-fixed current disturbance onto the body-fixed frame [3.24]. However, in some papers [3.25, 3.26], the effect of ocean current is simply modelled as a time-varying, body-fixed, disturbance that would result in an insignificant regressor for the current.

3.4.4 Restoring Forces and Moments

The last term on the left hand side of (3.43) is called the restoring forces or gravitational and buoyancy forces $\boldsymbol{g}(\boldsymbol{\eta})$. The restoring force is a function of the robot orientation $\boldsymbol{\eta}$ and the hydrostatic forces; buoyancy and gravity. Given the hydrostatic parameters; m is the mass of the robot, ∇ is the volume of fluid displaced by the robot, $\boldsymbol{g}$ is the

acceleration due to gravity (positive downwards) and ρ is the water density, then the submerged weight of an underwater robot is defined as $W = m\|\boldsymbol{g}\|$, while the buoyancy force is represented as $B = \rho\|\boldsymbol{g}\|\nabla$. The gravity and buoyancy forces can be represented in the body-fixed frame as

$$\boldsymbol{f}_G(\boldsymbol{\eta}_2) = J_1^T(\boldsymbol{\eta}_2)[0 \quad 0 \quad W]^T \tag{3.49}$$

$$\boldsymbol{f}_B(\boldsymbol{\eta}_2) = -J_1^T(\boldsymbol{\eta}_2)[0 \quad 0 \quad B]^T \tag{3.50}$$

The weight and buoyancy forces can be used along with the centre of gravity $\boldsymbol{r}_G = [x_G \quad y_G \quad z_G]^T$ and the centre of buoyancy $\boldsymbol{r}_B = [x_B \quad y_B \quad z_B]^T$ expressed in the body-fixed frame to obtain the restoring forces and moments vector $\boldsymbol{g}(\boldsymbol{\eta})$

$$\boldsymbol{g}(\boldsymbol{\eta}) = -\begin{bmatrix} (W-B)s\theta \\ -(W-B)c\theta s\phi \\ -(W-B)c\theta c\phi \\ -(y_G W - y_B B)c\theta c\phi + (z_G W - z_B B)c\theta s\phi \\ (z_G W - z_B B)s\theta + (x_G W - x_B B)c\theta c\phi \\ -(x_G W - x_B B)c\theta s\phi - (y_G W - y_B B)s\theta \end{bmatrix} \tag{3.51}$$

If the quaternion representation is used, then $\boldsymbol{g}(\boldsymbol{\epsilon})$ can be written as

$$\boldsymbol{g}(\boldsymbol{\epsilon}) = \begin{bmatrix} 2(\epsilon_0\epsilon_2 - \epsilon_1\epsilon_3)(W-B) \\ -2(\epsilon_0\epsilon_1 + \epsilon_2\epsilon_3)(W-B) \\ (-\epsilon_0^2 + \epsilon_1^2 + \epsilon_2^2 - \epsilon_3^2)(W-B) \\ (-\epsilon_0^2 + \epsilon_1^2 + \epsilon_2^2 - \epsilon_3^2)(y_G W - y_B B) + 2(\epsilon_0\epsilon_1 + \epsilon_2\epsilon_3)(z_G W - z_B B) \\ (\epsilon_0^2 - \epsilon_1^2 - \epsilon_2^2 + \epsilon_3^2)(x_G W - x_B B) + 2(\epsilon_0\epsilon_2 - \epsilon_1\epsilon_3)(z_G W - z_B B) \\ -2(\epsilon_0\epsilon_1 + \epsilon_2\epsilon_3)(x_G W - x_B B) - 2(\epsilon_0\epsilon_2 - \epsilon_1\epsilon_3)(y_G W - y_B B) \end{bmatrix} \tag{3.52}$$

It can be observed from (3.51) that the force $(W-B)$ and the moments Wr_G and Wr_B are known as constant system parameters.

Therefore, the restoring forces on the left-hand side of (3.43) can be written as [3.27]

$$\boldsymbol{g}(\boldsymbol{\eta}) = Z(\boldsymbol{\eta})\Phi \tag{3.53}$$

where $\Phi \in \mathbb{R}^{n_p}$ is a set of parameters and $Z(\boldsymbol{\eta}) \in \mathbb{R}^{n \times n_p}$ is the gravity regression matrix which represents the known part of $\boldsymbol{g}(\boldsymbol{\eta})$; n_p is the total number of physical parameters.

3.5. Equation of Motion for an Underwater Vehicle

As given in (3.43), the nonlinear equation of motion in a body-fixed frame can be written as

$$M_v\dot{\boldsymbol{v}} + C_v(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \boldsymbol{\tau}_e \tag{3.54}$$

where (3.54) considers the generalised forces, the hydrodynamic effects, the restoring forces and the current effects. Matrices $M_v = M_{RB} + M_A$ and $C_v(\boldsymbol{v}) = C_{RB}(\boldsymbol{v}) + C_A(\boldsymbol{v})$ have been defined in the previous section.

Next, some simple but fundamental properties of the dynamic model for underwater robots are presented. In spite of the complexity of the dynamic equation of motion (3.54), which describes the behaviour of underwater robots, this motion equation and the terms which constitute it have interesting properties for the control strategies. These properties are of particular importance in the study of control systems for underwater robots. Properties that are relevant to control design and stability analysis via Lyapunov's direct method (see Section A.2.4 in Appendix A) are presented. Further proofs are presented in [3.1, 3.14]. These properties, which are extensively used in the following chapters, can be described as follows:

Property 3.6: The inertia matrix M_v for a rigid body is symmetric and strictly positive definite, that is $M_v = M_v^T > 0$.

Proof: $M_v = M_{RB} + M_A$ is symmetric and positive definite under the assumptions that M_{RB} and M_A are both symmetric and positive definite matrices. ■

Property 3.7: The Coriolis and centripetal forces matrix $C_v(\boldsymbol{v})$ is skew-symmetric, that is $C_v(\boldsymbol{v}) = -C_v^T(\boldsymbol{v}) \ \forall \ \boldsymbol{v} \in \mathbb{R}^6$.

Proof: $C_v(\boldsymbol{v})$ is a skew-symmetric matrix under the assumptions that $C_{RB}(\boldsymbol{v})$ and $C_A(\boldsymbol{v})$ are both skew-symmetric. ■

3.6. Kinematics of Onboard Manipulators

For an n-link onboard manipulator, the joint position state vector is defined by $\boldsymbol{q} = [q_1 \ q_2 \ \cdots q_n]^T \in \mathbb{R}^n$ and the end-effector composition vector is described by $\boldsymbol{\zeta}_{v,m} = [\ {}^v\boldsymbol{p}_{v,m}^T \quad {}^v\boldsymbol{\epsilon}_{v,m}^T]^T \in \mathbb{R}^7$, where ${}^v\boldsymbol{p}_{v,m} \in \mathbb{R}^3$ and ${}^v\boldsymbol{\epsilon}_{v,m} \in \mathbb{R}^4$ are the position and unit quaternion orientation representations respectively. The superscript v indicates that the vectors are expressed in the body-fixed frame. The relationship between the body-fixed manipulator velocity $\dot{\boldsymbol{\zeta}}_{v,m}$ and joint velocity $\dot{\boldsymbol{q}}$ can be represented using the analytical Jacobian of the manipulator as [3.28]

$$\begin{bmatrix} {}^v\dot{\boldsymbol{p}}_{v,m} \\ {}^v\dot{\boldsymbol{\epsilon}}_{v,m} \end{bmatrix} = \begin{bmatrix} J_{mp}(\boldsymbol{q}) \\ J_{mo}(\boldsymbol{q}) \end{bmatrix} \dot{\boldsymbol{q}} \Leftrightarrow \dot{\boldsymbol{\zeta}}_{v,m} = J_m'(\boldsymbol{q})\, \dot{\boldsymbol{q}} \tag{3.55}$$

where $J_{mp} \in \mathbb{R}^{3\times n}$ and $J_{mo} \in \mathbb{R}^{4\times n}$ denote the position and orientation Jacobian matrices from the manipulator base to end-effector, respectively. The end-effector angular velocity expressed in the manipulator base frame ${}^v\boldsymbol{\omega}_{v,m}$ is related to ${}^v\dot{\boldsymbol{\epsilon}}_{v,m}$ through (3.14). Applying (3.16) and (3.55) into (3.14) means that the following expression relating the generalised end-effector velocity vector can be obtained

$$\begin{bmatrix} {}^v\dot{\boldsymbol{p}}_{v,m} \\ {}^v\boldsymbol{\omega}_{v,m} \end{bmatrix} = J_M\big({}^v\boldsymbol{\epsilon}_{v,m}, \boldsymbol{q}\big)\dot{\boldsymbol{q}} \tag{3.56}$$

where $J_M\big({}^v\boldsymbol{\epsilon}_{v,m}, \boldsymbol{q}\big) \in \mathbb{R}^{6\times n}$ is the end-effector Jacobian matrix and is explicitly given as follows

$$J_M\big({}^v\boldsymbol{\epsilon}_{v,m}, \boldsymbol{q}\big) \triangleq \begin{bmatrix} J_{mp}(\boldsymbol{q}) \\ 2E^T\big({}^v\boldsymbol{\epsilon}_{v,m}\big)J_{mo}(\boldsymbol{q}) \end{bmatrix} \tag{3.57}$$

where E is defined as in (3.15).

In the following expressions, a redundant manipulator is considered (i.e. $n > 6$), thus the inverse kinematics at the velocity level can be obtained using (3.56) as follows

$$\dot{\boldsymbol{q}} = J_M^{\dagger}\boldsymbol{v}_M + \left(I_n - J_M^{\dagger}J_M\right)\boldsymbol{z} \tag{3.58}$$

where I_n denotes the $n \times n$ identity matrix, $\left(I_n - J_M^{\dagger}J_M\right)$ is the projection matrix into the null space of J_M. $\left(I_n - J_M^{\dagger}J_M\right)z$ is the homogeneous solution of (3.58) orthogonal to its particular solution, $J_M^{\dagger}v_M$. $J_M^{\dagger} \in \mathbb{R}^{n\times m}$ is the Moore-Penrose pseudo-inverse of the manipulator Jacobian and is defined as

$$J_M^{\dagger} = J_M^T(J_M J_M^T)^{-1} \tag{3.59}$$

which is also termed as the right pseudo-inverse since $J_M J_M^{\dagger} = I_6$. From (3.58), the vector $\boldsymbol{z} \in \mathbb{R}^n$ denotes an auxiliary velocity which can be constructed to improve the performance of the manipulator according to an additional control objective (e.g. manipulability measure). This possible performance enhancement is achieved by optimising a proper performance criterion function, instead of $\boldsymbol{z}$. Let $H(\boldsymbol{q}) \in \mathbb{R}$ be the optimal positive function, then $\boldsymbol{z}$ is defined as

$$\boldsymbol{z} = \kappa\nabla H(\boldsymbol{q}) \tag{3.60}$$

where $\nabla H(\boldsymbol{q})$ is the gradient of $H(\boldsymbol{q})$ and κ is a real valued scalar. Note that, the pseudo-inverse defined by (3.59) satisfies the Moore-Penrose conditions and the properties of the null space matrix $\left(I_n - J_M^{\dagger}J_M\right)$ [3.29].

Remark 3.2: During the subsequent control development, the assumption is made that the minimum singular value of the onboard manipulator Jacobian matrix, denoted by σ_m, is greater than a known small positive constant $\delta > 0$, such that $\max\{\|J_M^{\dagger}(\boldsymbol{q})\|\}$ is known *a priori* and all kinematic singularities are always avoided.

3.7. Equations of Motion of an Underwater Vehicle-Manipulator System

The dynamic equations of motion for a manipulator mounted on a mobile platform were analysed in detail in [3.30]. In [3.11], a similar recursive concept was utilised, where the hydrodynamic effects (added mass, drag, lift and buoyancy) were included [3.1, 3.10, 3.31].

In order to obtain a complete model for the system, the balance forces and moments acting on the serial chain body can be described in the following forms:

$$F_i^i = M_i\left[\boldsymbol{a}_i^i + \dot{\boldsymbol{\omega}}_i^i \times \boldsymbol{r}_{i,c}^i + \boldsymbol{\omega}_i^i \times \left(\boldsymbol{\omega}_i^i \times \boldsymbol{r}_{i,c}^i\right)\right] \tag{3.61}$$

$$T_i^i = I_i^i \dot{\boldsymbol{\omega}}_i^i + \boldsymbol{\omega}_i^i \times \left(I_i^i \boldsymbol{\omega}_i^i\right) \tag{3.62}$$

where F_i^i are the total forces acting at the centre of mass of link i, T_i^i are the total moments acting about the centre of mass of link i, M_i is the mass and added mass matrix, I_i^i is the inertia matrix plus added inertia with respect to the centre of mass and $\boldsymbol{r}_{i,c}^i$ is the vector from the origin of frame i towards the centre of mass of link i expressed in frame i. The variable $\boldsymbol{\omega}_i^i$ denotes the angular velocity of the frame i, $\dot{\boldsymbol{\omega}}_i^i$ is the angular acceleration of the frame i and $\boldsymbol{a}_i^i$ is linear acceleration of the origin of frame i. Schjølberg and Fossen in [3.11] give further details of the total forces and moments acting on the general body of a serial chain.

Consider the velocity vector as $\boldsymbol{\xi} = [\boldsymbol{v} \quad \dot{\boldsymbol{q}}]^T$, thus the equation of motion of the underwater vehicle with an onboard manipulator is given in the body-fixed frame [3.11, 3.14]

$$M(\boldsymbol{q})\dot{\boldsymbol{\xi}} + C(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + D(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + \boldsymbol{g}(\boldsymbol{\eta}_e,\boldsymbol{q}) = \boldsymbol{\tau} \tag{3.63}$$

where $M(\boldsymbol{q}) \in \mathbb{R}^{(6+n)\times(6+n)}$ is the inertia matrix including added mass, $C(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} \in \mathbb{R}^{(6+n)}$ is the vector of Coriolis and centripetal terms, $D(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} \in \mathbb{R}^{(6+n)}$ is the vector of hydrodynamic damping, $\boldsymbol{g}(\boldsymbol{\eta}_e,\boldsymbol{q}) \in \mathbb{R}^{(6+n)}$ is the vector of gravity and buoyancy forces, $\boldsymbol{\tau} \in \mathbb{R}^{(6+n)}$ is the vector of generalised forces acting on the vehicle and joint

torques. The properties of the dynamic equation described by (3.63) are given as follows (see [3.11] for proofs):

Property 3.8: The inertia matrix $M(\boldsymbol{q})$ is positive definite due to positive kinetic energy, whereas symmetry is guaranteed by applying Newton's third law (action-reaction principle) such that $M(\boldsymbol{q}) = M^T(\boldsymbol{q}) > 0$.

Property 3.9: $\dot{M}(\boldsymbol{q}) - 2C(\boldsymbol{q}, \boldsymbol{\xi})$ is skew-symmetric.

Property 3.10: The hydrodynamic damping matrix $D(\boldsymbol{q}, \boldsymbol{\xi})$ is positive definite due to the dissipative nature of the underwater vehicle-manipulator system, i.e. $D(\boldsymbol{q}, \boldsymbol{\xi}) = D^T(\boldsymbol{q}, \boldsymbol{\xi}) > 0$.

The UVMS dynamic model as described in (3.63) is linear in a set of dynamic parameters $\boldsymbol{\Phi} \in \mathbb{R}^{n_p}$ and can be written as

$$M(\boldsymbol{q})\dot{\boldsymbol{\xi}} + C(\boldsymbol{q}, \boldsymbol{\xi})\boldsymbol{\xi} + D(\boldsymbol{q}, \boldsymbol{\xi})\boldsymbol{\xi} + \boldsymbol{g}(\boldsymbol{\eta}_e, \boldsymbol{q}) = Y\left(\boldsymbol{q}, \boldsymbol{\eta}_e, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}\right)\boldsymbol{\Phi} \tag{3.64}$$

where $Y\left(\boldsymbol{q}, \boldsymbol{\eta}_e, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}\right) \in \mathbb{R}^{(6+n)\times n_p}$ is the UVMS regression matrix; n_p is the total number of physical parameters. It is assumed that if the arguments of $Y(\cdot)$ are bounded then $Y(\cdot)$ is bounded.

3.8. Summary

The equations of motion for an underwater vehicle (3.54) and underwater vehicle-manipulator system (3.63) along with the properties of 3.6, 3.7, 3.8, 3.9 and 3.10 are extensively used in the next chapters for the stability analysis of the proposed regulation and tracking control schemes. In particular, Property 3.5 is used to construct non-negative functions and occasionally Lyapunov functions are used to study the stability and convergence properties for equilibrium in underwater vehicle control systems.

3.9. References

[3.1] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, 1st ed. New York: John Wiley and Sons, 1994.

[3.2] J. Yuh, "Modeling and Control of Underwater Robotic Vehicles," *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 20, no. 6, pp. 1475–1483, 1990.

[3.3] J. Yuh, "Design and Control of Autonomous Underwater Robots: A survey," *Autonomous Robots*, vol. 8, no. 1, pp. 7-24, 2000.

[3.4] J. Yuh and M. West, "Underwater Robotics," *International Journal of Advanced Robotics*, vol. 15, no. 5, pp. 609–639, 2001.

[3.5] M. Mahesh, J. Yuh and R. Lakshmi, "A Coordinated Control of an Underwater Vehicle and Robotic Manipulator," *Journal of Robotic Systems*, vol. 8, no. 3, pp. 339-370, 1991.

[3.6] S. McMillan, D. E. Orin and R. B. McGhee, "DynaMechs: An Object Oriented Software Package for Efficient Dynamic Simulation of URVs," in *Underwater Robotic Vehicles: Design and Control*. Albuquerque, NM: TSI Press, 1995, pp. 73–98.

[3.7] C. C. Cheah and Y. C. Sun , "Coordinated Control of Multiple Cooperative Underwater Vehicle-Manipulator Systems Holding a Common Load," in *Techno-Ocean'04*, Kobe, Japan, 2004, pp. 1542-1547.

[3.8] K. Ioi and K. Itoh, "Modeling and Simulation of an Underwater Manipulator," *Advanced Robotics*, vol. 4, no. 4, pp. 303-317, 1990.

[3.9] H. Janocha and I. Papadimitriou, "Simulation of the Dynamic Behavior of Robots in an Extreme Environment," *Robotics and Computer-Integrated Manufacturing*, vol. 8, no. 3, pp. 163-169, 1991.

[3.10] S. McMillan, D. E. Orin and R. B. McGhee, "Efficient Dynamic Simulation of an Underwater Vehicle with a Robotic Manipulator," *IEEE Transactions On Systems, Man and Cybernetics*, vol. 25, pp. 1194-1206, August 1995.

[3.11] I. Schjølberg and T. I. Fossen, "Modelling and Control of Underwater Vehicle-Manipulator Systems," in *Proc. 3rd Conf. Marine Craft Manoeuvring and Control*, Southampton, U.K., 1994, pp. 45–57.

[3.12] O. E. Fjellstad and T. I. Fossen, "Position and attitude tracking of AUV's: A quaternion feedback approach," *Journal of Oceanic Engineering*, vol. 19, no. 4, pp. 512 - 518, 1994.

[3.13] F. Lizarralde, J. T. Wen and L. Hsu, "Quaternion-Based Coordinated Control of a Subsea Mobile Manipulator with only Position Measurements," in *Proc. of the 34th IEEE Conference on Decision and Control*, vol. 4, New Orleans, LA, 1995, pp. 3996-4001.

[3.14] G. Antonelli, *Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems*. Germany: Springer-Verlag Berlin, 2003.

[3.15] SNAME, The Society of Naval Architects and Marine Engineers, Nomenclature for Treating the Motion of a Submerged Body Through a Fluid, In: Technical and Research Bulletin, no. 1-5, 1950, pp. 1-15.

[3.16] F. Lizarralde and J. T. Wen, "Attitude Control Without Angular Velocity Measurement: A Passivity Approach," *IEEE Transactions on Automatic Control*, vol. 41, pp. 468-472, 1996.

[3.17] J. S. C. Yuan, "Closed-Loop Manipulator Control Using Quaternion Feedback," *IEEE Transactions on Robotics and Automation*, vol. 4, pp. 434-440, 1988.

[3.18] S. W. Shepperd, "Quaternion from Rotation Matrix," *Journal of Guidance and Control*, vol. 1, pp. 223-224, 1978.

[3.19] H. Lamb, *Hydrodynamics*. London : Cambridge University Press , 1932.

[3.20] J. N. Newman, *Marine Hydrodynamics*. MA: MIT Press Cambridge, 1977.

[3.21] B. Steven and F. Lewis, *Aircraft Control and Simulations*. Chichester, United Kingdom: John Wiley & Sons Ltd., 1992.

[3.22] T. Sarpkaya and M. Isaacson, *Mechanics of Wave Forces on Offshore Structures*. New York: Van Nostrand Reinhold Co., 1981.

[3.23] G. Indiveri, *Modelling and Identification of Underwater Robotic Systems*.: Ph.D. Thesis in Electronic Engineering and Computer Science, University of Genova, Italy, 1998.

[3.24] G. Antonelli, "On the Use of Adaptive/Intergral Actions for 6-Degrees-of-Freedom Control of Autonomous Underwater Vehicles," *Journal of Oceanic Engineering*, vol. 32, no. 2, pp. 300-312, 2007.

[3.25] T. I. Fossen and J. Balchen, "The NEROV Autonomous Underwater Vehicle ," in *MTS/IEEE Techno-Ocean'91 Conference*, Honolulu, Hawaii, 1991.

[3.26] G. Antonelli, S. Chiaverini, N. Sarkar and M. West, "Adaptive Control of an Autonomous Underwater Vehicle: Experimental Results on ODIN," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 5, pp. 756 - 765, 2001.

[3.27] S. Arimoto , *Control Theory of Nonlinear Mechanical Systems - A Passivity-Based and Circuit-Theoretic Approach*.: Oxford: Clarendon Press, 1996.

[3.28] L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators (Advanced Textbooks in Control and Signal Processing)*, 2nd ed.: Springer, January 2005.

[3.29] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Reading, MA: Addison-Wesley, 1991.

[3.30] S. Murphy, J. Wen and G. Saridis, "Simulation of Cooperating Robot Manipulators on a Mobile Platform," *IEEE Transaction on Robotics and Automation*, vol. 7, pp. 468-478, 1991.

[3.31] J. P. V. S. Cunha, R. Dominguez, R. Costa and L. Hsu, "Design of a New High Performance VS Position Control of ROVs," in *Proceedings of the International Conferences on Offshore and Polar Engineering*, Edinburgh, UK, 1991.

CHAPTER 4
TASK-SPACE REGULATION CONTROL SCHEMES FOR UNDERWATER ROBOTIC SYSTEMS

4.1. Introduction

As discussed in Chapter 2, the regulation control objective for underwater robots can be achieved if the control command satisfies the following procedure. Firstly, recalling the dynamic model of an underwater vehicle defined in (3.54)

$$M_v\dot{\boldsymbol{v}} + C_v(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \tag{4.1}$$

where M_v is the inertia matrix including the added mass term, $C_v(\boldsymbol{v})$ represents the matrix of the Coriolis and centripetal forces including the added mass term, $D(\boldsymbol{v})$ denotes the hydrodynamic damping and lift force, and $\boldsymbol{g}(\boldsymbol{\eta})$ is the restoring force. The vectors $\boldsymbol{\eta}$, $\boldsymbol{v}$, $\dot{\boldsymbol{v}}$ denote the position/orientation, velocity and acceleration respectively. In (4.1), the environmental forces and moments are omitted.

Then, taking into account (4.1), the objective of regulation control consists in finding a control command $\boldsymbol{\tau}$ such that

$$\lim_{t\to\infty} \boldsymbol{\eta}(t) = \boldsymbol{\eta}_d \tag{4.2}$$

where the constant vector $\boldsymbol{\eta}_d \in \mathbb{R}^6$ is a desired position and orientation.

The performance of the regulation controllers are evaluated using the Lyapunov-type function where an analysis of the asymptotic stability of the origin of the closed-loop system is performed (*cf.* Appendix A). Therefore, the regulation control objective of (4.2) can be conveniently rewritten as

$$\lim_{t\to\infty} \tilde{\boldsymbol{e}}(t) = 0 \tag{4.3}$$

where $\tilde{\boldsymbol{e}} \coloneqq \boldsymbol{\eta}_d - \boldsymbol{\eta}(t) \in \mathbb{R}^6$ represents the vector of position and orientation errors. From (4.3), it can be said that the control objective is achieved, if for instance the origin of the closed-loop system is asymptotically stable.

In this chapter, a number of novel regulation control laws are presented for two types of underwater robotic systems, namely autonomous underwater vehicles and underwater vehicle-manipulator systems. At the beginning of the chapter, an overview of the conventional set-point and region reaching controllers are presented. Then, a sub-region priority reaching controller is proposed for an autonomous underwater vehicle. It also covers the method of region-decomposition which is employed to specify the particular sub-regions as motion constraints.

Subsequently, an adaptive-fuzzy sub-region priority reaching controller is formulated to deal with the uncertainties of the restoring forces and to manage the multiple sub-regions effectively. Several new control structures based on the region boundary are also presented which act as an alternative approach to control the underwater robot to reach the desired region boundary rather than into a region or a point. In addition, the least-squares estimation algorithm is utilised in the control law which is a novel departure from existing adaptive control approaches for underwater robots. Note that a fully actuated 6-DOF underwater system is used throughout this chapter.

4.2. Conventional Set-Point and Region Reaching Control Approaches

In the scope of regulation control schemes, the development of set-point controllers has been extensively studied for underwater robotic systems as reported in [4.1, 4.2, 4.3, 4.4]. Most of these developed controllers are inspired from the regulation control techniques of fixed-base manipulators. Moreover, they are known as “conventional” controllers as they are commonly used in industrial robots. As illustrated in Figure 4.1(a), the desired objective is defined by a point where the position error and potential energy are both zero.

The simplest global set-point control law, a proportional-derivative control plus gravity and buoyancy compensation, was proposed by Takegaki and Arimoto [4.5]. However, due to its requirement for exact knowledge of gravity and buoyancy forces, this controller is not suitable to be used in the underwater environment where the parameters

of these forces are difficult to obtain accurately. An adaptive control approach is known as one of the effective ways to deal with the parameter uncertainties of AUVs. As reported in previous research works [4.6], [4.7] and [4.8], the compensation of the unknown and time-varying system parameters were achieved using regressor-based adaptive control schemes with parameter estimation. However, the regressor-based approach which relies on the knowledge of the system to be controlled requires a large computational time as the number of dynamic parameters for estimation is significant. Therefore, an adaptive law with a gravity regressor term was proposed by Sun and Cheah [4.3, 4.9] to significantly reduce the computational burden of the controller. In the adaptive underwater robot control literature, researchers typically prefer to utilise gradient-type algorithms for parameter estimation due to their simplicity. It should be noted that the gradient-type update laws often exhibits slow parameter convergence [4.10], hence an adaptive control methodology that provides flexibility in the design of the parameter estimation update law is highly desired.

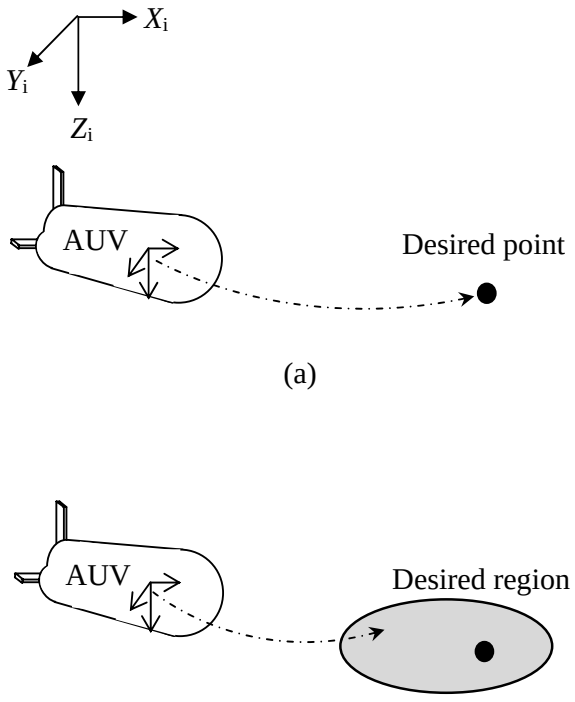


Figure 4.1: (a) Conventional set-point control (b) Region reaching control

Alternatively, the control objective of underwater robots could be defined as a region rather than a point [4.9, 4.11], as shown in Figure 4.1(b). For example, maintaining the underwater vehicle within a minimum and maximum depth in water; an underwater vehicle travelling inside a pipeline for a specific task; avoiding an obstacle located at a

specific region or an underwater region constraint to ensure visibility during motion. To illustrate region reaching control, define the desired region as a single objective function as follows

$$f(\delta\boldsymbol{\eta}) \leq 0 \tag{4.4}$$

where $\delta\boldsymbol{\eta} = \boldsymbol{\eta} - \boldsymbol{\eta}_d \in \mathbb{R}^6$, $\boldsymbol{\eta}$ is previously defined in Chapter 3 and $\boldsymbol{\eta}_d$ represents the reference point of the desired region and $f(\delta\boldsymbol{\eta}) \in \mathbb{R}$ is the scalar function with continuous partial derivatives. For instance, the desired region can be specified as a sphere using the following function

$$f(\delta\boldsymbol{\eta}_1) = (x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2 - r^2 \leq 0 \tag{4.5}$$

where the subscript d denotes the desired value and r is the radius of the sphere. The potential energy function for the desired region described in inequality (4.4) can be defined as

$$P(\delta\boldsymbol{\eta}) = \begin{cases} 0, & f(\delta\boldsymbol{\eta}) \leq 0, \\ \dfrac{k_p}{2} f^2(\delta\boldsymbol{\eta}), & f(\delta\boldsymbol{\eta}) > 0, \end{cases} \tag{4.6}$$

where k_p is a positive constant. Note that [4.9] briefly discussed the potential energy function. Partial differentiation of the potential energy function described by (4.6) with respect to $\boldsymbol{\eta}$ yields

$$\left(\frac{\partial P(\delta\boldsymbol{\eta})}{\partial(\delta\boldsymbol{\eta})}\right)^T = k_p \max\big(0, f(\delta\boldsymbol{\eta})\big)\left(\frac{\partial f(\delta\boldsymbol{\eta})}{\partial\boldsymbol{\eta}}\right)^T \tag{4.7}$$

Considering an underwater vehicle as in (4.1), the vector of (4.7) can be utilised in the region reaching control with exact restoring force compensation as follows

$$\boldsymbol{\tau} = K_v \dot{\tilde{\boldsymbol{e}}}_x + K_p \max\big(0, f(\delta\boldsymbol{\eta})\big)\left(\frac{\partial f(\delta\boldsymbol{\eta})}{\partial\boldsymbol{\eta}}\right)^T + \boldsymbol{g}(\boldsymbol{\eta}) \tag{4.8}$$

where $\dot{\tilde{\boldsymbol{e}}}_x$ is the velocity error, K_v is the velocity gain matrix and K_p is a matrix that consists of position and orientation gains and the transpose of the Jacobian matrix.

According to Theorem A.2 and La Salle's theorem (*cf.* Appendix A), this control guarantees the convergence of $\boldsymbol{\eta}$ into the desired region $f(\delta\boldsymbol{\eta}) \leq 0$ as $t \to \infty$.

With regards to the set-point control problem from the region control point of view, the set-point controller is known as a special case of the region reaching controller when the desired region is specified to be arbitrarily small or any closed region with an arbitrary shape reduces to zero. Given the nonnegative scalar κ_{xyz}, the desired region for the position can be specified as follows

$$f(\delta\boldsymbol{\eta}_1) = ((x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2)^{\frac{1}{2}} - \kappa_{xyz} \leq 0 \tag{4.9}$$

where $\delta\boldsymbol{\eta}_1 = [x - x_d \quad y - y_d \quad z - z_d]^T$. Thus,

$$\left(\frac{\partial f(\delta\boldsymbol{\eta}_1)}{\partial \boldsymbol{\eta}_1}\right)^T = \frac{1}{((x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2)^{\frac{1}{2}}}[\delta\boldsymbol{\eta}_1] \tag{4.10}$$

If κ_{xyz} reduces to zero, then $f(\delta\boldsymbol{\eta}_1)$ is nonnegative and hence

$$\max\big(0, f(\delta\boldsymbol{\eta}_1)\big)\left(\frac{\partial f(\delta\boldsymbol{\eta}_1)}{\partial \boldsymbol{\eta}_1}\right)^T = f(\delta\boldsymbol{\eta}_1)\left(\frac{\partial f(\delta\boldsymbol{\eta}_1)}{\partial \boldsymbol{\eta}_1}\right)^T = \delta\boldsymbol{\eta}_1 \tag{4.11}$$

where (4.11) represents the position error terms for the set-point controller.

4.3. Sub-Region Priority Reaching Controller for a 6-DOF AUV

A region reaching control scheme with a motion constraint was recently proposed by Cheah [4.12] for a robot manipulator. In this concept, the control objective is available to perform at least one region objective. When the robot is subject to additional region objectives, the controller executes it as a robot motion constraint. Moreover, it allows the specification of a primary region objective which is fulfilled with a higher priority with respect to a secondary region. It is interesting to note that for the non-redundant robot the primary and secondary region objectives overlap each other as presented in [4.12].

4.3.1. Region-Decomposition Method

To achieve effective point-to-point motion of an underwater vehicle, a sub-region priority reaching control scheme with an exact restoring force compensation technique is considered. Within this framework, an operational space can be broken down into several sub-regions with a priority order using the region-decomposition approach. For a redundant robot, the reaching task into each sub-region is performed using the degrees-of-freedom that remain after all the sub-regions with a higher priority have been implemented. In other words, the region reaching control problem for a redundant robot can be approached via the sub-region method by regarding a target region to be reached by a redundant system as the sub-region with first priority and using the redundancy to perform motion constraints defined by an additional sub-region as the sub-region with second priority.

In the case of a non-redundant system where the Jacobian matrix (in the relation between the task-space velocity and the derivative of the generalised coordinates) is an identity matrix, the primary sub-region overlaps with the secondary sub-region to form the intersection of the two sub-regions. Before the vehicle converges to an intersection of the sub-region, the vehicle must first execute the secondary sub-region objectives which are assigned along the pathway. At the final position, the vehicle converges into a region where both primary and secondary sub-region objectives are achieved simultaneously. Figure 4.2 depicts the underwater vehicle being kept within a particular region in order to maintain its depth prior to the convergence to the primary sub-region.

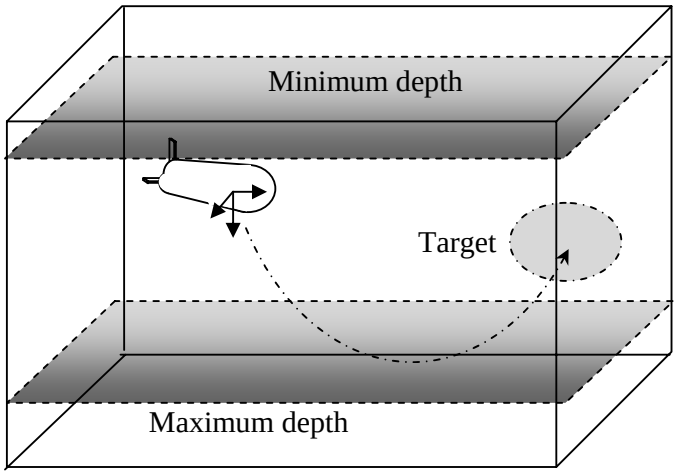


Figure 4.2: Sub-region priority reaching control for an underwater vehicle

4.3.2. Problem Formulation

To formulate the regulation control objective for an AUV, two desired sub-regions, namely attractive and repulsive regions are defined based on the region decomposition method. The attractive sub-region acts as a primary target with a higher priority than the repulsive sub-region. Using a similar definition as in (4.4), the primary sub-region can be formulated as

$$f_1(\delta \boldsymbol{p}_v) \leq 0 \tag{4.12}$$

where $\delta \boldsymbol{p}_v = \boldsymbol{p}_v - \boldsymbol{p}_{vd} \in \mathbb{R}^3$ are the continuous first partial derivatives of the primary region; $\boldsymbol{p}_{vd}$ is the stationary reference point inside the region such that $\dot{\boldsymbol{p}}_{vd}(t) = 0$. Note that this section and the rest of the thesis use the notation $\boldsymbol{p}_v$ instead of $\boldsymbol{\eta}_1$ to represent the vehicle position because the position of an onboard manipulator also shares a similar notation. Subsequently, a region with an obstacle or repulsive sub-region is defined as a second objective function given by

$$f_2\big(\delta \boldsymbol{p}_{v_{obs}}\big) \geq 0 \tag{4.13}$$

where $\delta \boldsymbol{p}_{v_{obs}} = \boldsymbol{p}_v - \boldsymbol{p}_{v_{obs}} \in \mathbb{R}^3$; $\boldsymbol{p}_{v_{obs}}$ is the stationary position of the obstacle with $\dot{\boldsymbol{p}}_{v_{obs}}(t) = 0$. The resultant potential energy functions for (4.12) and (4.13) are given by

$$P_1(\delta \boldsymbol{p}_v) = \frac{k_1}{2}\big[\max\big(0, f_1(\delta \boldsymbol{p}_v)\big)\big]^2 \triangleq \begin{cases} 0, & f_1(\delta \boldsymbol{p}_v) \leq 0 \\ \dfrac{k_1}{2} f_1^2(\delta \boldsymbol{p}_v), & f_1(\delta \boldsymbol{p}_v) > 0 \end{cases} \tag{4.14}$$

$$\begin{aligned} P_2\big(\delta \boldsymbol{p}_{v_{obs}}\big) &= \frac{k_2}{2}\Big[\max\Big(0, f_2\big(\delta \boldsymbol{p}_{v_{obs}}\big)\Big)\Big]^2 \\ &\triangleq \begin{cases} 0, & f_2\big(\delta \boldsymbol{p}_{v_{obs}}\big) \geq 0 \\ \dfrac{k_2}{2} f_2^2\big(\delta \boldsymbol{p}_{v_{obs}}\big), & f_2\big(\delta \boldsymbol{p}_{v_{obs}}\big) < 0 \end{cases} \end{aligned} \tag{4.15}$$

where k_1 and k_2 are positive scalars. Next, differentiating (4.14) with respect to $\delta \boldsymbol{p}_v$ and (4.15) with respect to $\delta \boldsymbol{p}_{v_{obs}}$ leads to

$$\left(\frac{\partial P_1(\delta \boldsymbol{p}_v)}{\partial(\delta \boldsymbol{p}_v)}\right)^T = k_1 \max\left(0, f_1(\delta \boldsymbol{p}_v)\right)\left(\frac{\partial f_1(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \tag{4.16}$$

$$\left(\frac{\partial P_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial\left(\delta \boldsymbol{p}_{v_{obs}}\right)}\right)^T = k_2 \max\left(0, f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)\right)\left(\frac{\partial f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial \boldsymbol{p}_{v_{obs}}}\right)^T \tag{4.17}$$

If the sub-region is defined within the position framework only, then the sub-region priority controller with exact gravity and buoyancy force compensation is proposed as

$$\boldsymbol{\tau} = -J_{v_a}^T(\boldsymbol{\eta}_2)\left[\begin{matrix}\left(\frac{\partial P_1(\delta \boldsymbol{p}_v)}{\partial(\delta \boldsymbol{p}_v)}\right)^T + \left(\frac{\partial P_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial\left(\delta \boldsymbol{p}_{v_{obs}}\right)}\right)^T \\ k_{\eta_2}\tilde{\boldsymbol{e}}_{\eta_2}\end{matrix}\right] - K_v \boldsymbol{v} + g(\boldsymbol{\eta}_2) \tag{4.18}$$

where $J_{v_a}^T(\boldsymbol{\eta}_2)$ is the kinematic transformation matrix defined in Chapter 3, k_{η_2} is a positive constant and $\tilde{\boldsymbol{e}}_{\eta_2} = \boldsymbol{\eta}_2 - \boldsymbol{\eta}_{2_d} \in \mathbb{R}^3$; $\boldsymbol{\eta}_{2_d}$ is the desired vehicle orientation, $K_v \in \mathbb{R}^6$ is a positive definite matrix. After substituting (4.18) into (4.1), the closed-loop equation can be obtained as follows

$$\begin{aligned} &M_v \dot{\boldsymbol{v}} + C_v(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) \\ &\qquad = -J_{v_a}^T(\boldsymbol{\eta}_2)\left[\begin{matrix}\left(\frac{\partial P_1(\delta \boldsymbol{p}_v)}{\partial(\delta \boldsymbol{p}_v)}\right)^T + \left(\frac{\partial P_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial\left(\delta \boldsymbol{p}_{v_{obs}}\right)}\right)^T \\ k_{\eta_2}\tilde{\boldsymbol{e}}_{\eta_2}\end{matrix}\right] - K_v \boldsymbol{v} + g(\boldsymbol{\eta}_2) \end{aligned} \tag{4.19}$$

which represents a nonlinear autonomous differential equation whose origin of the state-space vector is an equilibrium point. Next, define the Lyapunov function candidate in the following form

$$V = \frac{1}{2}\boldsymbol{v}^T M_v \boldsymbol{v} + P_1(\delta \boldsymbol{p}_v) + P_2\left(\delta \boldsymbol{p}_{v_{obs}}\right) \tag{4.20}$$

where Property 3.6 (*cf.* page 48) is used. The function V is positive definite since $\boldsymbol{v}: \boldsymbol{v}^T M \boldsymbol{v}$ is a positive definite term and the potential energy $P_1(\delta \boldsymbol{p}_v)$ and $P_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)$ are positive definite functions of $\delta \boldsymbol{p}_v$ and $\delta \boldsymbol{p}_{v_{obs}}$, respectively.

The total derivative of V with respect to time is

$$\dot{V} = \boldsymbol{v}^T M_v \dot{\boldsymbol{v}} + \frac{1}{2}\boldsymbol{v}^T \dot{M}_v \boldsymbol{v} + k_1 \max\left(0, f_1(\delta \boldsymbol{p}_v)\right) \left(\frac{\partial f_1(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \dot{\boldsymbol{p}}_v + k_2 \max\left(0, f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)\right) \left(\frac{\partial f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial \boldsymbol{p}_{v_{obs}}}\right)^T \dot{\boldsymbol{p}}_{v_{obs}} \quad (4.21)$$

where the last two terms of (4.21) are obtained from the derivatives of (4.16) and (4.17) with respect to time. Then, solving for $M_v \dot{\boldsymbol{v}}$ from the closed-loop equation (4.19) yields

$$\begin{aligned} \dot{V} = {} & \frac{1}{2}\boldsymbol{v}^T \dot{M}_v \boldsymbol{v} + k_1 \max\left(0, f_1(\delta \boldsymbol{p}_v)\right) \left(\frac{\partial f_1(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \dot{\boldsymbol{p}}_v \\ & + k_2 \max\left(0, f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)\right) \left(\frac{\partial f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial \boldsymbol{p}_{v_{obs}}}\right)^T \dot{\boldsymbol{p}}_{v_{obs}} \\ & - \boldsymbol{v}^T C_v(\boldsymbol{v})\boldsymbol{v} - \boldsymbol{v}^T D(\boldsymbol{v})\boldsymbol{v} \\ & - \boldsymbol{v}^T J_{v_a}^T(\boldsymbol{\eta}_2) \left[\left(\frac{\partial P_1(\delta \boldsymbol{p}_v)}{\partial(\delta \boldsymbol{p}_v)}\right)^T + \left(\frac{\partial P_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial\left(\delta \boldsymbol{p}_{v_{obs}}\right)}\right)^T \right] k_{\eta_2} \tilde{\boldsymbol{e}}_{\eta_2} - \boldsymbol{v}^T K_v \boldsymbol{v} \end{aligned} \quad (4.22)$$

Applying Property 3.5 (*cf.* page 44) and Property 3.7 (*cf.* page 49) and cancelling the common terms, (4.22) can be rewritten as

$$\dot{V} = -\boldsymbol{v}^T D(\boldsymbol{v})\boldsymbol{v} - \boldsymbol{v}^T K_v \boldsymbol{v} \leq 0 \quad (4.23)$$

Hence, the function V is a Lyapunov function since $\dot{V} \leq 0$ for all $\delta \boldsymbol{p}_v$, $\delta \boldsymbol{p}_{v_{obs}}$, $\boldsymbol{v}$ and consequently the origin is stable and according to Theorem A.3, all the solutions $\delta \boldsymbol{p}_v(t)$, $\delta \boldsymbol{p}_{v_{obs}}(t)$ and $\boldsymbol{v}(t)$ are bounded.

The stability of the region priority controller for the AUV system with exact gravity and buoyancy force compensation is specified by the following theorem:

Theorem 4.1: For an autonomous underwater vehicle (4.1), the proposed control law described in (4.18) guarantees the convergence of $\boldsymbol{p}_v$ to the priority sub-regions $f_1(\delta \boldsymbol{p}_v) \leq 0$ and $f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right) \geq 0$ whilst $\boldsymbol{v}$ and $\tilde{\boldsymbol{e}}_{\eta_2}$ are driven to zero as $t \to \infty$.

Proof: Since the closed-loop equation (4.19) is independent of time (explicitly), La-Salle's theorem (*cf.* Theorem A.7) can be utilised to analyse the asymptotic stability of the origin. Now, let Ω be the set in which $\dot{V} = 0$ (where $\dot{V} = dV/dt$). The set Ω is defined as follows

$$\Omega = \left\{ \frac{1}{k_1}\left(\frac{\partial f_1(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T, \frac{1}{k_2}\left(\frac{\partial f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial \boldsymbol{p}_{v_{obs}}}\right)^T, \tilde{\boldsymbol{e}}_{\eta_2} \in \mathbb{R}^3, \boldsymbol{v} \in \mathbb{R}^6 : \dot{V} = 0 \right\} \tag{4.24}$$

It can be observed that $\dot{V} = 0$ if and only if $\boldsymbol{v} = 0$. In order to ensure that the solution belongs to Ω for all $t \geq 0$, it is necessary and sufficient that $\boldsymbol{v}(t) = 0$ for all $t \geq 0$. Hence, it also holds that $\dot{\boldsymbol{v}}(t) = 0$ for all $t \geq 0$. Then, it can be concluded from the closed-loop equation (4.19) that if the solution belongs to Ω for all $t \geq 0$ and the matrix $J_{v_a}^T$ is non-singular (full rank), yields

$$0 = J_{v_a}^T(\boldsymbol{\eta}_2)\begin{bmatrix} \left(\frac{\partial P_1(\delta \boldsymbol{p}_v)}{\partial(\delta \boldsymbol{p}_v)}\right)^T + \left(\frac{\partial P_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial\left(\delta \boldsymbol{p}_{v_{obs}}\right)}\right)^T \\ k_{\eta_2}\tilde{\boldsymbol{e}}_{\eta_2} \end{bmatrix} \tag{4.25}$$

Then, according to La Salle's theorem (*cf.* Theorem A.7), this is enough to guarantee global asymptotically stability of the origin. Consequently, $\boldsymbol{v} \to 0$ and $\frac{1}{k_1}\left(\frac{\partial f_1(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T, \frac{1}{k_2}\left(\frac{\partial f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right)}{\partial \boldsymbol{p}_{v_{obs}}}\right)^T, \tilde{\boldsymbol{e}}_{\eta_2} \to 0$ as $t \to \infty$ which finally leads to the convergence of $f_1(\delta \boldsymbol{p}_v) \leq 0$, $f_2\left(\delta \boldsymbol{p}_{v_{obs}}\right) \geq 0$ and $\tilde{\boldsymbol{e}}_{\eta_2} = 0$ as $t \to \infty$. The proof of Theorem 4.1 is completed, and the control objective is achieved.

4.3.3. Simulation Results

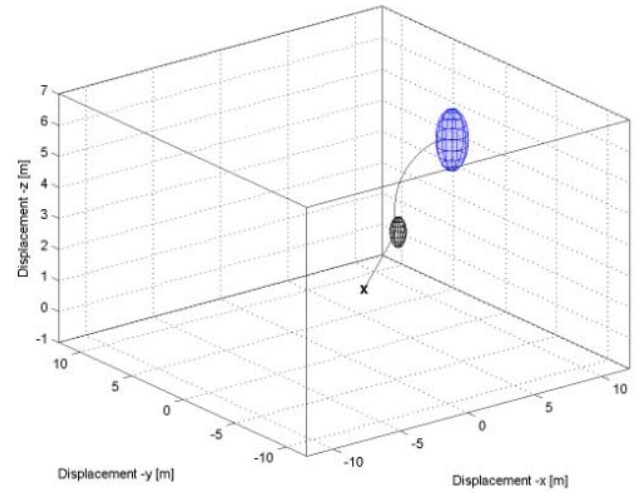
A numerical simulation with the purpose of illustrating the performance of the region priority controller with exact gravity and buoyancy force compensation for an autonomous underwater vehicle is now presented (*cf.* Appendix C for the parameters of the ODIN vehicle). In this simulation, the aim is to navigate the underwater vehicle into a primary sub-region (lower priority) while avoiding a secondary sub-region (higher priority) along its trajectory. The vehicle is initialised to position $\boldsymbol{p}_v(0) = [1.5 \quad 0 \quad 1]^T$m and orientation $\boldsymbol{\eta}_2(0) = [0 \quad 0 \quad 0]^T$. The target and obstacle's regions are specified by spherical shapes as follows

$$f_1(\delta \boldsymbol{p}_v) = (x_v - x_{vd})^2 + (y_v - y_{vd})^2 + (z_v - z_{vd})^2 - r_v^2 \leq 0 \quad (4.26)$$

$$f_2(\delta \boldsymbol{p}_{obs}) = (x_v - x_{obs})^2 + (y_v - y_{obs})^2 + (z_v - z_{obs})^2 - r_{obs}^2 \geq 0 \quad (4.27)$$

where the radii r_v and r_{obs} are specified as 1.0 m and 0.5 m, respectively. The reference points for the target and obstacle are set to: $[x_{vd} \quad y_{vd} \quad z_{vd}]^T = [8 \quad 0 \quad 5]^T$ m and $[x_{obs} \quad y_{obs} \quad z_{obs}]^T = [4 \quad 0 \quad 2.5]^T$ m. The gains are set to the following

$$k_1 = 18.8; k_{\eta_2} = 3; k_2 = 347.8; K_v = diag\{90, 90, 90, 40, 40, 40\};$$



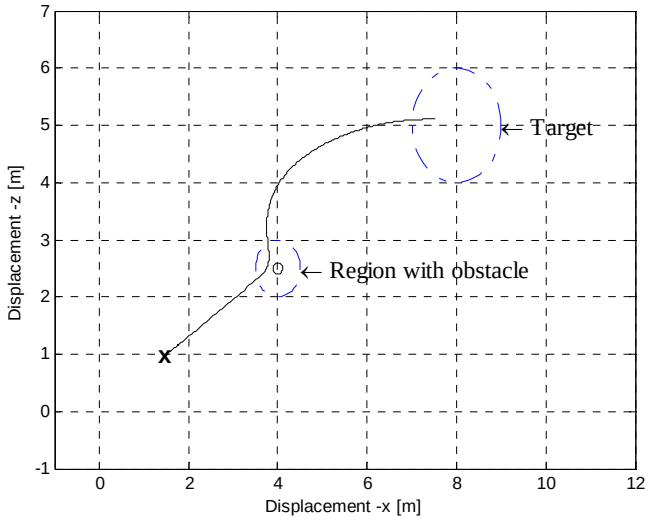


Figure 4.3: Position trajectory of the underwater vehicle

As shown in Figure 4.3, the vehicle moves into the primary sub-region while avoiding the repulsive sub-region on its trajectory. The initial position is marked with "**x**". It should be noted that the primary sub-region is assigned with a lower gain than the secondary sub-region so that the control objective achieves the first priority objective before executing tasks with less priority. The control formulation is straightforward when dealing with a single secondary sub-region. However, a more advanced control technique is needed if multiple secondary sub-regions exist as discussed in the next section.

4.4. Adaptive-Fuzzy Sub-Region Priority Reaching Controller for a 6-DOF AUV

By using the region-decomposition method explained in the previous section, numerous region objectives can be defined as additional motion constraints for the underwater robot, for instance a repulsive region, depth constraint, visibility constraint, region complexity and singularity avoidance. In other words, there is an infinite number of region criteria for a given generalised set-point control problem. This concept is also applicable for an underwater vehicle as illustrated in Figure 4.4, where multiple region criteria can be defined in order to achieve faster motion and energy consumption reduction.

To manage the multiple region criteria effectively, the perspective of an experienced human operator can play an important role in the successful fulfilment of a given reaching task. In an effort to include the skilled human operator into the region resolution problem, fuzzy logic can be used. Fuzzy logic incorporates human-like reasoning using if-then type fuzzy rules, reflecting a human pilot's expert knowledge. The proposed scheme is considered at the higher level with respect to the fuzzy logic based artificial pilot. In the past, many researchers have studied fuzzy approaches for marine system applications [4.13 - 4.22]. The autopilot fuzzy system, which had automatic adaptation and tuning ability, was initially proposed for ship control by Sutton and Jess [4.13]. Polkinghorne et al. [4.14] designed an industrial autopilot with a fuzzy controller that was adaptive to environment changes. In [4.15], Omerdic et al. presented a fuzzy autopilot for tracking control of a ship steering system. The design of a sliding mode fuzzy controller for an autonomous underwater vehicle (AUV) was proposed in [4.16] and Guo et al. presented the experimental results of an AUV using a sliding mode fuzzy control law [4.17]. Recently, a sliding mode control strategy with an

adaptive fuzzy logic algorithm for the depth control of a remotely operated underwater vehicle was proposed by Bessa [4.18]. Fuzzy logic techniques were also used for obstacle avoidance in autonomous underwater vehicle control described in [4.19, 4.20, 4.21, 4.22].

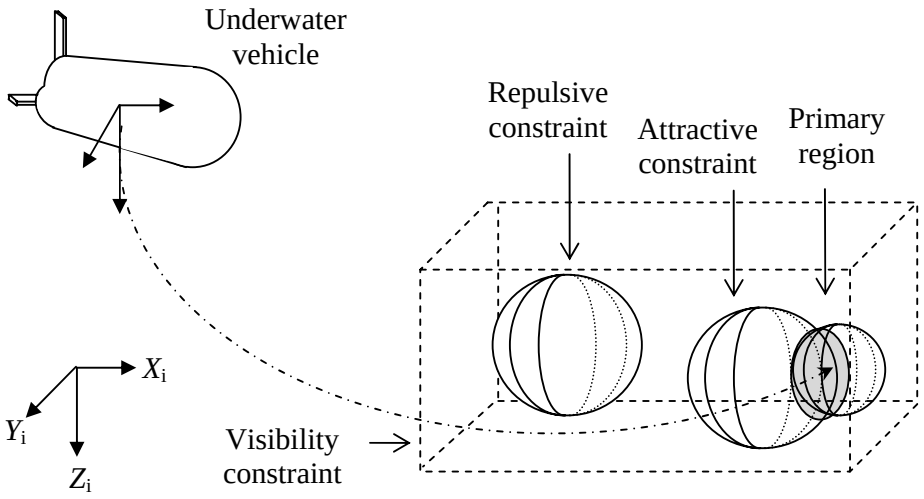


Figure 4.4: Sub-region control for an underwater vehicle with multiple criteria

In this subsection, the region priority reaching control law is merged with a fuzzy-logic approach that was proposed for an underwater vehicle subject to restoring force uncertainties [4.23]. While operating in an underwater environment, it is impossible to obtain an exact knowledge of the gravitational and buoyancy forces of an underwater vehicle so that a controller can compensate for this effect due to the variation of ocean water density. Therefore, the gravity regressor which is composed of known and unknown gravitational parameters can be utilised to overcome the uncertain restoring forces [4.1]. In addition to the proposed scheme, each sub-region criterion is defined within the framework of a sub-region priority technique. The hierarchy of the secondary sub-region tasks is established by a low-level artificial pilot that determines a weighting factor for each region criterion based on if-then type fuzzy rules that reflect an expert human pilot's knowledge. To interpret the fuzzy rules, a Mamdani type fuzzy inference is utilised [4.24].

4.4.1. Problem Formulation

Following the similar definition as in the previous section, the desired primary sub-region for the underwater vehicle as a single region is specified in the following equation

$$f(\delta \boldsymbol{p}_v) \leq 0 \tag{4.28}$$

where $\delta \boldsymbol{p}_v = \boldsymbol{p}_v - \boldsymbol{p}_{vd} \in \mathbb{R}^3$ are the continuous first partial derivatives of the primary region; $\boldsymbol{p}_{vd}$ is the stationary reference point inside the region such that $\dot{\boldsymbol{p}}_{vd}(t) = 0$. To incorporate the multiple secondary sub-regions, the following equation can be used

$$g_i\left(\delta \boldsymbol{p}_{vs_i}\right) \leq 0 \quad \text{with} \quad i = 1,2 \ldots N. \tag{4.29}$$

where N is the total number of secondary sub-regions and $\delta \boldsymbol{p}_{vs} = \boldsymbol{p}_{vs} - \boldsymbol{p}_{vsd_i} \in \mathbb{R}^3$ are the continuous first partial derivatives of the secondary sub-region; $\boldsymbol{p}_{vsd_i}$ is the stationary reference point inside the i^{th} secondary sub-region such that $\dot{\boldsymbol{p}}_{vsd_i}(t) = 0$. Since both objectives are accomplished concurrently, the position vector $\boldsymbol{p}_{vs}$ can be chosen as $\boldsymbol{p}_v$ and the body-fixed velocity vector for the secondary sub-region can be defined as follows

$$\dot{\boldsymbol{p}}_{vs} = \dot{\boldsymbol{p}}_v = E_1(\boldsymbol{e}_v)\boldsymbol{v}_1 \tag{4.30}$$

where the primary and secondary Jacobian matrices are identical. Note that, the rest of this thesis uses the notation of $\boldsymbol{e}_v$ instead of $\boldsymbol{\epsilon}$ to denote the vehicle orientation in unit quaternion form as the orientation of the onboard manipulator also shares a similar notation. During the region reaching task, the secondary sub-region is initially fulfilled before the fulfilment of both primary and secondary sub-regions simultaneously.

The corresponding potential energy function for the desired sub-region described in (4.28) can be specified as

$$P_p(\delta \boldsymbol{p}_v) = \frac{k_{p1}}{2}\left[\max\left(0, f(\delta \boldsymbol{p}_v)\right)\right]^2 \triangleq \begin{cases} 0, & f(\delta \boldsymbol{p}_v) \leq 0 \\ \frac{k_{p1}}{2} f^2(\delta \boldsymbol{p}_v), & f(\delta \boldsymbol{p}_v) > 0 \end{cases} \tag{4.31}$$

where $k_{p1} \in \mathbb{R}$ is a positive scalar. Similarly, the potential energy function for the multiple sub-regions in (4.29) can be defined as follows

$$\begin{aligned} P_s(\delta \boldsymbol{p}_{vs}) &= \sum_{i=1}^{N} \frac{\alpha_i}{2}\left[\max\left(0, g_i(\delta \boldsymbol{p}_{vs_i})\right)\right]^2 \\ &\triangleq \begin{cases} \sum_{i}^{N} 0, & f(\delta \boldsymbol{p}_{vs_i}) \leq 0 \\ \sum_{i}^{N} \frac{\alpha_i}{2} g^2(\delta \boldsymbol{p}_{vs_i}), & f(\delta \boldsymbol{p}_{vs_i}) > 0 \end{cases} \end{aligned} \tag{4.32}$$

Differentiating (4.31) with respect to $\delta \boldsymbol{p}_v$ and rewriting the equation in an advantageous form leads to

$$\left(\frac{\partial P_p(\delta \boldsymbol{p}_v)}{\partial(\delta \boldsymbol{p}_v)}\right)^T = k_{p1} \max\left(0, f(\delta \boldsymbol{p}_v)\right)\left(\frac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \tag{4.33}$$

Since multiple secondary sub-regions are specified, the normalised function can be utilised in order to prevent one single region criterion from excessively dominating the solution and this allows the particular secondary sub-region to be assigned with the pre-defined priority level. Therefore, the following function can be obtained using the normalised gradient of the scalar potential energy in (4.32)

$$\left(\frac{\partial P_s(\delta \boldsymbol{p}_{vs})}{\partial(\delta \boldsymbol{p}_{vs})}\right)^T = \sum_{i=1}^{N} \alpha_i \left[\max\left(0, g_i(\delta \boldsymbol{p}_{vs_i})\right)\right] \frac{\left(\frac{\partial g_i(\delta \boldsymbol{p}_{vs_i})}{\partial \boldsymbol{p}_{vs}}\right)^T}{\left\|\partial g_i(\delta \boldsymbol{p}_{vs_i})/\partial \boldsymbol{p}_{vs}\right\|_2} \tag{4.34}$$

where α_i is a weight that determines the extent of how much emphasis is placed on each secondary region criterion with respect to the other and $\|\cdot\|_2$ is the Euclidean norm. Note that when the norm of the function is equal to zero, the corresponding criterion is set to zero which leads the system to reach the primary region only.

Now, let (4.33) and (4.34) be represented as the primary region error $\tilde{\boldsymbol{e}}_p$ and secondary region error $\tilde{\boldsymbol{e}}_s$ respectively in the following form

$$\tilde{\boldsymbol{e}}_p = \max\big(0, f(\delta p_v)\big)\left(\frac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \tag{4.35}$$

$$\tilde{\boldsymbol{e}}_s = \sum_{i=1}^{N} \tilde{\boldsymbol{e}}_{si} \tag{4.36}$$

That is,

$$\tilde{\boldsymbol{e}}_{si} = \sum_{i=1}^{N} \beta_i \max\big(0, g_i(\delta \boldsymbol{p}_{vs})\big) \frac{\left(\frac{\partial g_i(\delta \boldsymbol{p}_{vs_i})}{\partial \boldsymbol{p}_{vs}}\right)^T}{\left\|\partial g_i(\delta \boldsymbol{p}_{vs_i})/\partial \boldsymbol{p}_{vs}\right\|_2} \tag{4.37}$$

where $\beta_i = \alpha_i / k_q$; k_q is a positive constant. Meanwhile, the error between the actual and desired orientation for the underwater vehicle can be formulated as in Chapter 3 and the quaternion tracking error can be considered as

$$\dot{\tilde{\boldsymbol{e}}}_v = \begin{bmatrix} -\frac{1}{2}\tilde{\boldsymbol{e}}_\varepsilon^T \\ \frac{1}{2}(\tilde{e}_0 I_3 + \tilde{\boldsymbol{e}}_\varepsilon{}^\times) \end{bmatrix} \tilde{\boldsymbol{\omega}} \Leftrightarrow \dot{\tilde{\boldsymbol{e}}}_v = \frac{1}{2} E_2(\tilde{\boldsymbol{e}}_v)\tilde{\boldsymbol{\omega}} \tag{4.38}$$

where $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} - \boldsymbol{\omega}_d$; $\boldsymbol{\omega}_v$ is the actual angular velocity and the desired angular velocity of the vehicle, $\boldsymbol{\omega}_d$, is always zero ($\boldsymbol{\omega}_d = 0$), hence $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega}$. The region reaching controller presented in [4.9] contains the Jacobian transpose in its update law and not its inverse which results in an inexact mapping and a coupling amongst the error directions [4.25]. Consequently, the Jacobian transpose based method is not suitable for design of the controller with restoring force compensation. As opposed to [4.9], the sub-region priority reaching controller with uncertain restoring force compensation for an underwater vehicle is proposed as follows

$$\boldsymbol{\tau} = -K_0\tilde{\boldsymbol{e}}_T - K_s\tilde{\boldsymbol{e}}_s - K_v\boldsymbol{v} + Z(\boldsymbol{e}_v)\hat{\boldsymbol{\varphi}} \tag{4.39}$$

with

$$\begin{aligned} K_0 &= diag\left\{E_1^T k_{p1}, \quad k_{p2} I_{3\times 3}\right\}; \\ K_s &= \left[E_1^T k_q \quad 0_{3\times 3}\right]^T; \\ K_v &= diag\{k_{v1} I_{3\times 3}, \quad k_{v2} I_{3\times 3}\}; \end{aligned} \tag{4.40}$$

where k_{p1} and k_q are previously defined in (4.33) and (4.37), respectively. k_{p2}, k_{v1} and k_{v2} are positive constants. $I_{3\times 3}$ and $0_{3\times 3}$ are the 3×3 identity and zero matrices, respectively. The proposed controller adopts the approach of the pseudo-inverse matrix in the parameters update law such that the drawback of the restoring compensation using the Jacobian transpose method is eliminated. Accordingly, the estimate vector $\hat{\boldsymbol{\varphi}}$ can be updated online by

$$\dot{\hat{\boldsymbol{\varphi}}} = -\Gamma^{-1}\mathrm{Z}^{\mathrm{T}}(\boldsymbol{e}_v)\left(\boldsymbol{v} + \sigma J_{v_e}^{\dagger}(\boldsymbol{e}_v)\tilde{\boldsymbol{e}}_{T_s}\right) \tag{4.41}$$

with

$$\tilde{\boldsymbol{e}}_{T_s} = \tilde{\boldsymbol{e}}_T + (k_q/k_{p1})\begin{bmatrix}\tilde{\boldsymbol{e}}_s \\ 0_{3\times 1}\end{bmatrix} \tag{4.42}$$

where Γ and σ denote the positive definite matrices and $J_{v_e}^{\dagger}(\boldsymbol{e}_v)$ is the pseudo-inverse of matrix $J_{v_e}(\boldsymbol{e}_v)$. The error $\tilde{\boldsymbol{e}}_T$ is defined as

$$\tilde{\boldsymbol{e}}_T = \left[\tilde{\boldsymbol{e}}_p^T \quad \tilde{\boldsymbol{e}}_\varepsilon^T\right]^T \tag{4.43}$$

while the quaternion vector $\tilde{\boldsymbol{e}}_\varepsilon$ is part of $\tilde{\boldsymbol{e}}_v$. Substituting (4.39) into (4.1) the closed-loop equation is obtained as

$$M_v\dot{\boldsymbol{v}} + C_v(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + K_0\tilde{\boldsymbol{e}}_T + K_s\tilde{\boldsymbol{e}}_s + K_v\boldsymbol{v} + Z(\boldsymbol{e}_v)\tilde{\boldsymbol{\varphi}} = 0 \tag{4.44}$$

where $\tilde{\boldsymbol{\varphi}} = \boldsymbol{\varphi} - \hat{\boldsymbol{\varphi}}$.

Now, consider the following conditions when choosing the feedback gains

$$\begin{gathered}\frac{1}{2}\left(k_{p1}+\sigma k_{v1}\right)/\left\|\frac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right\|^2-\sigma^2\lambda_m>0;\\ \left(k_{p2}+\sigma k_{v2}\right)-\sigma^2\lambda_m>0;\\ \frac{1}{2}\left(k_q+\sigma\left(\frac{k_q k_{v1}}{k_{p1}}\right)\right)\beta_{min}b_{min}-\sigma^2\lambda_m N>0;\end{gathered} \tag{4.45}$$

and

$$\begin{gathered}\lambda_{min}[D(\boldsymbol{v})]+\lambda_{min}[K_v]-\sigma c_0>0;\\ \lambda_{min}\left[J_{v_e}(\boldsymbol{e}_v)K_0\right]\frac{k_q}{k_{p1}}>0;\end{gathered} \tag{4.46}$$

with

$$\lambda_m \triangleq \lambda_{max}\left[J_{v_e}^{\dagger\,T}(\boldsymbol{e}_v)M_v J_{v_e}^{\dagger}(\boldsymbol{e}_v)\right],$$

$$\beta_{min} \triangleq \min_i\left(\frac{1}{\beta_i}\right),$$

$$b_{min} \triangleq \min_i\left(1/\left\|\left(\frac{\partial g_i(\delta \boldsymbol{p}_{vs_i})}{\partial \boldsymbol{p}_{vs}}\right)\right\|^2\right); i=1,2\ldots N$$

where $\lambda_{max}[A]$ and $\lambda_{min}[A]$ represent the maximum and minimum eigenvalues of matrix A and $c_0>0$ is a constant. Therefore, the next theorem can be stated as:

Theorem 4.2: The region priority control law in (4.39) guarantees the asymptotic convergence of $\boldsymbol{v}$, $\tilde{\boldsymbol{e}}_p$, $\tilde{\boldsymbol{e}}_s$, and $\tilde{\boldsymbol{e}}_\varepsilon$ for an underwater vehicle given by (4.1) provided that (4.40), (4.41) and (4.42) are fulfilled and the feedback gains are chosen to satisfy conditions (4.45) and (4.46).

Proof: See Appendix B.

4.4.2. Regional Constraints

There are various motion constraints of an underwater vehicle that can be defined as regions. In this section, several regional constraints, i.e. obstacle avoidance, shape and depth region constraints that are imposed for a 6-DOF underwater vehicle are briefly highlighted. The features of region constraints are assumed to be known.

- *Obstacle avoidance* - In order to take into account the obstacle avoidance goal, it is useful to define a specified sub-region containing an obstruction. When the obstacle is within the vehicle path or close to the primary target, then a repulsive sub-region must be specified to avoid a collision. A repulsive spherical shape sub-region can be specified as

$$g_1\left(\delta \boldsymbol{p}_{vs_1}\right) = r_{s_1}^2 - (x - x_{o1})^2 - (y - y_{o1})^2 - (z - z_{o1})^2 \leq 0 \tag{4.47}$$

  where r_{s_1} denotes the tolerance of the repulsive sub-region. The repulsive spherical region for an underwater vehicle was illustrated previously in Figure 4.4.

- *Shape Constraint* - Unlike a point, a region can be designed with various shapes such as a circle or a rectangle for a planar plane, or a sphere or cubic for 3D applications since they are associated with length, width and radius or diameter. To define a simple shape like a circle or a sphere is very straightforward as less boundary constraints are required. However, some regions (i.e. pentagon shape or hexagonal prism) are difficult to determine even though the robot can reach it. These regions are defined with abundant and complex boundary constraints. Therefore, it is undesirable not only for a robot to fall into these regions but also for it to even to come too close. In the case of an underwater application where a region is classified into two types, a simple region and a complex region, the complex region shape is applied when the vehicle is required to navigate through a pipeline with bends while the non-complex region is the region where the vehicle manoeuvres outside the pipeline as illustrated in Figure 4.5. Note that, it is difficult to determine the form of a complex region especially for a region inside a structure or a region that is very close to the seabed. In this work, it is proposed that the underwater vehicle is required to be kept inside a simple or attractive region shape instead of navigating through a complex region.

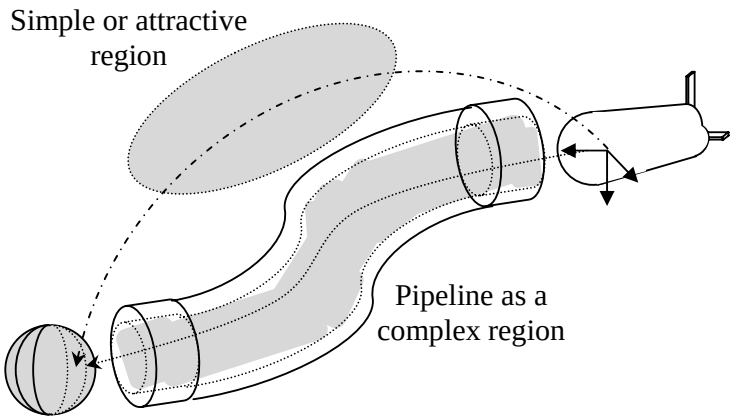


Figure 4.5: Illustration of simple and complex sub-region constraints

- *Depth Constraint* - The underwater vehicle exhibits different static forces depending on its depth. If the vehicle is positive buoyant, a large force is required for the vehicle to maintain a certain depth far below the water surface. Therefore, by defining the depth limit, less energy is required for the system. Moreover, it certainly assists the operator to monitor the vehicle navigation visually. Therefore, it is proposed that the depth boundary for vehicle navigation can be defined as follows

$$g_2(\delta \boldsymbol{p}_{vs_2}) = z - z_{o1} \leq r_{s_2} \tag{4.48}$$

$$g_2(\delta \boldsymbol{p}_{vs_2}) = z - z_{o1} \geq -r_{s_2} \tag{4.49}$$

where r_{s_2} is the depth tolerance value while (4.48) and (4.49) represent the maximum and minimum boundaries, respectively. Note that, these boundaries can be changed to the other way round depending on the initial and final position of the vehicle.

4.4.3. Fuzzy Inference System

In the preceding section, the regional motion constraints were introduced. However, a strategy must still be presented that determines the importance of each defined region objective with respect to the others. As mentioned previously, the weight factor $\alpha_i \in [0,1]$ can be used as a tool to define the relative importance of each objective within the set of objective functions. Greater values indicate a higher demand, whereas smaller values indicate a lower demand for that particular region objective. Note that, only the highest priority control gain is required to be activated at one particular time

due to the summation factor in the control law defined previously. Accordingly, it leads to less oscillatory movement of the underwater vehicle when approaching the primary sub-region.

Fuzzy logic is known as a decision-making process by mimicking the human pilot's actions. A Mamdani fuzzy inference method which involves the phases of fuzzifying the inputs, applying a fuzzy operator, applying an implication method, aggregating the output and defuzzifying the output fuzzy set can be used for a decision-making procedure. In this work, the input linguistic variables and the corresponding fuzzy sets are: $obstacle = \{exist, not\ exist\}$, $shape = \{simple, complex\}$, $depth = \{maintain, not\ maintain\}$. Regarding the output linguistic variables and the corresponding fuzzy sets, they are set to be $\alpha_k = \{low, high\};\ k = 1 \dots 3$. The following set of rules have been implemented:

- If the obstacle exists then α_1 is high;
- If the obstacle does not exist then α_1 is low;
- If the obstacle exists then α_2 is low;
- If the shape is simple then α_2 is low;
- If the obstacle does not exist and the shape is complex then α_2 is high;
- If the obstacle exists or the shape is complex then α_3 is low;
- If the depth is maintained then α_3 is low;
- If the obstacle does not exist and the shape is simple and the depth is not maintained then α_3 is high;

A complete and consistent set of fuzzy rules with three linguistic variables defined with two fuzzy sets for each linguistic variable requires a total of 24 rules. To reduce the number of rules, a hierarchical structure that gives higher priority to the obstacle avoidance and the lowest priority to the depth keeping in the form of $obstacle \rightarrow shape \rightarrow depth$. This leads to only 8 rules in total instead of 24, and hence provides a significant reduction in the number of rules. Given that only one sub-region is activated when there are multiple sub-regions overlapping, a prevention of any possible conflict between the regions is solved in favour of the higher priority sub-region. In addition, a higher gain can be avoided since only a single α_i is allowed to be high during the

overlapping phase, reducing the oscillatory movement of the vehicle when reaching its target.

The $and - or$ logic operations have been calculated based on the $min - max$ operations, respectively. In addition, the implication-aggregation operations have been carried out by implementing the $min - max$ operations, respectively. The values of $\alpha_k \in [0,1]$ are obtained by the defuzzification phase using the centroid technique with normalisation [4.24].

4.4.4. Simulation Results

Simulations are carried out to assess the effectiveness of the proposed fuzzy-logic based sub-region priority control law for an underwater vehicle (*cf.* Appendix C for parameters of the ODIN vehicle). The main objective is to ensure the underwater vehicle reaches the primary sub-region while fulfilling the secondary objectives (obstacle avoidance, shape and depth constraint) defined by the secondary sub-regions as much as possible. The vehicle is initialised to position $\boldsymbol{p}_v(0) = [0 \quad 0 \quad 1]^T$m and orientation $\boldsymbol{e}_v(0) = [0 \quad 0 \quad 0 \quad 1]^T$. In the simulation, the primary sub-region is specified in the following inequality function

$$f(\delta \boldsymbol{p}_v) = (x_v - x_{vd})^2 + (y_v - y_{vd})^2 + (z_v - z_{vd})^2 - r_v^2 \leq 0 \tag{4.50}$$

where $[x_{vd} \quad y_{vd} \quad z_{vd}]^T = [8 \quad 0 \quad 5]^T$m and the orientation in the unit quaternion is kept constant. Similarly, the objective functions for the secondary sub-regions with constraints can be specified as follows

$$g_1(\delta \boldsymbol{p}_{vs_1}) = \left(x_v - x_{vsd_1}\right)^2 + \left(y_v - y_{vsd_1}\right)^2 + \left(z_v - z_{vsd_1}\right)^2 - r_{vs_1}^2 \geq 0 \tag{4.51}$$

$$g_2(\delta \boldsymbol{p}_{vs_2}) = \left(x_v - x_{vsd_2}\right)^2 + \left(y_v - y_{vsd_2}\right)^2 + \left(z_v - z_{vsd_2}\right)^2 - r_{vs_2}^2 \geq 0 \tag{4.52}$$

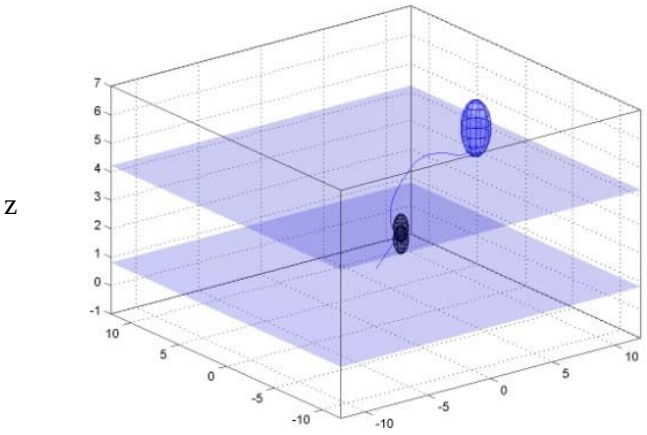
$$g_3(\delta \boldsymbol{p}_{vs_3}) = \left(z_v - z_{vsd_3}\right)^2 - r_{vs_3}^2 \geq 0 \tag{4.53}$$

where (4.51), (4.52) and (4.53) are assigned for obstacle avoidance and shape and depth constraints, respectively.

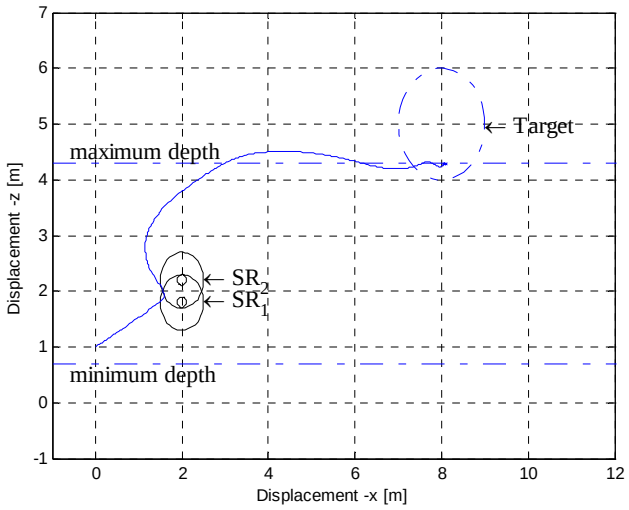
The reference points for each secondary sub-region are set to:

$$[x_{vsd_1} \quad y_{vsd_1} \quad z_{vsd_1}]^T = [2 \quad 0 \quad 1.8]^T \text{m}; [x_{vsd_2} \quad y_{vsd_2} \quad z_{vsd_2}]^T = [2 \quad 0 \quad 2.2]^T \text{m};$$
$$z_{vsd_3} = 2.5 \text{ m};$$

while the tolerances are chosen as $r_{vs_1} = 0.5$ m; $r_{vs_2} = 0.5$ m; $r_{vs_3} = 1.8$ m.



(a)



(b)

Figure 4.6: (a) 3D illustration of underwater vehicle position, (b) A planar plane view; SR_1 and SR_2 denote the obstacle and complex sub-regions

For the best performance of the region priority reaching controller with restoring forces compensation, the gains are set to the following

$$k_{p1} = 18.8;\ k_{p2} = 36;\ k_q = 100;\ K_v = diag\{90, 90, 90, 40, 40, 40\};$$
$$\sigma = diag\{0.08I_3, 0.9I_3\};$$
$$\alpha_i = [100\ 100\ 0.5]^T;\ \Gamma = 2I_4;$$

where $i = 1, 2, 3$ and I_4 is a 4×4 identity matrix. Figure 4.6 shows the convergence of an underwater vehicle into the primary sub-region while fulfilling the secondary sub-region constraints. The initial position and the intersection of the secondary sub-regions are marked with "**x**" and "I", respectively. At a secondary region intersection where it contains the obstacle, shape and depth constraints, the fuzzy approach is utilised in order to select that only one sub-region is activated. Consequently, the control effort leads the vehicle to move out from this undesired sub-region as can be clearly seen in Figure 4.6(b).

As shown in Figure 4.7, the controller error expressed in terms of the root mean square error converges to the specified tolerance value in parallel with the stability properties of the designed controller. The vehicle attitude is presented in Figure 4.8. As the figure reveals, the vehicle attitude was kept constant with a slight change at the initial start time due to the control needing to adapt for the uncertain restoring forces.

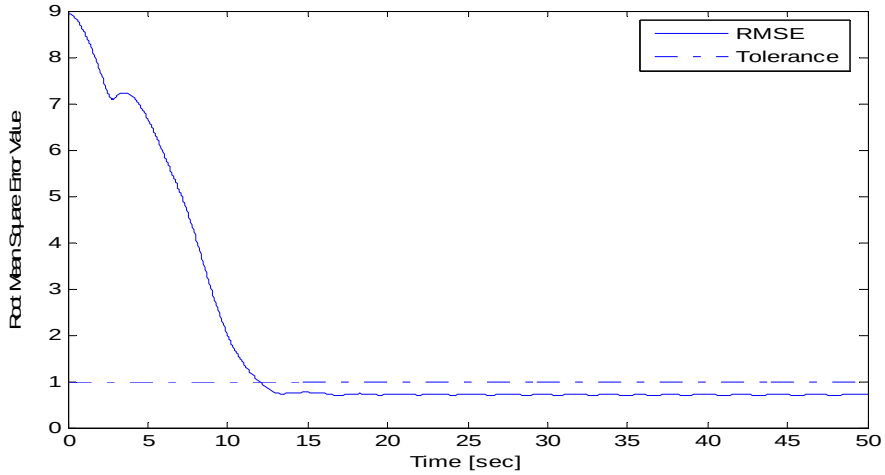


Figure 4.7: Convergence of position error inside the error band

An intersection of the secondary sub-regions occurs at 2.65 s to 2.95 s and its normalised index is presented in Figure 4.9. To avoid excessive control effort and oscillatory vehicle movement which might happen due to the high gains resulting from the summation feature in the control law, a fuzzy approach is implemented during this period. Therefore, only the highest priority secondary sub-region is allowed to be activated at any one time. In other words, a possible conflict amongst the sub-regions with constraints is prevented by solving in favour of the fuzzy inference system.

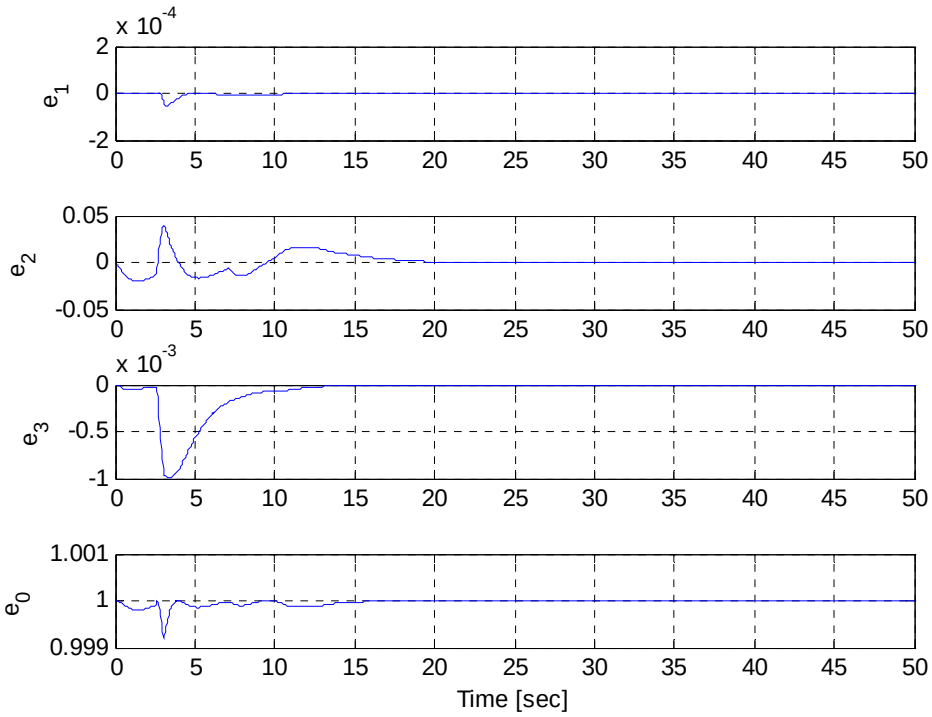


Figure 4.8: The vehicle attitude

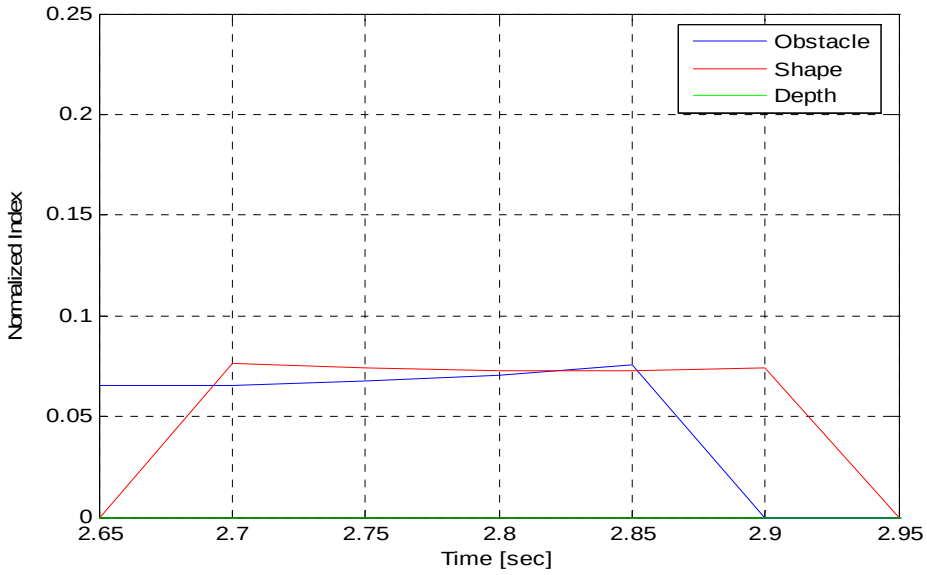
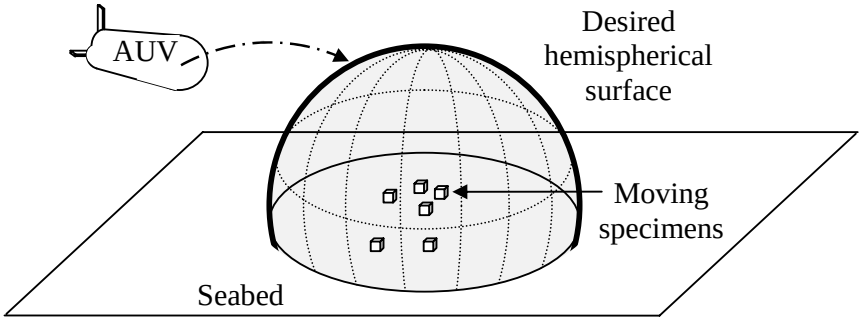


Figure 4.9: Normalised index at the region intersection

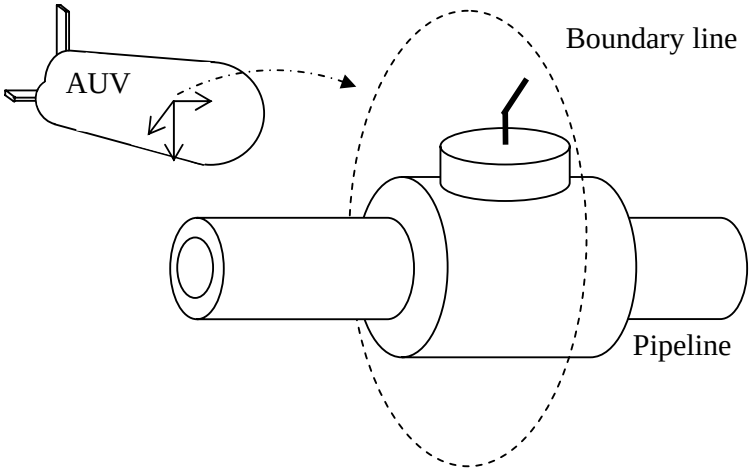
4.5. Adaptive Region Boundary-Based Controller for a 6-DOF AUV

As described earlier in this chapter, most underwater tasks require the vehicle to perform a point to point motion where the desired target is specified as a point. Nevertheless, the control objective favours a region rather than a point, for example, maintaining the underwater vehicle within a minimum and maximum depth in water; the underwater vehicle travelling inside a pipeline for specific task; avoiding an obstacle located at a specific region; and an underwater region constraint to ensure visibility during motion.

Recently, a region reaching control law was proposed for an autonomous underwater vehicle [4.25] and an underwater vehicle with a mounted arm [4.9]. Using this control concept, the robot is allowed to move into the desired region where it can be specified in various shapes and sizes depending on the underwater mission. On the other hand, underwater vehicle tasks such as heading towards a hemispherical surface to observe the specimen movement as in Figure 4.10(a) and monitoring the exterior of a pipeline as in Figure 4.10(b), utilise the boundary as a desired objective.



(a)



(b)

Figure 4.10: (a) AUV heading towards the hemispherical surface to observe the specimen movement, (b) Using an underwater vehicle to monitor the exterior structure of a pipeline

In this section, a new adaptive region boundary-based control law is proposed for an autonomous underwater vehicle [4.26]. Instead of specifying the desired target as a region or a point, the control objective is defined as a boundary of a region. A region boundary-based control concept is regarded as a generalised region or set-point control problem where the system is regulated to move near to the boundary, rather than into a region or a point. Therefore, the position of a vehicle can be initialised either from inside or outside of a region prior to its convergence into the boundary area within a specific time. Moreover, the proposed control law utilises the inverse Jacobian matrix in the adaptive law for an exact mapping for the compensation of persistent effects such as the vehicle restoring forces.

4.5.1. Problem Formulation

In region boundary-based control, the desired target is specified by at least two sub-regions with different sizes intersecting at the same point. The inner sub-region acts as a repulsive region while the outer sub-region acts as an attractive region. Using this control concept, the vehicle can be initially positioned within the region, or outside the region prior to its convergence into the boundary area. When the inner sub-region size is reduced to zero, the desired boundary is transformed into a desired region. Alternatively, when both inner and outer sub-regions are specified to be arbitrarily small, the desired boundary is transformed into a desired point with a specified accuracy. Therefore, the proposed region boundary-based control concept is also a generalisation of the region or set-point control problem. Figure 4.11 illustrates the region boundary-based control objective.

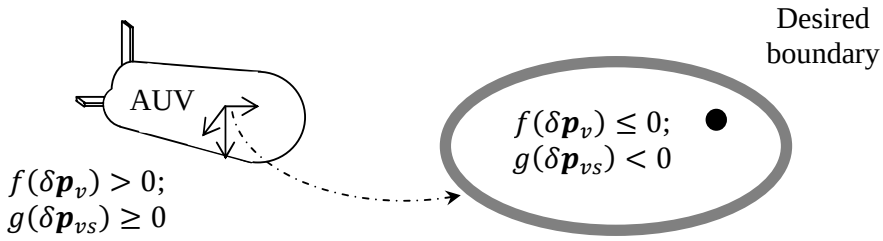


Figure 4.11: Region boundary-based control for an underwater vehicle

Define the desired outer sub-region for the underwater vehicle as follows

$$f(\delta \boldsymbol{p}_v) \leq 0 \tag{4.54}$$

where $\delta \boldsymbol{p}_v = \boldsymbol{p}_v - \boldsymbol{p}_{vd} \in \mathbb{R}^3$ are the continuous first partial derivatives of the outer sub-region; $\boldsymbol{p}_{vd}$ is the stationary reference point inside the region such that $\dot{\boldsymbol{p}}_{vd}(t) = 0$. The following inequality function can be used for the inner sub-region

$$g(\delta \boldsymbol{p}_{vs}) \geq 0 \tag{4.55}$$

where $\delta \boldsymbol{p}_{vs} = \boldsymbol{p}_{vs} - \boldsymbol{p}_{vd} \in \mathbb{R}^3$ are the continuous first partial derivatives of the secondary sub-region.

Note that, (4.54) and (4.55) are defined arbitrarily close to each other, such that

$$f(\delta \boldsymbol{p}_v) \approx g(\delta \boldsymbol{p}_{vs}) \tag{4.56}$$

The corresponding potential energy function for the desired sub-region described in (4.54) can be specified as

$$P_p(\delta \boldsymbol{p}_v) = \frac{k_{p1}}{2}\left[\max\left(0, f(\delta \boldsymbol{p}_v)\right)\right]^2 \triangleq \begin{cases} 0, & f(\delta \boldsymbol{p}_v) \leq 0 \\ \frac{k_{p1}}{2} f^2(\delta \boldsymbol{p}_v), & f(\delta \boldsymbol{p}_v) > 0 \end{cases} \tag{4.57}$$

where $k_{p1} \in \mathbb{R}$ is a positive scalar.

Similarly, the potential energy function for the inner sub-region in (4.55) can be defined as follows

$$\begin{aligned} P_s(\delta \boldsymbol{p}_{vs}) &= \frac{k_{q1}}{2}\left[\max\left(0, g(\delta \boldsymbol{p}_{vs})\right)\right]^2 \\ &\triangleq \begin{cases} 0, & g(\delta \boldsymbol{p}_{vs}) \geq 0 \\ \frac{k_{q1}}{2} g^2(\delta \boldsymbol{p}_{vs}), & g(\delta \boldsymbol{p}_{vs}) < 0 \end{cases} \end{aligned} \tag{4.58}$$

where $k_{q1} \in \mathbb{R}$ is a positive scalar.

Differentiating (4.57) and (4.58) with respect to $\delta \boldsymbol{p}_v$ and $\delta \boldsymbol{p}_{vs}$ gives

$$\left(\frac{\partial P_p(\delta \boldsymbol{p}_v)}{\partial(\delta \boldsymbol{p}_v)}\right)^T = k_{p1} \max\left(0, f(\delta \boldsymbol{p}_v)\right)\left(\frac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \tag{4.59}$$

$$\left(\frac{\partial P_s(\delta \boldsymbol{p}_{vs})}{\partial(\delta \boldsymbol{p}_{vs})}\right)^T = k_{q1} \max\left(0, g(\delta \boldsymbol{p}_{vs})\right)\left(\frac{\partial g(\delta \boldsymbol{p}_{vs})}{\partial \boldsymbol{p}_{vs}}\right)^T \tag{4.60}$$

Now, let (4.59) and (4.60) be represented as the primary region error $\tilde{\boldsymbol{e}}_p$ and secondary region error $\tilde{\boldsymbol{e}}_s$, respectively, in the following form

$$\tilde{\boldsymbol{e}}_p = \max\big(0, f(\delta \boldsymbol{p}_v)\big)\left(\frac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \tag{4.61}$$

$$\tilde{\boldsymbol{e}}_s = \max\big(0, g(\delta \boldsymbol{p}_{vs})\big)\left(\frac{\partial g(\delta \boldsymbol{p}_{vs})}{\partial \boldsymbol{p}_{vs}}\right)^T \tag{4.62}$$

It is interesting to note that since the control objectives are accomplished concurrently, the position vector $\boldsymbol{p}_{vs}$ can be chosen as $\boldsymbol{p}_v$ and the body-fixed velocity vector for the inner sub-region can be defined as follows

$$\dot{\boldsymbol{p}}_{vs} = \dot{\boldsymbol{p}}_v = E_1(\boldsymbol{e}_v)\boldsymbol{v} \tag{4.63}$$

where the inner and outer rotational matrix are identical. At this point, the quaternion propagation equation can be considered as in Chapter 3 (*cf.* Equation (3.22))

$$\dot{\tilde{\boldsymbol{e}}}_v = \begin{bmatrix} -\frac{1}{2}\tilde{\boldsymbol{e}}_\varepsilon^T \\ \frac{1}{2}(\tilde{e}_0 I_3 + \tilde{\boldsymbol{e}}_\varepsilon{}^\times) \end{bmatrix}\tilde{\boldsymbol{\omega}} \Leftrightarrow \dot{\tilde{\boldsymbol{e}}}_v = \frac{1}{2}E_2(\tilde{\boldsymbol{e}}_v)\tilde{\boldsymbol{\omega}} \tag{4.64}$$

where $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} - \boldsymbol{\omega}_d$; $\boldsymbol{\omega}_{vd}$ is always zero which results in $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega}$.
Then, the region boundary-based controller for an underwater vehicle can be proposed as

$$\boldsymbol{\tau} = -K_0\tilde{\boldsymbol{e}}_T - K_s\tilde{\boldsymbol{e}}_s - K_v\boldsymbol{v} + Z(\boldsymbol{\eta}_e)\hat{\boldsymbol{\varphi}} \tag{4.65}$$

with

$$\begin{aligned} K_0 &= diag\left\{E_1^T k_{p1}, \quad k_{p2} I_{3\times 3}\right\}; \\ K_s &= \left[E_1^T k_{q1} \quad 0_{3\times 3}\right]^T; \\ K_v &= diag\{k_{v1} I_{3\times 3}, \quad k_{v2} I_{3\times 3}\}; \end{aligned} \tag{4.66}$$

where k_{p1} and k_{q1} are previously defined in (4.57) and (4.58), respectively. k_{p2}, k_{v1} and k_{v2} are positive constants.

The proposed controller adopts the approach of the pseudo-inverse matrix in the parameters update law such that the drawback of the persistent effects compensation using the Jacobian transpose method is avoided. Accordingly, the estimate vector $\widehat{\boldsymbol{\varphi}}$ can be updated online by

$$\dot{\widehat{\boldsymbol{\varphi}}} = -\Gamma^{-1}\mathrm{Z}^{\mathrm{T}}(\boldsymbol{\eta}_e)\left(\boldsymbol{v} + \sigma J_{v_e}^{\dagger}(\boldsymbol{e}_v)\tilde{\boldsymbol{e}}_{T_s}\right) \tag{4.67}$$

with

$$\tilde{\boldsymbol{e}}_{T_s} = \tilde{\boldsymbol{e}}_T + (k_{q1}/k_{p1})\begin{bmatrix}\tilde{\boldsymbol{e}}_s \\ 0_{3\times 1}\end{bmatrix} \tag{4.68}$$

where Γ denotes the positive definite matrix and σ is a positive constant. $J_{v_e}^{\dagger}(\boldsymbol{e}_v)$ is the pseudo-inverse of matrix $J_{v_e}(\boldsymbol{e}_v)$. The vehicle error is denoted by $\tilde{\boldsymbol{e}}_T = \begin{bmatrix}\tilde{\boldsymbol{e}}_p^T & \tilde{\boldsymbol{e}}_\varepsilon^T\end{bmatrix}^T$; the quaternion vector $\tilde{\boldsymbol{e}}_\varepsilon$ is part of $\tilde{\boldsymbol{e}}_v$. Substituting (4.65) into (4.1) the closed-loop equation is obtained as

$$M_v\dot{\boldsymbol{v}} + C_v(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + K_0\tilde{\boldsymbol{e}}_T + K_s\tilde{\boldsymbol{e}}_s + K_v\boldsymbol{v} + Z(\boldsymbol{\eta}_e)\tilde{\boldsymbol{\varphi}} = 0 \tag{4.69}$$

where $\tilde{\boldsymbol{\varphi}} = \boldsymbol{\varphi} - \widehat{\boldsymbol{\varphi}}$. It is interesting to note that the sign of the last term in (4.65) depends to the parameter estimation error $\tilde{\boldsymbol{\varphi}}$ (*cf.* Proof of Theorem in Appendix B). Now, consider the following conditions when choosing the feedback gains

$$\begin{gathered}\frac{1}{2}\left(k_{p1} + \sigma k_{v1}\right)\Big/\left\|\frac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right\|^2 - \sigma^2\lambda_m > 0; \\ \left(k_{p2} + \sigma k_{v2}\right) - \sigma^2\lambda_m > 0; \\ \left(k_{q1} + \sigma\left(\frac{k_{q1}k_{v1}}{k_{p1}}\right)\right)\Big/\left\|\frac{\partial g(\delta \boldsymbol{p}_{vs})}{\partial \boldsymbol{p}_{vs}}\right\|^2 - 2\sigma^2\lambda_m > 0;\end{gathered} \tag{4.70}$$

and

$$\begin{gathered}\lambda_{min}[D(\boldsymbol{v})] + \lambda_{min}[K_v] - \sigma c_0 > 0; \\ \lambda_{min}\left[J_{v_e}^{\dagger\ T}(\boldsymbol{e}_v)K_0\right] > 0; \\ \lambda_{min}\left[J_{v_e}^{\dagger\ T}(\boldsymbol{e}_v)K_s\right] > 0;\end{gathered} \tag{4.71}$$

with $\lambda_m \triangleq \lambda_{max}\left[J_{v_e}^{\dagger\ T}(\boldsymbol{e}_v)M_v J_{v_e}^{\dagger}(\boldsymbol{e}_v)\right]$.

Therefore, the next theorem can be stated as:

Theorem 4.3: The proposed control law described in (4.65) and (4.66) and the adaptation law defined in (4.67) and (4.68) guarantee that $\boldsymbol{v}$, $\tilde{\boldsymbol{e}}_p$, $\tilde{\boldsymbol{e}}_s$, and $\tilde{\boldsymbol{e}}_\varepsilon$ are driven to zero as $t \to \infty$ for an underwater vehicle given by (4.1) provided that the feedback gains are chosen to satisfy conditions (4.70) and (4.71).

Proof: See Appendix B.

4.5.2. Edge-Based Segmentation Approach

As reported in [4.26, 4.27], the vehicle was only forced to travel to the nearest target on the boundary line with respect to its initial position. Nevertheless, the vehicle should reach the target on the boundary line regardless of the relative distance between the initial and final position. Therefore, this allows the system to be initialised at any position and converges to any specified final position on the boundary line. For instance, the vehicle is heading towards the minimum depth limit as illustrated in Figure 4.12, instead of navigating to the maximum depth limit (the nearest point on the boundary lines with respect to its initial position).

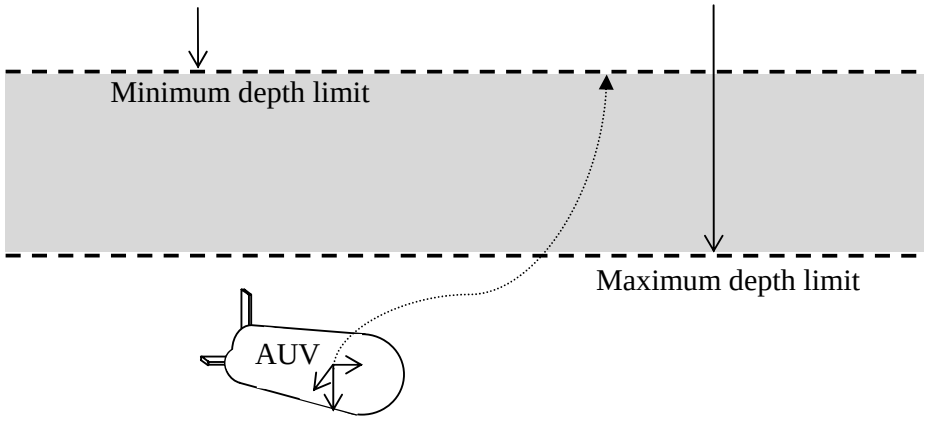


Figure 4.12: An underwater vehicle moves towards a specific depth boundary

In this section, an edge-based segmentation method is employed to achieve this control objective which enables the system to approach the specific range of positions within the boundary line. It is interesting to note that the boundary is primarily divided into multiple segments so that a desired position on the boundary line can be easily specified

depending on the underwater mission. If the segmented range is defined to be arbitrarily small, then the system is allowed to reach an exact point on the line.

As illustrated in Figure 4.13, the target can be specified as a segmented range or a point along the boundary line. Firstly, define the additional region constraints

$$f_{seg_i}\left(\delta \boldsymbol{p}_{v_{seg_i}}\right) \leq 0 \quad \text{with} \quad i = 1,2,\dots N \tag{4.72}$$

where N is the total number of segmentations, $\delta \boldsymbol{p}_{v_{seg}} = \boldsymbol{p}_v - \boldsymbol{p}_{v_{seg_i}} \in \mathbb{R}^3$ are the continuous first partial derivatives; $\boldsymbol{p}_{v_{seg_i}}$ is the reference point of the i^{th} segment on the boundary line with $\dot{\boldsymbol{p}}_{v_{seg_i}}(t) = 0$. Note that (4.72) acts as segmented range for the boundary line whilst the line that lies outside this region is recognised as a non-segmented range. Then, the segmented range error can be formulated as follows

$$\tilde{\boldsymbol{e}}_{seg} = \sum_{i=1}^{N} \frac{k_{seg_i}}{k_q} \max\left(0, f_{seg_i}\left(\delta \boldsymbol{p}_{v_{seg_i}}\right)\right)\left(\frac{\partial f_{seg_i}\left(\delta \boldsymbol{p}_{v_{seg_i}}\right)}{\partial \boldsymbol{p}_{v_{seg}}}\right)^T \tag{4.73}$$

For simplicity of presentation, the controller with exact compensation of restoring forces is employed. Taking into account (4.73) as a secondary objective yields

$$\boldsymbol{\tau} = -K_0 \tilde{\boldsymbol{e}}_T - K_s\left(\tilde{\boldsymbol{e}}_s + \tilde{\boldsymbol{e}}_{seg}\right) - K_v \boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}_2) \tag{4.74}$$

where K_0, K_s and K_v are previously as defined in (4.66).

Since the last term of (4.74) is the exactly known restoring forces, a similar stability analysis to Section 4.3 can be performed to show the asymptotic stability of the proposed region boundary-based controller with the segmentation approach. Thus, the control input of (4.74) results in the convergence of an underwater vehicle to the desired segmented range along the boundary line. When (4.72) is reduced to be arbitrarily small, then the target is specified as a point on a boundary line as depicted in Figure 4.13(b).

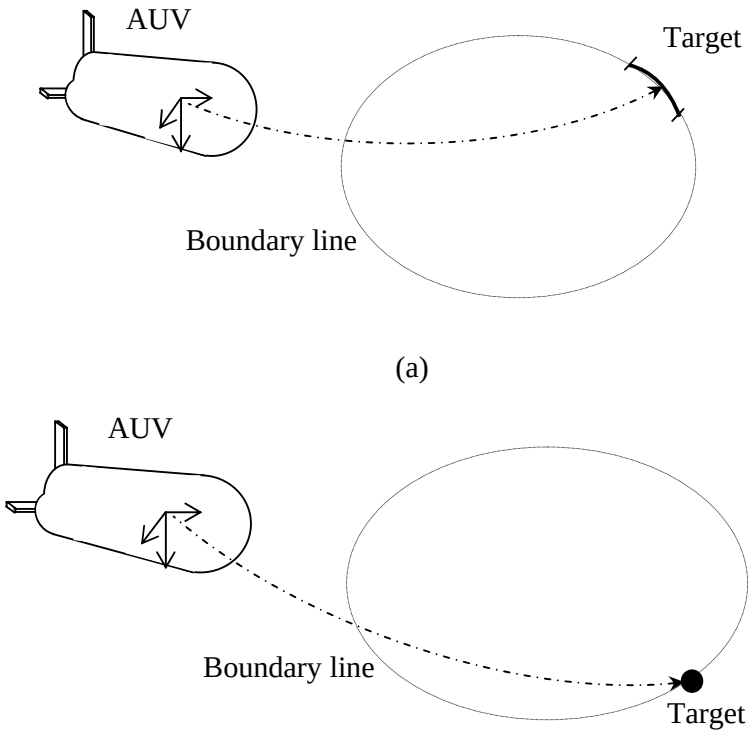


Figure 4.13: Region boundary-based control for an underwater vehicle; (a) Target as a segmented range, (b) Target as a point

4.5.3. Simulation Results

To verify the effectiveness of the region boundary-based control laws, several simulation studies are conducted considering the omni directional intelligent navigator (ODIN) as the underwater vehicle model (*cf.* Appendix C). The orientation in the unit quaternion is kept constant where the initial value is set to $\boldsymbol{e}_v(0) = [1 \quad 0 \quad 0 \quad 0]^T$. In the simulations, the desired outer and inner sub-regions are both specified as spheres which can be represented by the following functions

$$f(\delta \boldsymbol{p}_v) = (x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2 - r_1^2 \leq 0 \tag{4.75}$$

$$g(\delta \boldsymbol{p}_v) = (x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2 - r_2^2 \geq 0 \tag{4.76}$$

where $[x_d \quad y_d \quad z_d]^T = [8 \quad 0 \quad 5]^T$ m. The radii are chosen as $r_1 = r_2 = 1.0$ m.

In the first simulation, the desired restoring forces are assumed to be exactly known, thus the similar concept of the restoring forces compensation in (4.8) can be utilised in (4.65) along with the desired region boundary of (4.75) and (4.76). The vehicle is initialised to position $\boldsymbol{p}_v(0) = [0 \quad 0 \quad 1]^T$m. The simulation results are shown in Figure 4.14 and Figure 4.15. In both figures, the initial position is marked with "**x**". Figure 4.14 shows the convergence of the underwater vehicle into the planar plane of the desired spherical surface and the three-dimensional views within 50 s. The position error is depicted in Figure 4.15. For the best performance of the region boundary-based controller with exact restoring effects compensation, the gains are set to the following:

$$k_{p_1} = 18;\ k_{p_2} = 36;\ k_{q_1} = 9.4;\ K_v = \text{diag}\{90, 90, 90, 40, 40, 40\};$$

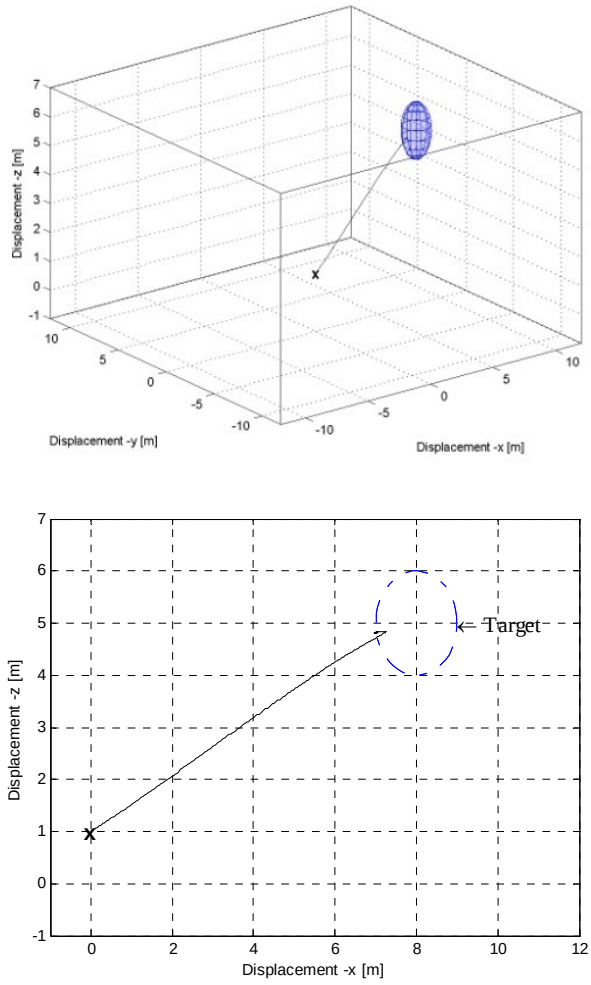


Figure 4.14: Position trajectory of the underwater vehicle (Simulation 1)

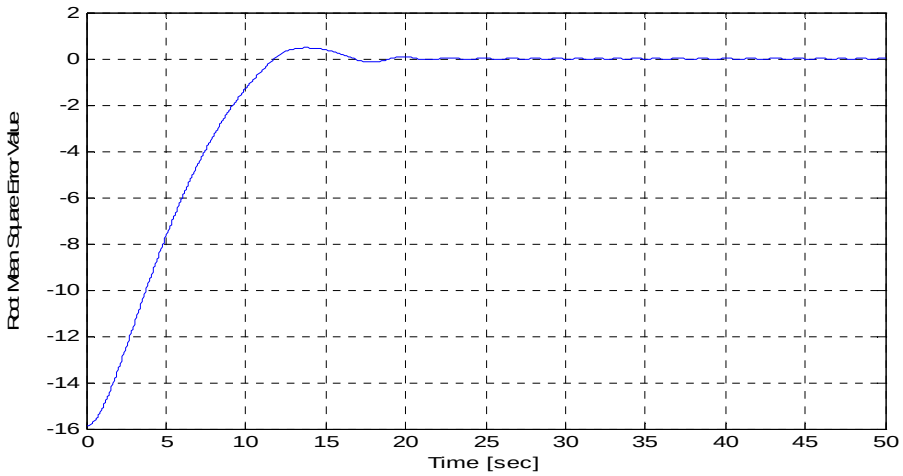


Figure 4.15: Convergence of position error in terms of root mean square error (Simulation 1)

The next simulation is performed to allow the vehicle to be initialised inside the region at $p(0) = [7.5 \quad 0 \quad 4.5]^T$m. Again, the initial position is marked with "**x**". Figures 4.16 and 4.17 show the simulation results where the controller gains are similar to the previous case, except $k_{q_1} = 1.8$. As can be seen from these figures, the proposed region boundary-based control law has been shown to perform effective AUV navigation to the desired boundaries, regardless of where it is started.

An additional simulation was performed to illustrate the effectiveness of the proposed control law (4.65) in the presence of unknown constant parameters in the restoring forces (*cf.* Section 3.4.4 in Chapter 3). The initial position of the vehicle is kept as in the first simulation. The gains for the region boundary-based controller with compensation of the uncertain restoring effects are set to:

$$k_{p1} = 18;\ k_{p2} = 36;\ k_{q1} = 2;\ \Gamma = 2I_4;$$
$$\sigma = 0.05;\ K_v = diag\{90\ 90\ 90\ 40\ 40\ 40\};$$

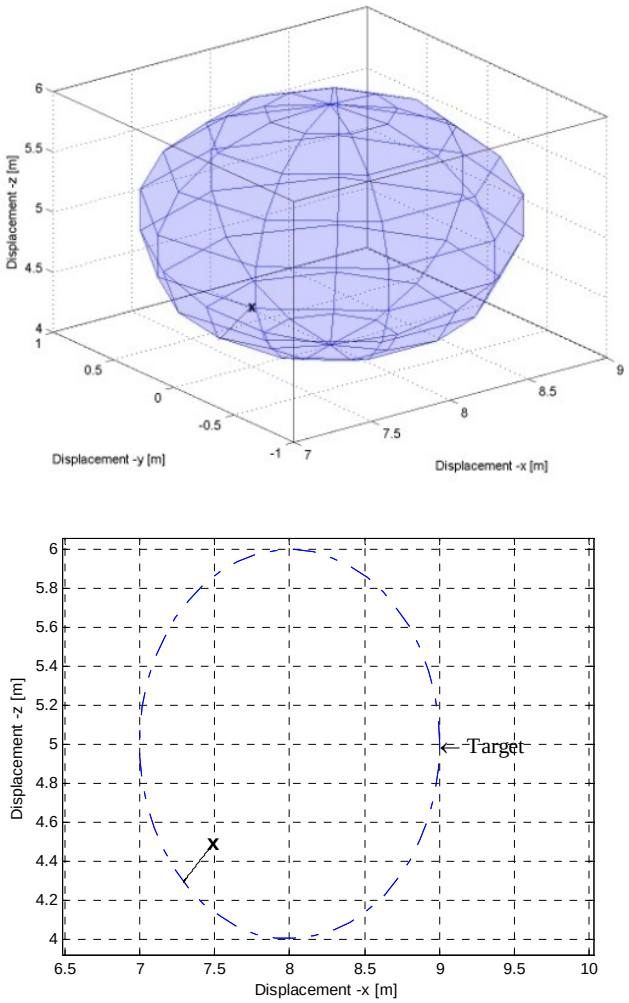


Figure 4.16: Position trajectory of the underwater vehicle (Simulation 2)

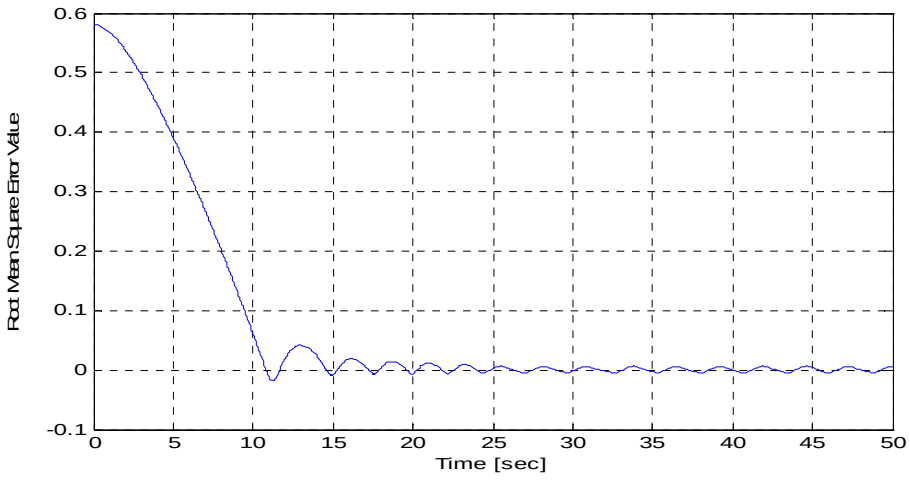


Figure 4.17: Convergence of position error in terms of root mean square error (Simulation 2)

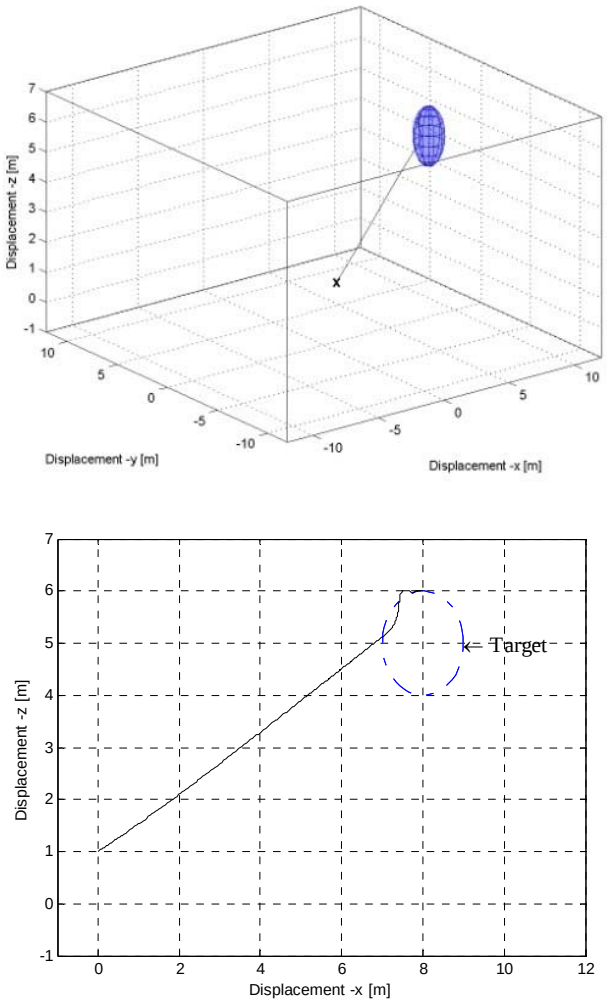


Figure 4.18: Position trajectory of the underwater vehicle (Simulation 3)

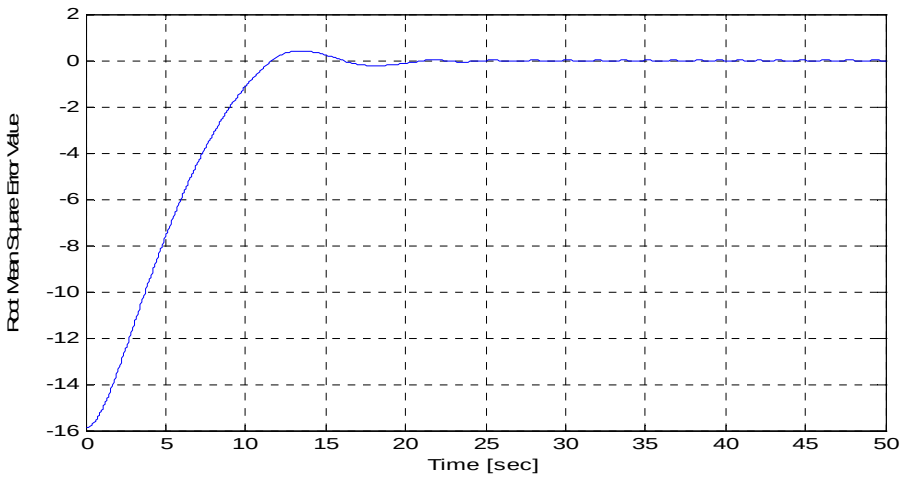
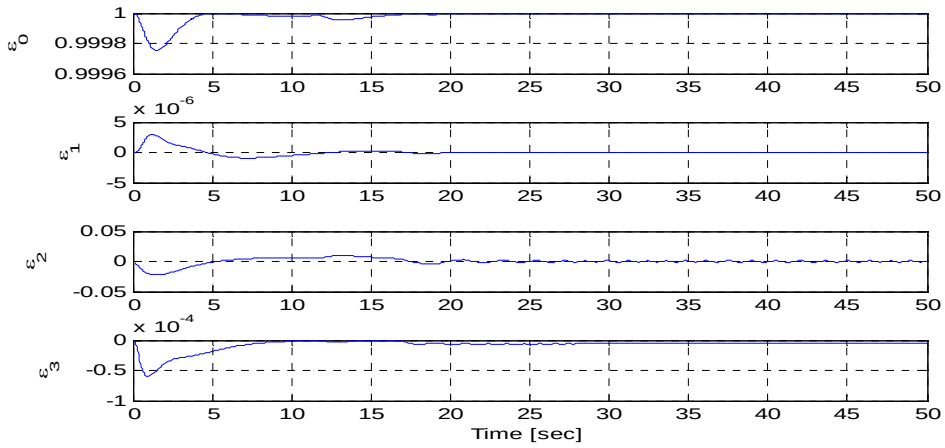
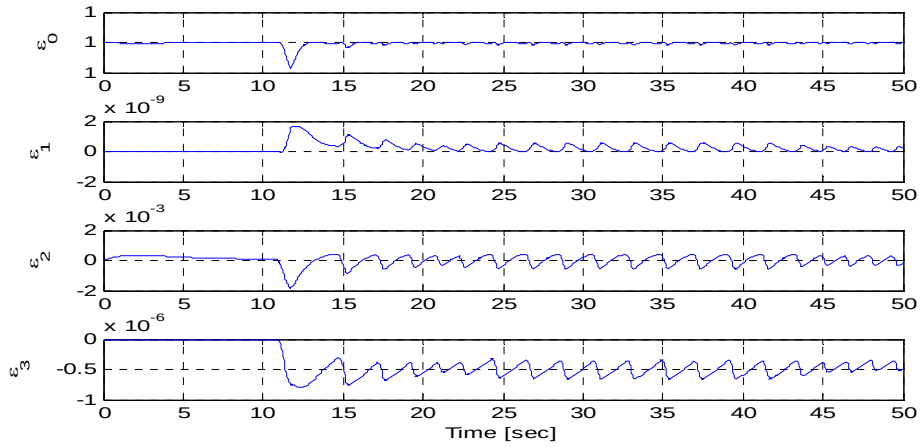


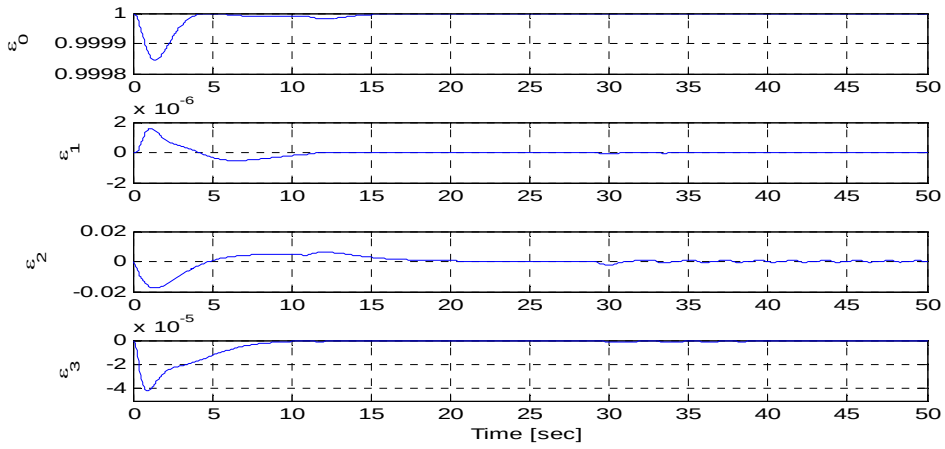
Figure 4.19: Convergence of position error in terms of root mean square error (Simulation 3)



(a) Simulation 1



(b) Simulation 2



(c) Simulation 3

Figure 4.20: Attitude error representations

The vehicle position converged to the desired boundary as shown in Figure 4.18 while the position error expressed in terms of the root mean square error is depicted in Figure 4.19. The attitude representations in all simulations are presented in Figure 4.20.

4.6. Adaptive Sub-Region Boundary-Based Control for a UVMS

In this section, an implementation of the adaptive sub-region boundary-based controller is extended to an underwater vehicle-manipulator system [4.28]. Instead of specifying the desired target as a region or a point, the control objective is defined as a boundary of a region. The region reaching control concept is applicable for controlling an underwater vehicle (macro) with onboard manipulator (micro) system as presented in [4.11]. The underwater vehicle-manipulator task, i.e. maintaining the exterior of a pipeline as in Figure 4.21, utilises the sub-region boundaries as the desired objective.

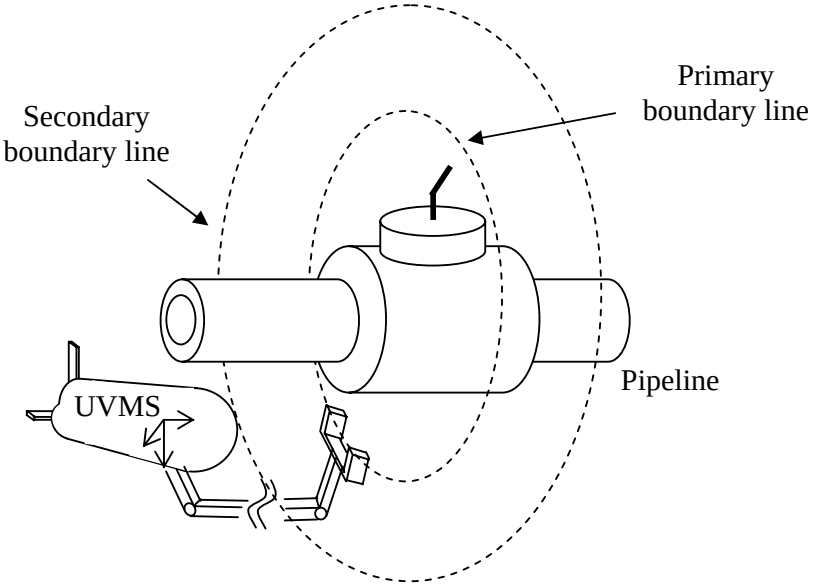


Figure 4.21: A UVMS task - exterior pipeline maintenance

A region boundary-based control concept is regarded as a generalised region or set-point control problem where the system is regulated to move to the region boundary rather than into a region or a point. Therefore, the position of both subsystems (vehicle and the onboard manipulator) can be initialised either from inside or outside of the regions prior to their convergence into the primary (vehicle) and secondary (onboard manipulator) boundary areas within a specific time. Moreover, the proposed control law utilises the least-squares estimation algorithm and the inverse Jacobian matrix in the adaptive law for a mapping of the persistent effects, such as the restoring forces. In the

underwater robot literature, researchers typically prefer gradient-type algorithm for parameter estimation. Thus, the proposed control law in this section utilises a least-squares algorithm for an adaptive region-boundary based control concept that represents a novel departure from the adaptive control of an underwater robot with macro-micro structures.

4.6.1. Problem Formulation

As discussed in the previous section, the desired target for the region boundary-based control approach is specified by at least two sub-regions with different sizes intersecting at the same point. The inner sub-region acts as a repulsive region while the outer sub-region acts as an attractive region. Using this control concept, the vehicle can be initially located within the region or outside the region prior to its convergence into the boundary area. When the inner sub-region size is reduced to zero, the desired boundary is transformed into a desired region. Alternatively, when both inner and outer sub-regions are specified to be arbitrarily small, the desired boundary is transformed into a desired point with a specified accuracy. Therefore, the proposed region boundary-based control concept is also a generalisation of the region or set-point control problem.

For underwater vehicle-manipulator systems, two desired sub-region boundaries are specified, that is a primary region boundary and a secondary region boundary. As previously illustrated in Figure 4.21, the end-effector and the vehicle are kept on the primary (manipulator) and secondary (vehicle) sub-region boundary lines, where it can be ensured that the end-effector operates in the safe working area. The desired outer sub-region for the underwater vehicle can be defined as follows

$$f_v(\delta \boldsymbol{p}_v) \leq 0 \tag{4.77}$$

where $\delta \boldsymbol{p}_v = \boldsymbol{p}_v - \boldsymbol{p}_{vd} \in \mathbb{R}^3$ are the continuous first partial derivatives of the outer sub-region; $\boldsymbol{p}_{vd}$ is the stationary reference point inside the outer sub-region such that $\dot{\boldsymbol{p}}_{vd}(t) = 0$.

The following inequality function can be used for the inner sub-region

$$g_v(\delta \boldsymbol{p}_{vs}) \geq 0 \tag{4.78}$$

where $\delta \boldsymbol{p}_{vs} = \boldsymbol{p}_{vs} - \boldsymbol{p}_{vd} \in \mathbb{R}^3$ are the continuous first partial derivatives of the inner sub-region. To obtain a desired secondary sub-region boundary, (4.77) and (4.78) are defined arbitrarily close to each other, such that

$$f_v(\delta \boldsymbol{p}_v) \approx g_v(\delta \boldsymbol{p}_{vs}) \tag{4.79}$$

Since the control objectives are accomplished concurrently, the position vector $\boldsymbol{p}_{vs}$ can be chosen as $\boldsymbol{p}_v$ and the body-fixed velocity vector for the outer and inner vehicle sub-region can be defined as follows

$$\dot{\boldsymbol{p}}_{vs} = \dot{\boldsymbol{p}}_v = E_1(\boldsymbol{e}_v)\boldsymbol{v} \tag{4.80}$$

In addition, the desired outer sub-region for the manipulator can be defined as follows

$$f_m(\delta\, {}^v\boldsymbol{p}_{v,m}) \leq 0 \tag{4.81}$$

where $\delta\, {}^v\boldsymbol{p}_{v,m} = {}^v\boldsymbol{p}_{v,m} - {}^v\boldsymbol{p}_{v,md} \in \mathbb{R}^3$ are the continuous first partial derivatives of the outer sub-region; ${}^v\boldsymbol{p}_{v,md}$ is the stationary reference point inside the outer sub-region such that ${}^v\dot{\boldsymbol{p}}_{v,md}(t) = 0$. The following inequality function can be used for the inner sub-region

$$g_m(\delta\, {}^v\boldsymbol{p}_{v,ms}) \geq 0 \tag{4.82}$$

where $\delta\, {}^v\boldsymbol{p}_{v,ms} = {}^v\boldsymbol{p}_{v,ms} - {}^v\boldsymbol{p}_{v,md} \in \mathbb{R}^3$ are the continuous first partial derivatives of the inner region. Hence, to obtain a desired primary sub-region boundary, (4.81) and (4.82) are defined to be arbitrarily close to each other, such that

$$f_m(\delta\, {}^v\boldsymbol{p}_{v,m}) \approx g_m(\delta\, {}^v\boldsymbol{p}_{v,ms}) \tag{4.83}$$

The position vector ${}^{v}\boldsymbol{p}_{v,ms}$ can be chosen as ${}^{v}\boldsymbol{p}_{v,m}$ and the body-fixed velocity vector for the inner sub-region can be defined as follows

$$ {}^{v}\dot{\boldsymbol{p}}_{v,ms} = {}^{v}\dot{\boldsymbol{p}}_{v,m} = J_{mp}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{4.84} $$

The reference points $\boldsymbol{p}_{vd}$ and ${}^{v}\boldsymbol{p}_{v,md}$ can be obtained using a method similar to that in [4.29, 4.30] such that the convergence of the end-effector into the primary sub-region boundary as the main task is achieved. Now, define the outer sub-region error $\tilde{\boldsymbol{e}}_{p_v}$ and inner sub-region error $\tilde{\boldsymbol{e}}_{s_v}$ of the vehicle in the following form

$$ \tilde{\boldsymbol{e}}_{p_v} = \max\big(0, f_v(\delta\boldsymbol{p}_v)\big)\left(\frac{\partial f_v(\delta\boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \tag{4.85} $$

$$ \tilde{\boldsymbol{e}}_{s_v} = \max\big(0, g_v(\delta\boldsymbol{p}_{vs})\big)\left(\frac{\partial g_v(\delta\boldsymbol{p}_{vs})}{\partial \boldsymbol{p}_{vs}}\right)^T \tag{4.86} $$

Similarly, the outer sub-region error $\tilde{\boldsymbol{e}}\,{}^{v}{}_{p_{v,m}}$ and inner sub-region error $\tilde{\boldsymbol{e}}\,{}^{v}{}_{s_{v,m}}$ of the manipulator can be defined as

$$ \tilde{\boldsymbol{e}}\,{}^{v}{}_{p_{v,m}} = \max\left(0, f_m\big(\delta\,{}^{v}\boldsymbol{p}_{v,m}\big)\right)\left(\frac{\partial f_m\big(\delta\,{}^{v}\boldsymbol{p}_{v,m}\big)}{\partial\,{}^{v}\boldsymbol{p}_{v,m}}\right)^T \tag{4.87} $$

$$ \tilde{\boldsymbol{e}}\,{}^{v}{}_{s_{v,m}} = \max\left(0, g_m\big(\delta\,{}^{v}\boldsymbol{p}_{v,ms}\big)\right)\left(\frac{\partial g_m\big(\delta\,{}^{v}\boldsymbol{p}_{v,ms}\big)}{\partial\,{}^{v}\boldsymbol{p}_{v,ms}}\right)^T \tag{4.88} $$

As defined in Chapter 3, the error between the actual and desired orientation for the vehicle and manipulator can be generalised as follows [4.25]

$$ \tilde{R} \triangleq RR_d^T = \big(\tilde{e}_0{}^2 - \tilde{\boldsymbol{e}}_\varepsilon{}^T\tilde{\boldsymbol{e}}_\varepsilon\big)I_3 + 2\tilde{\boldsymbol{e}}_\varepsilon\tilde{\boldsymbol{e}}_\varepsilon{}^T - 2\tilde{e}_0\tilde{\boldsymbol{e}}_\varepsilon{}^\times \tag{4.89} $$

where R is the rotational matrix and R_d is the rotational matrix of R expressing the desired orientation which is also described by the quaternion $\boldsymbol{\epsilon} \triangleq \begin{bmatrix}\epsilon_{0,d} & \boldsymbol{\epsilon}_{\varepsilon,d}^T\end{bmatrix}^T$. The

corresponding unit quaternion representation is denoted by $\tilde{\boldsymbol{e}} \triangleq [\tilde{e}_0 \quad \tilde{\boldsymbol{e}}_\varepsilon^T]^T$. Thus, the quaternion propagation equation can be considered as

$$\begin{aligned} \dot{\tilde{e}}_0 &= -\frac{1}{2}\tilde{\boldsymbol{e}}_\varepsilon^T \tilde{\boldsymbol{\omega}} \\ \dot{\tilde{\boldsymbol{e}}}_\varepsilon &= \frac{1}{2}(\tilde{e}_0 I_3 + \tilde{\boldsymbol{e}}_\varepsilon{}^\times)\tilde{\boldsymbol{\omega}} \end{aligned} \tag{4.90}$$

where $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} - \boldsymbol{\omega}_d$; $\boldsymbol{\omega}$ is defined in Chapter 3 for both the vehicle and manipulator. The desired angular velocity of the vehicle/manipulator $\boldsymbol{\omega}_d$ is always zero such that $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega}$.

Based on the preceding error definitions, the sub-region boundary-based controller with uncertain persistent effects compensation for an underwater vehicle-manipulator system is proposed as follows

$$\boldsymbol{\tau} = -K_0 K_p \tilde{\boldsymbol{e}}_{T_p} - K_0 K_s \tilde{\boldsymbol{e}}_{T_s} - K_{\hat{\xi}} \boldsymbol{\xi} + Z(\boldsymbol{\eta}_e, \boldsymbol{q})\hat{\boldsymbol{\varphi}} \tag{4.91}$$

with

$$\begin{aligned} K_0 &= diag\{E_1^T, I_{3\times3}, J_M^T\} \in \mathbb{R}^{(6+n)\times12}; \\ K_p &= diag\left\{k_{v_{p1}} I_{3\times3}, k_{v_{p2}} I_{3\times3}, k_{m_{p1}} I_{3\times3}, k_{m_{p2}} I_{3\times3}\right\} \in \mathbb{R}^{12\times12}; \\ K_s &= diag\{k_{v_s} I_{3\times3}, 0_{3\times3}, k_{m_s} I_{3\times3}, 0_{3\times3}\} \in \mathbb{R}^{12\times12}; \end{aligned} \tag{4.92}$$

where $k_{v_{p1}}$, $k_{v_{p2}}$, $k_{m_{p1}}$, $k_{m_{p2}}$, k_{v_s} and k_{m_s} are all positive constants. $K_{\hat{\xi}} \in \mathbb{R}^{(6+n)\times(6+n)}$ is a positive definite and diagonal gain matrix. The errors in the control law (4.91) are defined as $\tilde{\boldsymbol{e}}_{T_p} = \left[\tilde{\boldsymbol{e}}_{p_v}^T \quad \tilde{\boldsymbol{e}}_{\varepsilon_v}^T \quad \tilde{\boldsymbol{e}}^T{}_{v_{p_{v,m}}} \quad \tilde{\boldsymbol{e}}^T{}_{v_{\varepsilon_{v,m}}}\right]^T$; the subscripts p_v and ε_v denote the vehicle position and unit quaternion of the vehicle outer sub-region, respectively, while ${}^v p_{v,m}$ and ${}^v \varepsilon_{v,m}$ are the manipulator position and unit quaternion of the manipulator outer sub-region, respectively, and $\tilde{\boldsymbol{e}}_{T_s} = \left[\tilde{\boldsymbol{e}}_{s_v}^T \quad 0_{1\times3} \quad \tilde{\boldsymbol{e}}^T{}_{v_{s_{v,m}}} \quad 0_{1\times3}\right]^T$; the subscript s_v represents the vehicle position of the vehicle inner sub-region while ${}^v s_{v,m}$ is the manipulator position of the manipulator inner sub-region.

Accordingly, the estimate vector $\widehat{\boldsymbol{\varphi}}$ can be obtained using the least-squares update law as follows

$$\dot{\widehat{\boldsymbol{\varphi}}} = -\Gamma Z^{\mathrm{T}}(\boldsymbol{\eta}_e, \boldsymbol{q})\left(\boldsymbol{\xi} + \sigma J_T^{\dagger}\left(\boldsymbol{e}_v,\ {}^{v}\boldsymbol{e}_{v,m}, \boldsymbol{q}\right)\tilde{\boldsymbol{e}}_T\right) + \Gamma Z^{\mathrm{T}}(\boldsymbol{\eta}_e, \boldsymbol{q}) Z(\boldsymbol{\eta}_e, \boldsymbol{q})\widetilde{\boldsymbol{\varphi}} \tag{4.93}$$

with $J_T^{\dagger} = diag\{J_{v_e}^{\dagger}(\boldsymbol{e}_v), J_M^{\dagger}\left({}^{v}\boldsymbol{e}_{v,m}, \boldsymbol{q}\right)\} \in \mathbb{R}^{(6+n)\times 12}$ and

$$\tilde{\boldsymbol{e}}_T = \tilde{\boldsymbol{e}}_{T_p} + K_T \tilde{\boldsymbol{e}}_{T_s} \tag{4.94}$$

where σ is a positive constant and $J_{v_e}^{\dagger}(\boldsymbol{e}_v)$ and $J_M^{\dagger}\left({}^{v}\boldsymbol{e}_{v,m}, \boldsymbol{q}\right)$ are the pseudo-inverse of the matrices $J_{v_e}(\boldsymbol{e}_v)$ and $J_M\left({}^{v}\boldsymbol{e}_{v,m}, \boldsymbol{q}\right)$, respectively. The square matrix K_T is given by $K_T = diag\left\{\left(\frac{k_{v_s}}{k_{v_{p1}}}\right) I_3, 0_3, \left(\frac{k_{m_s}}{k_{m_{p1}}}\right) I_3, 0_3\right\} \in \mathbb{R}^{12\times 12}$ while $\Gamma(t) \in \mathbb{R}^{n_p \times n_p}$ is a least-squares estimation gain matrix designed as follows

$$\frac{d}{dt}(\Gamma^{-1}) \triangleq Z^{\mathrm{T}}(\boldsymbol{\eta}_e, \boldsymbol{q}) Z(\boldsymbol{\eta}_e, \boldsymbol{q}); \qquad \Gamma(0) = \Gamma^{\mathrm{T}}(0) > 0; \tag{4.95}$$

Remark 4.1: When $\Gamma^{-1}(t_0)$ is chosen to be positive definite and symmetric, then it is clear that $\Gamma(t_0)$ is also positive definite and symmetric. Hence, it follows that both $\Gamma^{-1}(t)$ and $\Gamma(t)$ will remain positive definite and symmetric $\forall t$. From (4.95), the following expression can be obtained

$$\dot{\Gamma} = -\Gamma Z^{\mathrm{T}}(\boldsymbol{\eta}_e, \boldsymbol{q}) Z(\boldsymbol{\eta}_e, \boldsymbol{q})\Gamma \tag{4.96}$$

According to (4.96), it is shown that $\dot{\Gamma}(t)$ is negative semidefinite; therefore, the estimation gain matrix $\Gamma(t)$ is always constant or decreasing, and leads to the boundedness of $\Gamma(t)$ [4.10, 4.31]. To this end, the closed-loop equation can be obtained by substituting (4.91) into (3.63) to yield

$$M(\boldsymbol{q})\dot{\boldsymbol{\xi}} + C(\boldsymbol{q}, \boldsymbol{\xi})\boldsymbol{\xi} + D(\boldsymbol{q}, \boldsymbol{\xi})\boldsymbol{\xi} + K_0 K_p \tilde{\boldsymbol{e}}_{T_p} + K_0 K_s \tilde{\boldsymbol{e}}_{T_s} + K_{\xi}\boldsymbol{\xi} + Z(\boldsymbol{\eta}_e, \boldsymbol{q})\widetilde{\boldsymbol{\varphi}} = 0 \tag{4.97}$$

where $\widetilde{\boldsymbol{\varphi}} = \boldsymbol{\varphi} - \widehat{\boldsymbol{\varphi}}$.

Now, consider the following conditions when choosing the feedback gains to ensure the Lyapunov function is valid locally [4.3]

$$\frac{\left(k_{v_{p1}}+\sigma k_{\xi_{v1}}\right)}{\left\|\frac{\partial f_v(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right\|^2}-2\sigma^2\lambda_m>0;\ \frac{\left(k_{m_{p1}}+\sigma k_{\xi_{m1}}\right)}{\left\|\frac{\partial f_m(\delta\ {}^v\boldsymbol{p}_{v,m})}{\partial\ {}^v\boldsymbol{p}_{v,m}}\right\|^2}-2\sigma^2\lambda_m>0;$$

$$\frac{\left(k_{v_s}+\sigma k_{w_v}\right)}{\left\|\frac{\partial g_v(\delta \boldsymbol{p}_{vs})}{\partial \boldsymbol{p}_{vs}}\right\|^2}-2\sigma^2\lambda_m>0; \tag{4.98}$$

$$\left(k_{m_s}+\sigma k_{w_m}\right)/\left\|\frac{\partial g_m(\delta\ {}^v\boldsymbol{p}_{v,ms})}{\partial\ {}^v\boldsymbol{p}_{v,ms}}\right\|^2-2\sigma^2\lambda_m>0;$$

$$k_{v_{p2}}+\sigma k_{\xi_{v2}}-\sigma^2\lambda_m>0;\ \ k_{m_{p2}}+\sigma k_{\xi_{m2}}-\sigma^2\lambda_m>0;$$

and also,

$$\lambda_{min}[D(\boldsymbol{q},\boldsymbol{\xi})]+\lambda_{min}\left[K_\xi\right]-\sigma c_0>0;$$
$$\lambda_{min}\left[J_T^{\dagger T}K_0K_p\right]>0; \tag{4.99}$$
$$\lambda_{min}\left[J_T^{\dagger T}K_0K_s\right]>0;$$

where $k_{w_v}=k_{v_s}k_{\xi_{v1}}/k_{v_{p1}}$ and $k_{w_m}=k_{m_s}k_{\xi_{m1}}/k_{m_{p1}}$; $k_{\xi_{v1}}$ and $k_{\xi_{m1}}$ are the components of K_ξ. $\lambda_m \triangleq \lambda_{max}\left[J_T^{\dagger T}M(\boldsymbol{q})J_T^{\dagger}\right]$ and $c_0>0$ is a constant. Therefore, the next theorem can be stated as:

Theorem 4.4: The proposed control law described in (4.91) and the least-squares update law defined in (4.93), (4.94) and (4.95) guarantee that $\boldsymbol{\xi}$, $\tilde{\boldsymbol{e}}_{T_p}$ and $\tilde{\boldsymbol{e}}_{T_s}$ are driven to zero as $t\to\infty$ for an underwater vehicle mounted with a manipulator given by (3.63), provided that the feedback gains are chosen to satisfy conditions (4.98) and (4.99).

Proof: See Appendix B.

4.6.2. Simulation Results

A simulation study is performed to assess the effectiveness of the proposed boundary-based control law for an underwater vehicle-manipulator system. An omni directional intelligent navigator (ODIN) mounted with a revolute joint two-link manipulator is

chosen for the numerical simulation. The parameter models for ODIN are given in Appendix C. The desired outer and inner sub-regions for the vehicle are specified as

$$f_v(\delta \boldsymbol{p}_v) = (x_v - x_{vd})^2 + (y_v - y_{vd})^2 + (z_v - z_{vd})^2 - r_v^2 \leq 0 \tag{4.100}$$

$$g_v(\delta \boldsymbol{p}_{vs}) = (x_{vs} - x_{vd})^2 + (y_{vs} - y_{vd})^2 + (z_{vs} - z_{vd})^2 - r_{vs}^2 \geq 0 \tag{4.101}$$

where these functions act as the secondary sub-region boundary. The desired position is set to $[x_{vd} \quad y_{vd} \quad z_{vd}]^T = [7 \quad 0 \quad 2]^T$ m and the radii are chosen as $r_v = r_{vs} = 0.5$ m. Meanwhile, the desired outer and inner sub-regions for the manipulator are specified as

$$f_m(\delta\, {}^v\boldsymbol{p}_{v,m}) = \left({}^vx_{v,m} - {}^vx_{v,md}\right)^2 + \left({}^vz_{v,m} - {}^vz_{v,md}\right)^2 - {}^vr_{v,m}^2 \leq 0 \tag{4.102}$$

$$g_m(\delta\, {}^v\boldsymbol{p}_{v,ms}) = \left({}^vx_{v,ms} - {}^vx_{v,md}\right)^2 + \left({}^vz_{v,ms} - {}^vz_{v,md}\right)^2 - {}^vr_{v,ms}^2 \geq 0 \tag{4.103}$$

where these functions act as the primary sub-region boundary. Since the manipulator has two degrees-of-freedom, only the end-effector position in the *x*-axis and *z*-axis is controllable. Therefore, the position in the *y*-axis follows the position of the vehicle and hence, the gain $k_{m_{p2}}$ is set to zero. The desired position is set to $\left[{}^vx_{v,md} \quad {}^vz_{v,md}\right]^T = [0.31 \quad -0.48]^T$ m and the tolerances are chosen as ${}^vr_{v,m} = {}^vr_{v,ms} = 0.05$ m.

Note that, only the position vector is specified as a boundary since the operator is incapable of observing the boundary reaching of the orientation in the Cartesian space. Hence, the vehicle orientation in the unit quaternion is kept constant where the initial value is set to orientation $e_v(0) = [0 \quad 0 \quad 0 \quad 1]^T$. In this simulation, the vehicle and the manipulator are initialised to $\boldsymbol{p}_v(0) = [2 \quad 0 \quad 1]^T$m and $\boldsymbol{q}(0) = [0.71 \quad -1.42]^T$rad, respectively.

Figure 4.22 shows the convergence of an underwater vehicle and its onboard manipulator into their respective desired sub-region boundaries in a planar plane view within 40 s. Figure 4.23 illustrates the final positions in a close-up view. In both figures, the initial positions of the vehicle (solid lines) and manipulator (dash-dot lines) are marked with "**x**". The position errors in terms of root-mean-square errors for both the vehicle and manipulator are depicted in Figure 4.24(a) and Figure 4.24(b), respectively.

For the best performance of the sub-region boundary-based controller with persistent effects compensation, the gains are obtained using a trial and error approach and are set to the following:

$$k_{v_{p1}} = 88.8, ; k_{v_{p2}} = 36; k_{m_{p1}} = 40; k_{m_{p2}} = 0; k_{v_s} = 8.8; k_{m_s} = 4; \sigma = 0.08;$$
$$\Gamma(0) = diag\{1.0, 1.0, 1.0, 1.0, 0.1, 0.1\}; K_\xi = diag\{140, 140, 140, 40, 40, 40, 10, 10\};$$

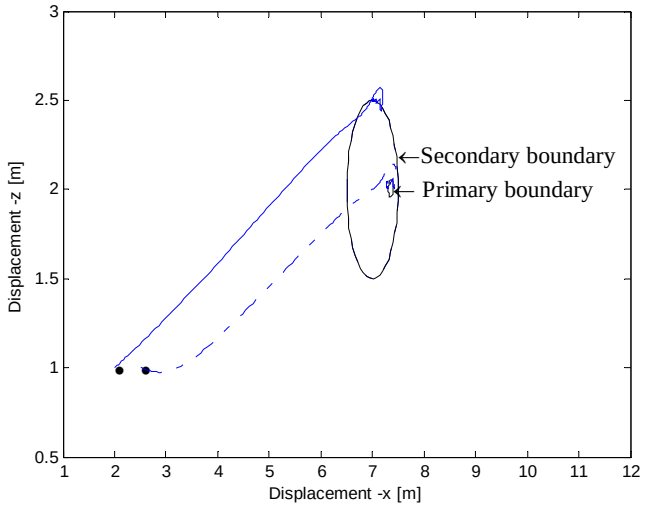


Figure 4.22: A planar plane of the UVMS position

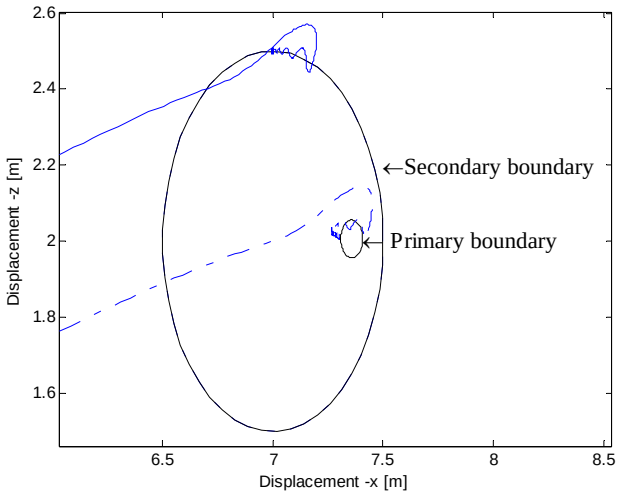
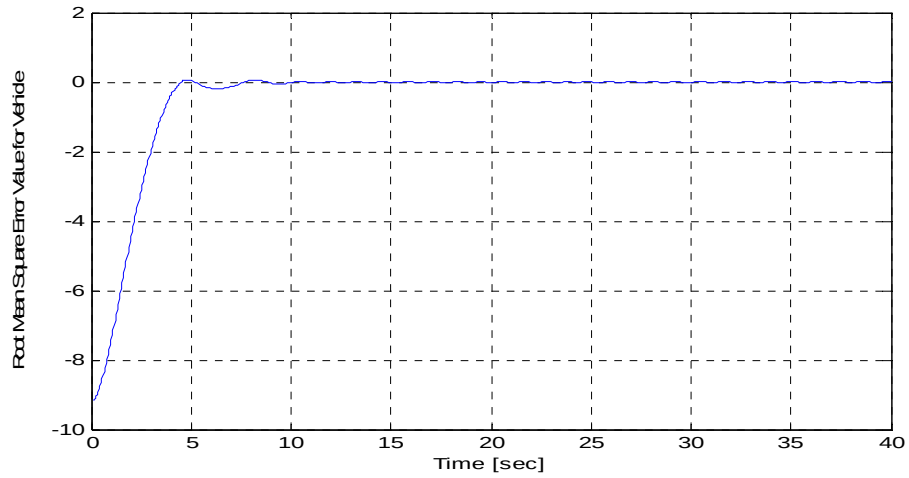
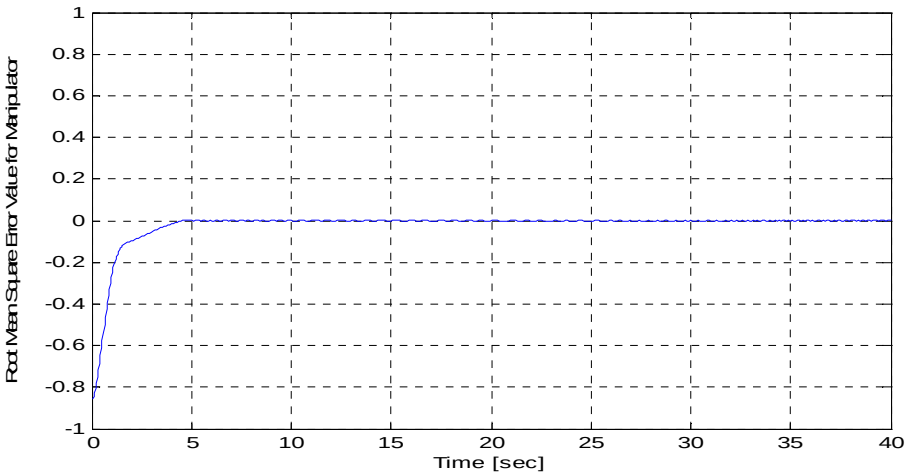


Figure 4.23: A close-up view of the final position



(a) Underwater vehicle



(b) Onboard manipulator

Figure 4.24: Convergence of position errors in terms of root mean square error

4.7. Summary

This chapter has presented a series of new regulation controllers and their simulation results in terms of the task-space which have been specifically designed for underwater robotic systems. The benefits of using new regulation controllers over conventional set-point controllers have been clearly explained. Moreover, the stability analyses of the proposed controllers in the Lyapunov sense were also performed. Note that, the proofs of theorems in this chapter are presented in Appendix B.

Firstly, the background of the conventional set-point and region reaching controllers was reported. The control law for the region reaching approach was also briefly formulated.

The sub-region priority reaching controller was then proposed for an autonomous underwater vehicle. This encompassed the region-decomposition method which is very useful when specifying the particular sub-regions as motion constraints.

An adaptive-fuzzy sub-region priority reaching controller was then formulated to deal with the uncertainties of the restoring forces and to manage the multiple sub-regions effectively. The obstacle avoidance, depth constraint and region complexity were among the selected inputs for the fuzzy inference system.

Several new control structures derived from the region boundary approach were also presented which act as alternative approaches to control the underwater robot when reaching the desired region boundary rather than into a region or a point. In addition, the least-squares estimation algorithm was performed as a novel departure from the adaptive control of an underwater robot.

Simulation results were presented within each section that demonstrate the effectiveness of each of the proposed regulation control laws. Some of the results showed the ability of the adaptive control strategies to cope with unknown and changing restoring forces.

4.8. References

[4.1] S. Arimoto , *Control Theory of Nonlinear Mechanical Systems - A Passivity-Based and Circuit-Theoretic Approach*.: Oxford: Clarendon Press, 1996.

[4.2] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, 1st ed. New York: John Wiley and Sons, 1994.

[4.3] Y. C. Sun and C. C. Cheah, "Adaptive Setpoint Control for Autonomous Underwater Vehicles," in *Proc. 42nd IEEE Conference on Decision and Control*, Maui, Hawaii USA, 2003, pp. 1262 - 1267.

[4.4] P. Herman, "Decoupled PD Set-Point Controller for Underwater Vehicles," *Ocean Engineering*, vol. 36, pp. 529-534, 2009.

[4.5] M. Takegaki and S. Arimoto, "A New Feedback Method for Dynamic Control of Manipulators," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 102, pp. 119–125, 1981.

[4.6] T. I. Fossen and S. I. Sagatun, "Adaptive Control of Nonlinear Systems: A Case Study of Underwater Robotic Systems," *Journal of Robotic Systems*, vol. 8, pp. 393-412, 1991.

[4.7] G. Antonelli, S. Chiaverini, N. Sarkar and M. West, "Adaptive Control of an Autonomous Underwater Vehicle: Experimental Results on ODIN," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 5, pp. 756 - 765, 2001.

[4.8] S. Zhao and J. Yuh, "Adaptive DOB Control of Underwater Robots," in *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, 2003, pp. 571-576.

[4.9] Y. C. Sun and C. C. Cheah, "Adaptive Control Schemes for Autonomous Underwater Vehicle," *Robotica*, vol. 27, pp. 119-129, 2009.

[4.10] M. de Queiroz, D. Dawson, S. Nagarkatti, and F. Zhang, *Lyapunov-Based Control of Mechanical Systems*. Boston, MA: Birkhauser, 1999.

[4.11] C. C. Cheah and Y. C. Sun, "A Region Reaching Control Scheme for Underwater Vehicle-Manipulator Systems," in *Proc. IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007, pp. 4576-4579.

[4.12] C. C. Cheah, "Region Reaching Control of Robots with Motion Constraints," in *Proc. 10th Int. Conf. on Control, Automation, Robotics and Vision*, Hanoi, Vietnam, 2008, pp. 1752-1757.

[4.13] R. Sutton and I. M. Jess, "A Design Study of a Self-Organising Fuzzy Autopilot for Ship Control," in *Proceedings of the IMechE*, vol. 205, 1991, pp. 35–45.

[4.14] M. N. Polkinghorne, G. N. Roberts, R. S. Burns, and D. Winwood, "The Implementation of Fixed Rulebase Fuzzy Logic to the Control of Small Surface Ships," *Control Engineering Practice*, vol. 3, pp. 321-328, 1995.

[4.15] E. Omerdic, G. N. Roberts, and Z. Vukic, "A Fuzzy Track-Keeping Autopilot for Ship Steering," *Proceedings of IMarEST - Part A - Journal of Marine Engineering and Technology*, pp. 23-35, 2003.

[4.16] F. Song and S.M.Smith, "Design of Sliding Mode Fuzzy Controllers for an Autonomous Underwater Vehicle without System Model," in *MTS/IEEE Oceans*, 2000, pp. 835-840.

[4.17] J. Guo, F. C. Chiu, and C. C. Huang, "Design of a Sliding Mode Fuzzy Controller for the Guidance and Control of an Autonomous Underwater Vehicle," *Ocean Engineering*, vol. 30, pp. 2137-2155, 2003.

[4.18] W. M. Bessa, M. S. Dutra and E. Kreuzer, "Depth Control of Remotely Operated Underwater Vehicles using an Adaptive Fuzzy Sliding Mode Controller," *Robot. Auton. Syst.*, vol. 56, pp. 670-677, 2008.

[4.19] S. Khanmohammadi, G. Alizadeh, and M. Poormahmood, "Design of a Fuzzy Controller for Underwater Vehicles to Avoid Moving Obstacles," in *IEEE Int. Fuzzy Systems Conf., 2007. FUZZ-IEEE 2007*, 2007, pp. 1-6.

[4.20] N. Zhao, D. Xu, J. Gao, and W. Yan, "Fuzzy Behavioral Navigation for Bottom Collision Avoidance of Autonomous Underwater Vehicles," in *Proc. of the 1st Int. Conf. on Intelligent Robotics and Applications: Part I*, Wuhan, China, 2008, pp. 122-130.

[4.21] V. Kanakakis, K. P. Valavanis and N. C. Tsourveloudis, "Fuzzy-Logic Based Navigation of Underwater Vehicles," *J. Intell. Robotics System*, vol. 40, pp. 45-88, 2004.

[4.22] P.-S. Lee and L.-L. Wang, "Collision Avoidance by Fuzzy Logic Control for Automated Guided Vehicle Navigation," *Journal of Robotic Systems*, vol. 11, pp. 743-760, 1994.

[4.23] Z. H. Ismail and M. W. Dunnigan, "A Sub-Region Priority Reaching Control Scheme With A Fuzzy-Logic Algorithm For An Underwater Vehicle Subject To Uncertain Restoring Forces," in *Proceedings of the OCEANS'10 IEEE Sydney*.

[4.24] D. Driankov, H. Hellendoorn and M.Reinfrank, *An Introduction to Fuzzy Control*. Berlin, Germany: Springer-Verlag, 1995.

[4.25] G. Antonelli, *Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems*. Germany: Springer-Verlag Berlin, 2003.

[4.26] Z. H. Ismail and M. W. Dunnigan, "A Region Boundary-Based Control Scheme for an Autonomous Underwater Vehicle," Submitted to Journal of Ocean Engineering, 2010.

[4.27] Z. H. Ismail and M. W. Dunnigan, "An Adaptive Region Boundary-Based Control Scheme for an Autonomous Underwater Vehicle," , Accepted in the 11th International Conference on Control, Automation, Robotics and Vision, 2010.

[4.28] Z. H. Ismail and M. W. Dunnigan, "A Sub-Region Boundary-Based Control Scheme with a Least-Squares Estimation Algorithm for an Underwater Robotic System," , Accepted in the 11th International Conference on Control, Automation, Robotics and Vision, 2010.

[4.29] F. Lizarralde and J. T. Wen, "Attitude Control Without Angular Velocity Measurement: A Passivity Approach," in *IEEE Transactions on Automatic Control*, vol. 41, 1996, pp. 468-472.

[4.30] Y. Yamamoto and X. Yun, "Modeling and Compensation of the Dynamic Interaction of a Mobile Manipulator," in *Proc. IEEE Conf. Robotics and Automation*, vol. 1, 1994, pp. 1-6.

[4.31] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*. New York, NY: John Wiley and Sons, 1995.

CHAPTER 5
TASK-SPACE TRACKING CONTROL FOR REDUNDANT UNDERWATER ROBOTS

5.1. Introduction

As explained in Chapter 2, many research efforts have been devoted to the development of tracking control strategies for redundant underwater robots and it is suggested that the control objective can be achieved if the control command satisfies the following procedure. First, consider the velocity vector as $\boldsymbol{\xi} = [\boldsymbol{v} \quad \dot{\boldsymbol{q}}]^T$ and recall the dynamic model of an underwater vehicle-manipulator system defined in (3.63)

$$M(\boldsymbol{q})\dot{\boldsymbol{\xi}} + C(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + D(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + \boldsymbol{g}(\boldsymbol{\eta}_e,\boldsymbol{q}) = \boldsymbol{\tau} \tag{5.1}$$

where $M(\boldsymbol{q}) \in \mathbb{R}^{(6+n)\times(6+n)}$ is the inertia matrix including added mass, $C(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} \in \mathbb{R}^{(6+n)}$ is the vector of Coriolis and centripetal terms, $D(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} \in \mathbb{R}^{(6+n)}$ is the vector of hydrodynamic damping, $\boldsymbol{g}(\boldsymbol{\eta}_e,\boldsymbol{q}) \in \mathbb{R}^{(6+n)}$ is the vector of gravity and buoyancy forces, $\boldsymbol{\tau} \in \mathbb{R}^{(6+n)}$ is the vector of generalised forces acting on the vehicle and joint torques. Note that, $\dot{\boldsymbol{\xi}} \in \mathbb{R}^{(6+n)}$ denotes the acceleration vector and $[\boldsymbol{\eta}_e^T \quad \boldsymbol{q}^T]^T \in \mathbb{R}^{(6+n)}$ represents the position and orientation vector.

Then, taking into account (5.1) and a set of bounded functions of time $\boldsymbol{\eta}_{ed}(t)$, $\boldsymbol{q}_d(t)$, $\boldsymbol{\xi}_d(t)$ and $\dot{\boldsymbol{\xi}}_d(t)$, the objective of tracking control consists in finding a control command $\boldsymbol{\tau}$ such that

$$\lim_{t\to\infty} \tilde{\boldsymbol{e}}_{\eta_e}(t) = 0 \quad \text{and} \quad \lim_{t\to\infty} \tilde{\boldsymbol{e}}_q(t) = 0 \tag{5.2}$$

where $\tilde{\boldsymbol{e}}_{\eta_e} \coloneqq \boldsymbol{\eta}_{ed}(t) - \boldsymbol{\eta}_e(t) \in \mathbb{R}^6$ represents for the vehicle position and orientation error and $\tilde{\boldsymbol{e}}_q \coloneqq \boldsymbol{q}_d(t) - \boldsymbol{q}(t) \in \mathbb{R}^n$ is the joint position error vector. The subscript d denotes the desired value of the vehicle and manipulator variables. Then, the velocity error can be defined as

$$\dot{\tilde{\boldsymbol{e}}}_{\xi}(t) = \boldsymbol{\xi}_d(t) - \boldsymbol{\xi}(t) \tag{5.3}$$

Using the Lyapunov-type approach that was explained in Appendix A for stability analysis, it can be concluded that the control objective is achieved if the vehicle and manipulator variables asymptotically follow the trajectory of the desired motion.

In this chapter, several task-space tracking control laws are proposed for underwater vehicle-manipulator systems. At the beginning of the chapter, an overview of set-point and region tracking control scheme in the task-space formulation is briefly discussed. After that, a novel tracking control scheme for a UVMS is presented where the proposed controller is not only used to track the prescribed sub-region but also allows the use of self-motion to perform various sub-tasks (i.e. drag minimisation, obstacle avoidance and manipulability) as the system is kinematically redundant. In the proposed control scheme, the desired primary task of the UVMS is specified as two sub-regions that are assigned for the vehicle and end-effector. Despite the parametric uncertainty associated with the underwater dynamic model, the controller ensures the sub-task tracking without affecting the sub-region and attitude tracking control objective. The Lyapunov type approach is utilised to design the controller.

Then, an extension to an adaptive-robust control scheme with multiple sub-regions and sub-task objectives is also performed to illustrate the flexibility of the approach. The presence of variable ocean currents creates hydrodynamic forces and moments that are not well-known or predictable, even though they are bounded. Therefore, the control task of tracking a prescribed sub-region trajectory is challenging due to these additive bounded disturbances. Furthermore, multiple sub-task criteria which are formulated using a weighted-sum approach are added to the control objective.

5.2. Task-Space Tracking Control Strategies

The desired path for the underwater robot is commonly specified in a task-space such as Cartesian space. In this case, a continuous curve, or path in the task-space which is parameterised in time, is available to achieve a desired task. The motion control problem then consists of making the robot follow the trajectory as closely as possible. This can be referred to as task-space trajectory tracking. In order to allow the underwater robot to track the desired trajectory in task-space (i.e. Cartesian space), an inverse kinematics problem needs to be solved to generate the desired angle in joint

space for manipulator and desired position/orientation in the body-fixed frame for the vehicle. However, the need to solve the inverse kinematics problem is eliminated if the control problem is formulated directly in task-space.

It is well known that to achieve better performance, it is imperative to incorporate the nonlinear robot dynamics into the controller design. However, underwater robot dynamics not only exhibit significant nonlinearities but also have the problem of uncertainty. To deal with trajectory tracking control problems in the presence of dynamic uncertainty, several advanced tracking controllers have been investigated in recent years as thoroughly discussed in Chapter 2. Within this section, two types of task-space trajectory tracking, specifically set-point and region tracking control are briefly explained. It is interesting to note that the set-point defines its desired target as a moving point while the latter defines it as a moving region.

5.2.1 Set-Point Tracking Control

To illustrate set-point tracking control, consider the dynamic model of an underwater vehicle defined in (3.54) without the presence of environmental forces and moments as follows

$$M_v\dot{\boldsymbol{v}} + C_v(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \tag{5.4}$$

where M_v is the inertia matrix including the added mass term, $C_v(\boldsymbol{v})$ represents the matrix of the Coriolis and centripetal forces including the added mass term, $D(\boldsymbol{v})$ denotes the hydrodynamic damping and lift force, and $\boldsymbol{g}(\boldsymbol{\eta})$ is the restoring force. The vectors $\boldsymbol{\eta}$, $\boldsymbol{v}$, $\dot{\boldsymbol{v}}$ denote the position/orientation, velocity and acceleration respectively.

From (5.4), the equation of motion is expressed in the body-fixed frame of the underwater vehicle because it is convenient to measure and control the motion of the system with respect to the moving frame. However, the integration of the angular velocity vector does not lead to generalised coordinates denoting the orientation of the underwater vehicle. Generally, the derivative of the generalised coordinates and the velocity vector in the body-fixed frame can be related through the following linear transformation

$$\dot{\boldsymbol{\eta}} = B\boldsymbol{\nu} \tag{5.5}$$

where $B = J_{v_a} \in \mathbb{R}^{6\times6}$ is a transformation matrix defined in (3.29) (*cf.* Chapter 3). Differentiating (5.5) with respect to time leads to the following acceleration relationship

$$\ddot{\boldsymbol{\eta}} = B\dot{\boldsymbol{\nu}} + \dot{B}\boldsymbol{\nu} \tag{5.6}$$

At this point, consider a desired motion trajectory for the underwater vehicle represented in the Cartesian space. The task-space (i.e. the Cartesian space) velocity and the derivative of the generalised coordinates are related by the following equation

$$\dot{\boldsymbol{x}} = J\dot{\boldsymbol{\eta}} \tag{5.7}$$

where $\dot{\boldsymbol{x}} \in \mathbb{R}^m$ **is** the position and orientation vector in the task-space ($m \leq 6$) and $J \in \mathbb{R}^{m\times6}$ is the Jacobian matrix. In order to incorporate the desired trajectory into the dynamics of the system, (5.7) can be differentiated with respect to time to yield

$$\ddot{\boldsymbol{x}} = J\ddot{\boldsymbol{\eta}} + \dot{J}\dot{\boldsymbol{\eta}} \tag{5.8}$$

Substituting $\dot{\boldsymbol{\eta}}$ from (5.5) and $\ddot{\boldsymbol{\eta}}$ from (5.6) into (5.8), gives

$$\ddot{\boldsymbol{x}} = JB\dot{\boldsymbol{\nu}} + \left(J\dot{B} + \dot{J}B\right)\boldsymbol{\nu} \tag{5.9}$$

Therefore, taking into account (5.4), the relationship between the task-space acceleration and the generalised forces is given by

$$\ddot{\boldsymbol{x}} = JBM^{-1}\boldsymbol{\tau} + \left(J\dot{B} + \dot{J}B\right)\boldsymbol{\nu} - JBM^{-1}\left(C(\boldsymbol{\nu})\boldsymbol{\nu} + D(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}(\boldsymbol{\eta})\right) \tag{5.10}$$

where the inertia matrix M, the transformation matrix B and the Jacobian matrix J are, in general, full rank except for singular configurations (e.g. pitch singularity). Therefore, the non-square thruster control matrix $JBM^{-1} \in \mathbb{R}^{m\times n}$ from the first term in (5.10) is also a full rank matrix. Now, the model-based set-point tracking control law for an underwater vehicle in task-space can be formulated in the following form

$$\boldsymbol{\tau} = (JBM^{-1})^{\dagger}\Big[\Big(\ddot{\boldsymbol{x}}_d - \big(J\dot{B} + \dot{J}B\big)\boldsymbol{v} - JBM^{-1}\big(C(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta})\big)\Big) + K_v\dot{\tilde{\boldsymbol{e}}}_x + K_p\tilde{\boldsymbol{e}}_x\Big] \tag{5.11}$$

where $(JBM^{-1})^{\dagger} = (JBM^{-1})^{T}\big((JBM^{-1})(JBM^{-1})^{T}\big)^{-1}$ is the pseudo-inverse of $JBM^{-1} \in \mathbb{R}^{m\times n}$, K_v is the velocity gain matrix and K_p denotes the position gain matrix. $\dot{\tilde{\boldsymbol{e}}}_x = \dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}}$ and $\tilde{\boldsymbol{e}}_x = \boldsymbol{x}_d - \boldsymbol{x}$ represents the task-space velocity and position errors, respectively. Using the specific theorems in Appendix A (Lyapunov's direct method), it can be guaranteed that this control scheme achieves the globally uniformly asymptotically stable equilibrium point of the closed-loop equation. In fact, this is equivalent to global exponential stability of the origin if the closed-loop system is linear and autonomous (*cf.* Theorem A.5). Therefore, the set-point tracking control objective is fulfilled and the task can be illustrated as in Figure 5.1.

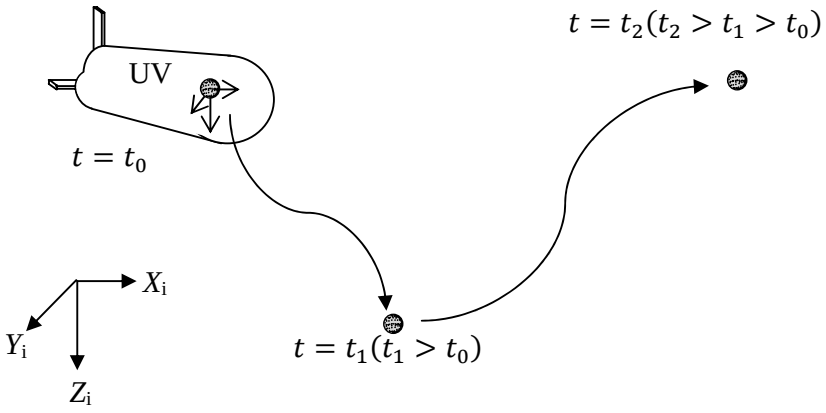


Figure 5.1: Set-point tracking task of an underwater vehicle

5.2.2 Region Tracking Control

The results in [5.1]-[5.5] focus on set-point control where the desired target is specified as a point. Alternatively, the desired target can also be defined as a region instead of a point. Recently, a region reaching control scheme was proposed for an underwater robot [5.1, 5.2]. Nevertheless, the region control laws in [5.1] are focused on reaching a stationary region. In many underwater tracking problems, the system is required to follow the moving target in a particular time, i.e. a pipeline maintenance task, rather than reaching a static target. To overcome this, a region tracking scheme was introduced for a fixed-base robot manipulator in [5.3], where an adaptive inertia-related approach is

utilised such that the desired target is specified as a moving target. Inspired from this previous work, this thesis presents sub-region tracking control approaches for underwater robots in the presence of dynamic uncertainties.

In order to demonstrate the theoretical framework of region tracking control, the same definition of region error is utilised as in the previous chapter

$$\tilde{\boldsymbol{e}}_{Rx} = \max\big(0, f(\delta\boldsymbol{x})\big)\left(\frac{\partial f(\delta\boldsymbol{x})}{\partial \boldsymbol{x}}\right)^{T} \tag{5.12}$$

where (5.12) implies for a single region function $f(\delta\boldsymbol{x}) \leq 0$; $\delta\boldsymbol{x} = \boldsymbol{x}_d - \boldsymbol{x} \in \mathbb{R}^3$.

Therefore, using the similar relationship between the task-space acceleration and the generalised forces as in (5.10), the model-based region tracking control law for an underwater vehicle given in (5.4) can be formulated in the following form

$$\begin{aligned}\boldsymbol{\tau} = (JBM^{-1})^{\dagger}\Big[\Big(\ddot{\boldsymbol{x}}_d - \big(J\dot{B} + \dot{J}B\big)\boldsymbol{v} - JBM^{-1}\big(C(\boldsymbol{v})\boldsymbol{v} + D(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta})\big)\Big) \\ + K_v\dot{\tilde{\boldsymbol{e}}}_x + K_p\tilde{\boldsymbol{e}}_{Rx}\Big]\end{aligned} \tag{5.13}$$

where $\tilde{\boldsymbol{e}}_x$ is substituted with $\tilde{\boldsymbol{e}}_{Rx}$. According to Lyapunov's direct method (*cf.* Appendix A) or in particular Theorem A.4, this control scheme guarantees the convergence of the region tracking error to zero. In view of Theorem A.5, this is equivalent to global exponential stability if the closed-loop system is linear and autonomous. The region tracking control objective is illustrated in Figure 5.2.

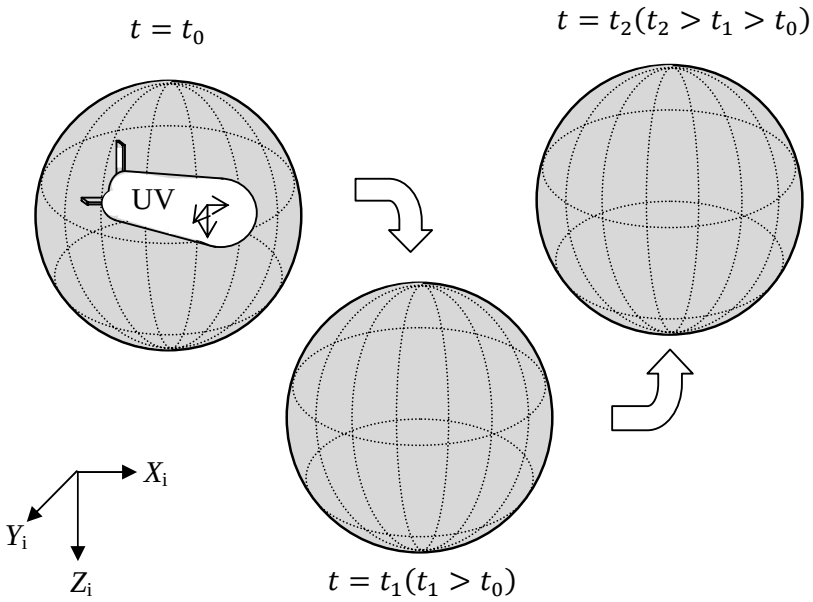


Figure 5.2: Region tracking task of an underwater vehicle

5.3. A Sub-Region and Sub-Task Tracking Control Scheme for a UVMS

As previously discussed, a region reaching control scheme was proposed for an underwater robot [5.1, 5.2], where the desired target is specified as a region instead of a point. In the UVMS region reaching control technique [5.1], there are two separate regions that were assigned for two different structures. The secondary region is specified for the macro system (underwater vehicle) while the primary region is specified for the mini system (onboard manipulator). However, it has been noted that for a particular case, it is necessary to keep the vehicle and onboard manipulator inside the moving sub-regions rather than specifying the desired target as static sub-regions.

Therefore, the design of a sub-region tracking controller for a kinematically redundant underwater robotic system using a generalised pseudo-inverse based formulation is considered in this section. As illustrated in Figure 5.3, two sub-regions are assigned for the vehicle and end-effector. The proposed controller is developed in order to achieve the sub-region and sub-task tracking despite the underwater parametric uncertainties. The proposed control strategy does not require the computation of the inverse kinematics and does not place any restrictions on the self-motion of the manipulator. Hence, the extra degrees-of-freedom are available for the sub-task. In addition, a non-

minimum four-parameter representation is used to resolve the singularities related to the three-parameter representation.

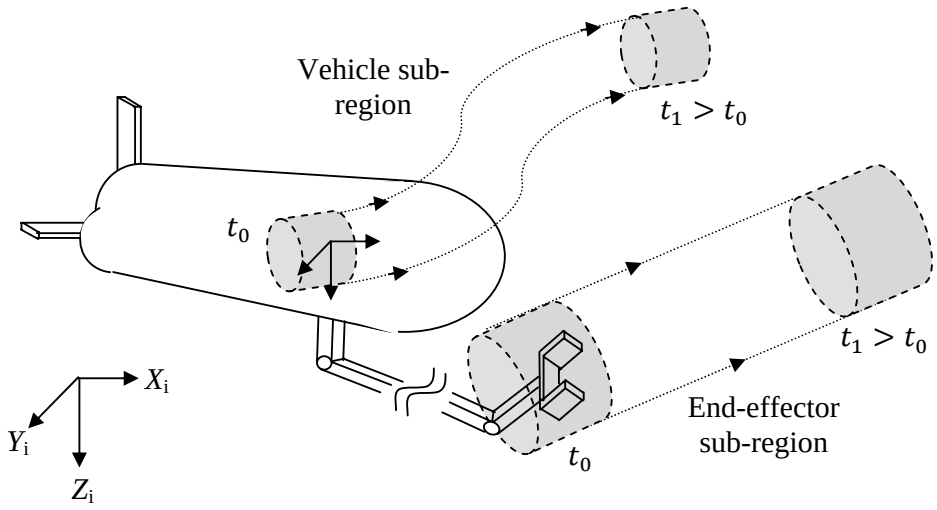


Figure 5.3: Sub-region tracking tasks of a UVMS

5.3.1 Control and Error System Formulation

The control objective is to design the control law such that the vehicle and end-effector in the task-space can follow their individual desired position region and orientation as closely as possible. Additionally, the designed control signal should also enable the redundancy of the manipulator to execute sub-tasks defined by a motion optimisation measure. Defining the vehicle region tracking error $\tilde{\boldsymbol{e}}_{pv}$ and end-effector region tracking error $\tilde{\boldsymbol{e}}_{pm}$ as follows

$$\tilde{\boldsymbol{e}}_{pv} = \begin{cases} 0, & f(\delta \boldsymbol{p}_v) \leq 0 \\ f(\delta \boldsymbol{p}_v)\left(\dfrac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T, & f(\delta \boldsymbol{p}_v) > 0 \end{cases}$$
$$\Leftrightarrow \tilde{\boldsymbol{e}}_{pv} \triangleq \max\left(0, f(\delta \boldsymbol{p}_v)\right)\left(\frac{\partial f(\delta \boldsymbol{p}_v)}{\partial \boldsymbol{p}_v}\right)^T \tag{5.14}$$

$$\tilde{\boldsymbol{e}}_{pm} = \begin{cases} 0, & f\left(\delta\, {}^v\boldsymbol{p}_{v,m}\right) \leq 0 \\ f\left(\delta\, {}^v\boldsymbol{p}_{v,m}\right)\left(\dfrac{\partial f\left(\delta\, {}^v\boldsymbol{p}_{v,m}\right)}{\partial\, {}^v\boldsymbol{p}_{v,m}}\right)^T, & f\left(\delta\, {}^v\boldsymbol{p}_{v,m}\right) > 0 \end{cases}$$
$$\Leftrightarrow \tilde{\boldsymbol{e}}_{pm} \triangleq \max\left(0, f\left(\delta\, {}^v\boldsymbol{p}_{v,m}\right)\right)\left(\frac{\partial f\left(\delta\, {}^v\boldsymbol{p}_{v,m}\right)}{\partial\, {}^v\boldsymbol{p}_{v,m}}\right)^T \tag{5.15}$$

where $\boldsymbol{p}_v$ and ${}^v\boldsymbol{p}_{v,m}$ are as defined previously. Similar to the region definition in [5.1, 5.3], the desired vehicle and end-effector region, represented by the scalar function $f(\delta\boldsymbol{p}_v) \in \mathbb{R}$ and $f\left(\delta\,{}^v\boldsymbol{p}_{v,m}\right) \in \mathbb{R}$, respectively, are defined as follows

$$f(\delta\boldsymbol{p}_v) \leq 0 \tag{5.16}$$

$$f\left(\delta\,{}^v\boldsymbol{p}_{v,m}\right) \leq 0 \tag{5.17}$$

where $\delta\boldsymbol{p}_v = \left(\boldsymbol{p}_{v,d} - \boldsymbol{p}_v\right) \in \mathbb{R}^3$ and $\delta\,{}^v\boldsymbol{p}_{v,m} = \left({}^v\boldsymbol{p}_{v,m,d} - {}^v\boldsymbol{p}_{v,m}\right) \in \mathbb{R}^3$ denote the continuous first partial derivatives. $\boldsymbol{p}_{v,d}(t) \in \mathbb{R}^3$ is the reference point inside the desired vehicle region and ${}^v\boldsymbol{p}_{v,m,d}(t) \in \mathbb{R}^3$ is the reference point inside the desired end-effector region. $\boldsymbol{p}_{v,d}$ and ${}^v\boldsymbol{p}_{v,m,d}$ can be obtained using a method similar to that in [5.4, 5.5] such that the end-effector tracking as the primary task is achieved. To ensure simultaneous region tracking of the vehicle and end-effector, the region functions are bounded by

$$f(\delta\boldsymbol{p}_v) \leq f\left(\delta\boldsymbol{p}_{v,m}\right) \tag{5.18}$$

where $\delta\boldsymbol{p}_{v,m}$ represents the continuous first partial derivatives of the end-effector expressed in the inertial-fixed frame. Now, defining the two vectors $\dot{\boldsymbol{p}}_{vr}$ and ${}^v\dot{\boldsymbol{p}}_{v,mr}$

$$\dot{\boldsymbol{p}}_{vr} = \dot{\boldsymbol{p}}_{v,d} + \lambda_{v1}\tilde{\boldsymbol{e}}_{pv} \tag{5.19}$$

$${}^v\dot{\boldsymbol{p}}_{v,mr} = {}^v\dot{\boldsymbol{p}}_{v,m,d} + \lambda_{m1}\tilde{\boldsymbol{e}}_{pm} \tag{5.20}$$

where $\tilde{\boldsymbol{e}}_{pv}$ and $\tilde{\boldsymbol{e}}_{pm}$ are defined in (5.14) and (5.15), respectively, λ_{v1} and λ_{m1} are diagonal and positive definite gain matrices. It is assumed that $\boldsymbol{p}_{v,d}(t)$, $\dot{\boldsymbol{p}}_{v,d}(t)$, $\ddot{\boldsymbol{p}}_{v,d}(t)$, ${}^v\boldsymbol{p}_{v,m,d}(t)$, ${}^v\dot{\boldsymbol{p}}_{v,m,d}(t)$, ${}^v\ddot{\boldsymbol{p}}_{v,m,d}(t)$ are all bounded functions of time. Note that, for a boundedness of $f(\delta\boldsymbol{p}_v)$, $\delta\boldsymbol{p}_v$ is bounded. Likewise, if $f\left(\delta\,{}^v\boldsymbol{p}_{v,m}\right)$ is bounded, then $\delta\,{}^v\boldsymbol{p}_{v,m}$ is also bounded.

As previously discussed in Chapter 3, the error between the actual and desired orientation for the vehicle and manipulator can be generalised as follows

$$\tilde{R} \triangleq RR_d^T = \left(\tilde{e}_0{}^2 - \tilde{\boldsymbol{e}}_\varepsilon{}^T\tilde{\boldsymbol{e}}_\varepsilon\right)I_3 + 2\tilde{\boldsymbol{e}}_\varepsilon\tilde{\boldsymbol{e}}_\varepsilon{}^T - 2\tilde{e}_0\tilde{\boldsymbol{e}}_\varepsilon{}^\times \tag{5.21}$$

where R is the rotational matrix and R_d is the rotational matrix of R expressing the desired orientation which also is described by the quaternion $\boldsymbol{\epsilon} \triangleq \left[\epsilon_{0,d} \quad \boldsymbol{\epsilon}_{\varepsilon,d}^T\right]^T$. The corresponding unit quaternion representation is denoted by $\tilde{\boldsymbol{e}}(t) \triangleq [\tilde{e}_0(t) \quad \tilde{\boldsymbol{e}}_\varepsilon^T(t)]^T$. Thus, the quaternion propagation equation can be considered as

$$\begin{aligned} \dot{\tilde{e}}_0 &= -\frac{1}{2}\tilde{\boldsymbol{e}}_\varepsilon^T\tilde{\boldsymbol{\omega}} \\ \dot{\tilde{\boldsymbol{e}}}_\varepsilon &= \frac{1}{2}(\tilde{e}_0 I_3 + \tilde{\boldsymbol{e}}_\varepsilon{}^\times)\tilde{\boldsymbol{\omega}} \end{aligned} \tag{5.22}$$

where $\tilde{\boldsymbol{\omega}}(t) = \boldsymbol{\omega}_d(t) - \boldsymbol{\omega}(t)$; $\boldsymbol{\omega}(t)$ is defined in Chapter 3 for both the vehicle and manipulator. $\boldsymbol{\omega}_d(t)$ is the desired angular velocity of the vehicle and manipulator. Note that the determinant of $(\tilde{e}_0 I_3 + \tilde{\boldsymbol{e}}_\varepsilon{}^\times)$ in (5.22) is $\tilde{e}_0(t)$ and it satisfies the following remark:

Remark 5.1: The second equation in (5.22) is invertible provided $\tilde{e}_0(t) \neq 0$ for any time. To ensure that $\tilde{e}_0(t) \neq 0$ for all time, the desired trajectory must be initialised to guaranteed that $\tilde{e}_0(t) \neq 0$, and the subsequent control design must ensure that $\tilde{e}_0(t) \neq 0$ after the initial time.

The general filtered tracking error vector is defined as $\boldsymbol{r}(t) = [\boldsymbol{r}_v(t) \quad \boldsymbol{r}_m(t)]^T \in \mathbb{R}^{(6+n)}$; $\boldsymbol{r}_v(t) = [\boldsymbol{r}_{v_v} \quad \boldsymbol{r}_{\omega_v}]^T \in \mathbb{R}^6$ and $\boldsymbol{r}_m(t) \in \mathbb{R}^n$. Based on the structure of (5.19) and (5.22) and the subsequent stability analysis, an auxiliary signal for the vehicle is defined as follows

$$\boldsymbol{r}_v \triangleq \begin{bmatrix} E_1^T\dot{\boldsymbol{p}}_{v,d} + E_1^T\lambda_{v1}\,\tilde{\boldsymbol{e}}_{pv} \\ \boldsymbol{\omega}_{v,d} + \lambda_{v2}\tilde{\boldsymbol{e}}_{\varepsilon v} \end{bmatrix} - \boldsymbol{v} = \begin{bmatrix} \widetilde{\boldsymbol{v}}_1 \\ \widetilde{\boldsymbol{v}}_2 \end{bmatrix} + \begin{bmatrix} E_1^T\lambda_{v1}\,\tilde{\boldsymbol{e}}_{pv} \\ \lambda_{v2}\tilde{\boldsymbol{e}}_{\varepsilon v} \end{bmatrix} \tag{5.23}$$

where λ_{v2} is a diagonal gain matrix. On the other hand, the available redundancy of the manipulator can be exploited to introduce additional constraints to be satisfied along

with the manipulator motion coordination. Thus, after taking account of (5.20) and the structure of (5.22), the filtered tracking error for the redundant manipulator can be obtained as

$$\begin{aligned}\boldsymbol{r}_m &\triangleq J_M^{\dagger}\begin{bmatrix} {}^{v}\dot{\boldsymbol{p}}_{v,m,d} + \lambda_{m1}\tilde{\boldsymbol{e}}_{pm} \\ {}^{v}\boldsymbol{\omega}_{v,m,d} + \lambda_{m2}\tilde{\boldsymbol{e}}_{\varepsilon m}\end{bmatrix} + \left(I_n - J_M^{\dagger}J_M\right)\boldsymbol{z} - \dot{\boldsymbol{q}} \\ &= \dot{\tilde{\boldsymbol{q}}} + J_M^{\dagger}\begin{bmatrix}\lambda_{m1}\tilde{\boldsymbol{e}}_{pm} \\ \lambda_{m2}\tilde{\boldsymbol{e}}_{\varepsilon m}\end{bmatrix} + \left(I_n - J_M^{\dagger}J_M\right)\boldsymbol{z}\end{aligned} \tag{5.24}$$

with λ_{m2} being a diagonal and positive definite gain matrix and $\dot{\tilde{\boldsymbol{q}}}$ denotes the joint velocity error. The vector $\boldsymbol{z}$ is defined in Chapter 3 and I_n is the $n \times n$ identity matrix. Define the sub-task tracking error as follows [5.6]

$$\tilde{\boldsymbol{e}}_{sub} = \left(I_n - J_M^{\dagger}J_M\right)(\boldsymbol{z} - \dot{\boldsymbol{q}}) \tag{5.25}$$

The properties of null space of the pseudo-inverse can be used to show that the sub-task tracking error defined in (5.25) is also regulated when $\boldsymbol{r}_m(t)$ is regulated to obtain

$$\tilde{\boldsymbol{e}}_{sub} = \left(I_n - J_M^{\dagger}J_M\right)\boldsymbol{r}_m \tag{5.26}$$

where (5.24) is pre-multiplied by $\left(I_n - J_M^{\dagger}J_M\right)$. Therefore, the sub-task control is also achieved.

In general, the development of the open-loop error system for $\boldsymbol{r}(t)$ can be obtained by pre-multiplying the inertia matrix with the time derivative of $\boldsymbol{r}(t)$ to yield

$$M\dot{\boldsymbol{r}} = Y\left(\boldsymbol{q}, \boldsymbol{\eta}_v, \boldsymbol{\xi}, \boldsymbol{\xi}_r, \dot{\boldsymbol{\xi}}_r, \boldsymbol{z}, \dot{\boldsymbol{z}}\right)\boldsymbol{\Phi} - \boldsymbol{\tau} - C\boldsymbol{r} - D\boldsymbol{r} \tag{5.27}$$

where $Y\left(\boldsymbol{q}, \boldsymbol{\eta}_v, \boldsymbol{\xi}, \boldsymbol{\xi}_r, \dot{\boldsymbol{\xi}}_r, \boldsymbol{z}, \dot{\boldsymbol{z}}\right) \in \mathbb{R}^{(6+n)\times n_p}$ denotes a measurable regression matrix and $\boldsymbol{\Phi}$ is a set of UVMS dynamic parameters; subscript $\boldsymbol{r}$ represents the components of $\boldsymbol{\xi}$.

Based on the error system development and the subsequent stability analysis, the proposed control law is

$$\boldsymbol{\tau} = K_0 K_p \begin{bmatrix} \tilde{\boldsymbol{e}}_{pv} \\ \tilde{\boldsymbol{e}}_{\varepsilon v} \\ \tilde{\boldsymbol{e}}_{pm} \\ \tilde{\boldsymbol{e}}_{\varepsilon m} \end{bmatrix} + K_r \boldsymbol{r} + Y \hat{\boldsymbol{\Phi}} \tag{5.28}$$

with

$$K_0 = diag\{E_1^T, I_{3\times 3}, J_M^T\} \in \mathbb{R}^{(6+n)\times 12}, \text{ and}$$
$$K_p = diag\{k_{v1}, k_{v2} I_{3\times 3}, k_{m1}, k_{m2} I_{3\times 3}\} \in \mathbb{R}^{12\times 12}$$

where k_{v1}, $k_{m1} \in \mathbb{R}^{3\times 3}$ are symmetric positive definite matrices, $I_{3\times 3}$ is a 3×3 identity matrix, k_{v2} and k_{m2} are positive scalars, $K_r \in \mathbb{R}^{(6+n)\times(6+n)}$ is a positive definite and diagonal gain matrix. $\hat{\boldsymbol{\Phi}}$ represents the parameter estimate vector which is updated according to

$$\dot{\hat{\boldsymbol{\Phi}}} = \Gamma^{-1} Y^T(\cdot) \boldsymbol{r} \tag{5.29}$$

where Γ^{-1} is a constant, positive definite, diagonal gain matrix. Substituting (5.28) into (5.27) produces the closed-loop dynamics for $\boldsymbol{r}(t)$ as follows

$$M\dot{\boldsymbol{r}} = Y(\cdot)\tilde{\Phi} - K_r \boldsymbol{r} - K_0 K_p \begin{bmatrix} \tilde{\boldsymbol{e}}_{pv} \\ \tilde{\boldsymbol{e}}_{\varepsilon v} \\ \tilde{\boldsymbol{e}}_{pm} \\ \tilde{\boldsymbol{e}}_{\varepsilon m} \end{bmatrix} - C\boldsymbol{r} - D\boldsymbol{r} \tag{5.30}$$

where $\tilde{\boldsymbol{\Phi}}(t) = \boldsymbol{\Phi} - \hat{\boldsymbol{\Phi}}(t)$ denotes the parameter estimation error. The stability of the sub-region, orientation and sub-task tracking control is stated by the following theorem:

Theorem 5.1: The control law described in (5.28) and (5.29) guarantees asymptotic sub-region, orientation and sub-task tracking for the kinematically redundant robot manipulator mounted on the underwater vehicle given by (3.63) in the sense that all the tracking errors go to zero as $t \to \infty$.

Proof: See Appendix B.

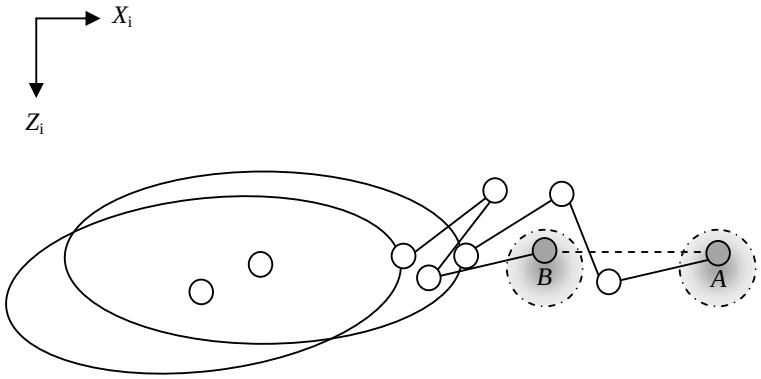
5.3.2 A Sub-Task Objective

In this section, a sub-task objective based on singularity avoidance for the kinematically redundant onboard manipulator is chosen, in addition to the main sub-region tracking objective. The manipulability measure is defined by [5.7]

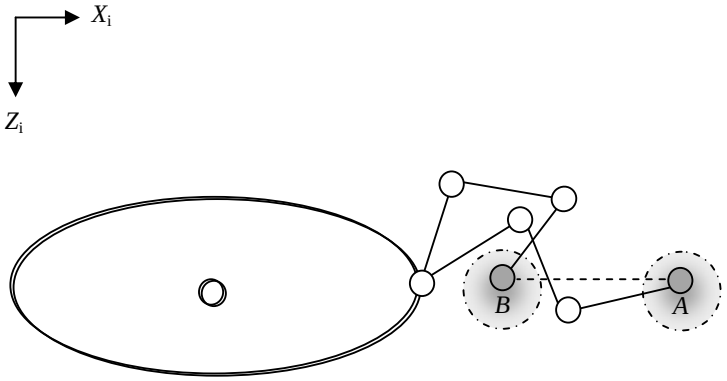
$$\mathrm{H}(\boldsymbol{q}) = \sqrt{\det(JJ^T)} \tag{5.31}$$

where $\det(\cdot)$ represents the determinant of the matrix and J is the Jacobian matrix. If this measure is maximised, then redundancy of the system is exploited so as to move away from singularities. Note that, the vehicle is required to move only when the manipulator itself is in undesired situations. Alternatively, it is preferable to have a manipulator that can reconfigure by itself without vehicle movement which also results in a safe configuration.

Figure 5.4 illustrates a planar framed view of the UVMS's movement where the manipulator is in a singular configuration ($q_2 \approx 180°$) provided that no particular joint limit is enforced. When the trajectory becomes very close to the vehicle body (i.e. from region A to region B), the vehicle has to contribute to the end-effector motion as shown in Figure 5.4(a). Alternatively, the redundant joint of the manipulator has to be reconfigured in a dexterous posture as in Figure 5.4(b) in order to avoid a singular configuration that would occur if the second joint is retracted. A similar situation could be presented for the outstretched case ($q_2 \approx 0°$).



(a)



(b)

Figure 5.4: Planar view of the UVMS with its onboard manipulator in singularity free configuration mode

5.3.3 Simulation Results

The simulation studies are performed on a 6-DOFs underwater vehicle equipped with a 3-link arm with revolute joints. The dynamic model of the ellipsoid-shape vehicle is characterised by a parametric representation as in [5.8] (*cf.* Appendix D). For simplicity and presentational effectiveness, planar motion is considered for the manipulator working in the vertical plane; hence the number of redundant DOFs of the UVMS is one. The masses and lengths of the manipulator links are set as $m_{L_i} = 3.38$ kg and $L_i = 0.33$ m where $i = 1$ to 3. The links are cylindrical and the radii of each link is 0.05 m.

In the simulations, arbitrary constant values are used for unknown parameters such as hydrodynamic damping matrices and added masses. The values for added mass are

approximated to 50% of the link's rigid inertia and the hydrodynamic damping is assumed to be the summation of linear and quadratic damping whose effects are non-coupled with respect to other velocities. The regression matrix $Y(\boldsymbol{q}, \boldsymbol{\eta}_e, \boldsymbol{\xi}, \boldsymbol{\xi}_r, \dot{\boldsymbol{\xi}}_r, \boldsymbol{z}, \dot{\boldsymbol{z}}) \in \mathbb{R}^{9\times 29}$ is defined in the following form:

$$Y(\cdot) = \begin{bmatrix} Y_v(\boldsymbol{q}, \boldsymbol{\eta}_e, \boldsymbol{\xi}, \boldsymbol{\xi}_r, \dot{\boldsymbol{\xi}}_r)_{6\times 19} & 0_{6\times 10} \\ 0_{3\times 19} & Y_m(\boldsymbol{q}, \boldsymbol{\eta}_e, \boldsymbol{\xi}, \boldsymbol{\xi}_{r,m}, \dot{\boldsymbol{\xi}}_{r,m}, \boldsymbol{z}, \dot{\boldsymbol{z}})_{3\times 10} \end{bmatrix} \tag{5.32}$$

where subscripts v, m and r represent the vehicle, manipulator and components of $\boldsymbol{\xi}$, respectively. The constant parameter vector is constructed as follows: $\boldsymbol{\Phi} = [\boldsymbol{\Phi}_v^{\mathrm{T}} \quad \boldsymbol{\Phi}_m^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{29}$.

To verify the effectiveness of the controller proposed in (5.28) and (5.29), the simulation is performed by specifying a desired moving region for each sub-system using (5.16) and (5.17). The control objective for the vehicle is to track a desired spherical sub-region with a radius of γ_v. The centre of the desired sub-region is moving in a straight line trajectory in operational space. The end-effector is required to execute a circular sub-region with a radius of γ_m. The centre of the desired sub-region is moving in a line expressed in the body-fixed frame. The desired spherical and circular sub-regions are specified as

$$f(\delta \boldsymbol{p}_v) = (x_{vd} - x_v)^2 + (y_{vd} - y_v)^2 + (z_{vd} - z_v)^2 - \gamma_v^2 \leq 0 \tag{5.33}$$

$$f\left(\delta\, {}^{v}\boldsymbol{p}_{v,m}\right) = (x_{md} - x_m)^2 + (z_{md} - z_m)^2 - \gamma_m^2 \leq 0 \tag{5.34}$$

The system was initialised to be at rest at the following pose:

$$\boldsymbol{p}_v(0) = [2.2 \quad 0 \quad -1.6]^T \text{m},$$
$$\boldsymbol{e}_v(0) = [0 \quad 0 \quad 0 \quad 1]^T \text{ and}$$
$$\boldsymbol{q}_i(0) = [\pi/2.75 \quad -\pi/1.75 \quad \pi/18]^T \text{rad}.$$

For comparison purposes, $H(\boldsymbol{q})$ was initially set to zero when there was no restriction on the self-motion so that only the sub-region tracking objective is enforced. Then, a single sub-task $H(\boldsymbol{q})$ was selected to maximise the manipulability as follows

$$H(\boldsymbol{q}) = \det(J_M J_M^T) \tag{5.35}$$

where $\det(\cdot)$ represents the determinant of the matrix. Note that there are no mechanical joint limits assigned in these simulations. Given that $t_f = 50$ s, the spherical sub-region for the vehicle in Simulation 1 is kept stationary at

$$\begin{bmatrix} x_{vd} \\ y_{vd} \\ z_{vd} \end{bmatrix} = \begin{bmatrix} 2.2 \\ 0 \\ -1.6 \end{bmatrix} \text{[m]} \tag{5.36}$$

The trajectory of the centre of the circular sub-region for the manipulator in Simulation 1 with respect to the body-fixed frame is defined as follows

$$\begin{bmatrix} x_{md} \\ z_{md} \end{bmatrix} = \begin{bmatrix} 0.69 + 0.01\left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \\ -0.05 + 0.18\left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \end{bmatrix} \text{[m]} \tag{5.37}$$

Meanwhile, the trajectory of the centre of the spherical sub-region for the vehicle in Simulation 2 is specified as

$$\begin{bmatrix} x_{vd} \\ y_{vd} \\ z_{vd} \end{bmatrix} = \begin{bmatrix} 2.2 - 0.22\left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \\ 0 \\ -1.6 + 0.01\left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \end{bmatrix} \text{[m]} \tag{5.38}$$

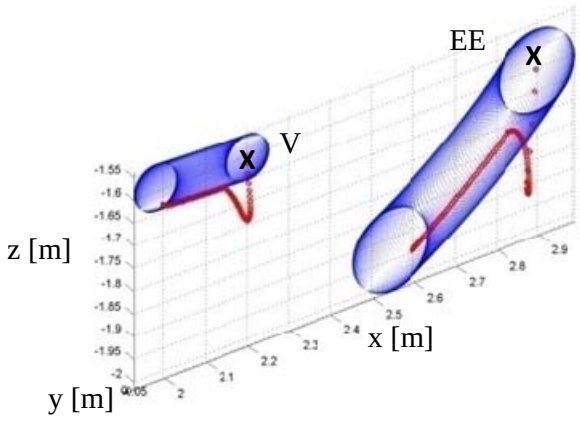
The trajectory of the centre of the circular sub-region for the manipulator in Simulation 2 is similar to that of (5.37). The sub-regions for both simulations are set to $\gamma_v = 0.05$ m and $\gamma_m = 0.09$ m. For the best performance, the controller gains are tuned to $K_p = diag\{208, 208, 208, 33, 99, 99, 42.5, 42.5, 42.5\}$ and $K_r = diag\{100, 100, 100, 30, 90, 90, 85, 85, 85\}$.

Figure 5.5 shows the desired sub-regions and actual paths for the vehicle and end-effector in 3-D operational space. Note that the initial position, vehicle and end-effector are marked with "**X**","V" and "EE", respectively. Initially, the joint errors are non-zero due to mismatch in the restoring torques compensation. It converges to zero at the final state configuration regardless of the sub-task.

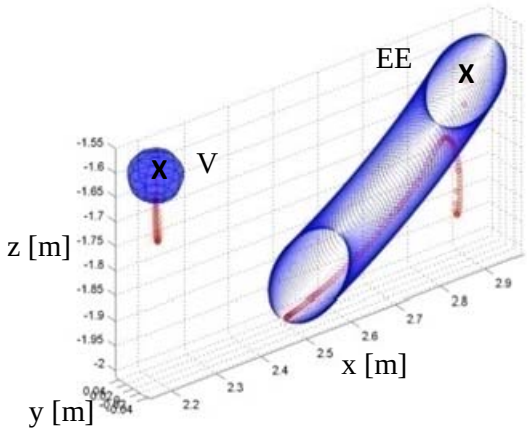
The importance and the effect of the sub-task on the UVMS can be seen in Figure 5.6 and from the manipulability index given in Figure 5.7. In Figure 5.6, the initial pose of the UVMS is drawn by blue lines and the final pose by red lines. The black lines represent traces of the actual tracking trajectory for the end-effector and vehicle while magenta lines are those of the reference line inside the sub-regions. The yellow lines are intermediate poses of the UVMS.

As observed in Figure 5.6(a), both subsystems track their individual region. However, due to the uncontrolled self-motion of the system, the manipulator is retracted when the end-effector trajectory requires the displacement to be very close to the vehicle body. In this case it is clear that arm singularities occur when $q_2 \cong -\pi$ rad, which is undesirable and the arm should not even come close to such a configuration. Therefore, the vehicle needs to change its position and orientation in order to achieve the tracking objective. However, more energy is needed to move the vehicle. To overcome this drawback, the sub-task of keeping the manipulator in dexterous configurations is considered.

Using the manipulability measure (5.35), the singular configuration is avoided; hence the end-effector can move dexterously in its sub-region even if the vehicle and the end-effector sub-regions are very close to each other as shown in Figure 5.6(b). In addition, only the manipulator is needed to complete the tracking task while keeping the underwater vehicle in a stationary mode. Therefore, the energy usage for the UVMS system is reduced significantly compared to the one without the sub-task as illustrated in Figures 5.8, 5.9 and 5.10.

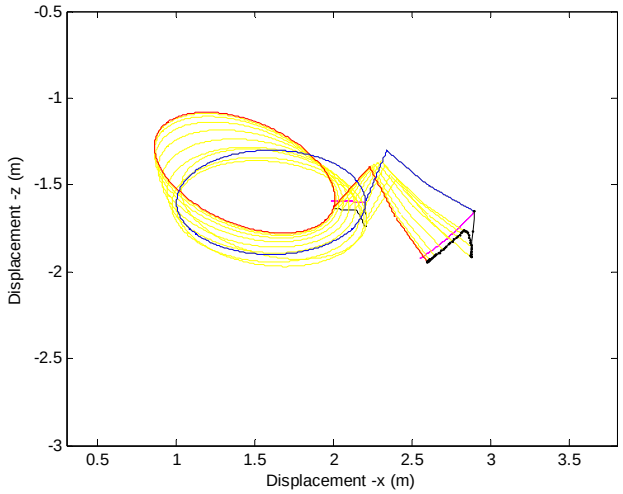


(a) Simulation 1

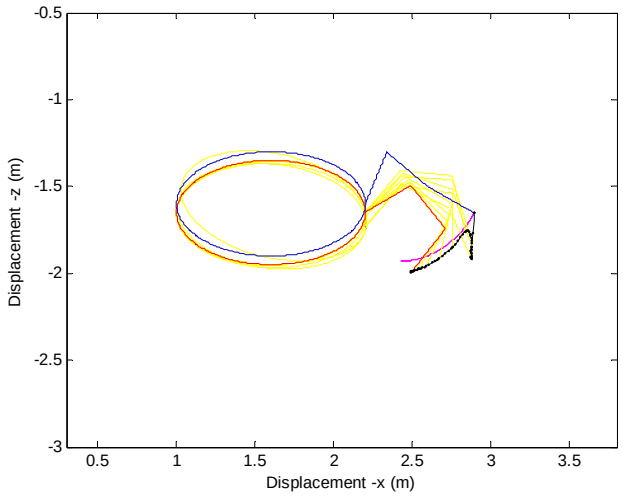


(b) Simulation 2

Figure 5.5: Desired sub-region and actual tracking trajectories for both simulations



(a) Simulation 1



(b) Simulation 2

Figure 5.6: Comparison of two UVMS configurations in a planar plane: (a) No sub-task tracking, (b) Manipulability measure to avoid a kinematic singularity ($q_2 \approx -\pi$ rad)

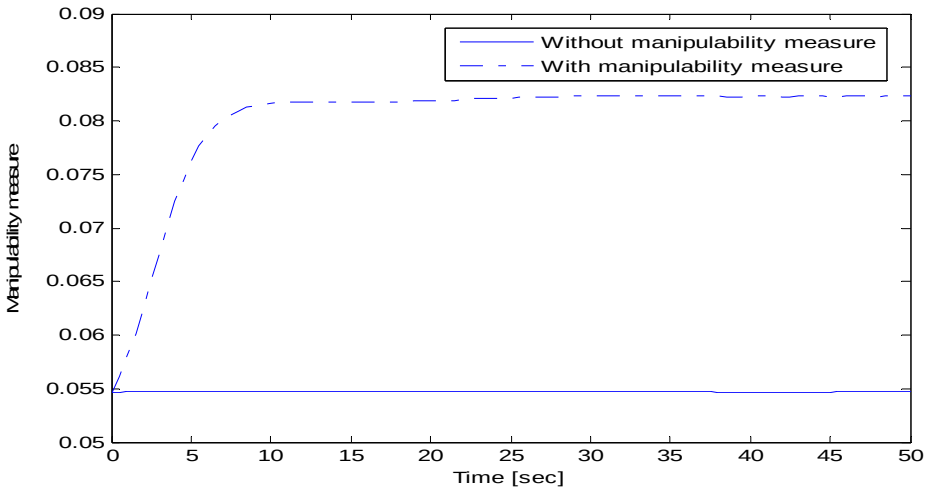


Figure 5.7: The changes of the manipulability measure for each simulation

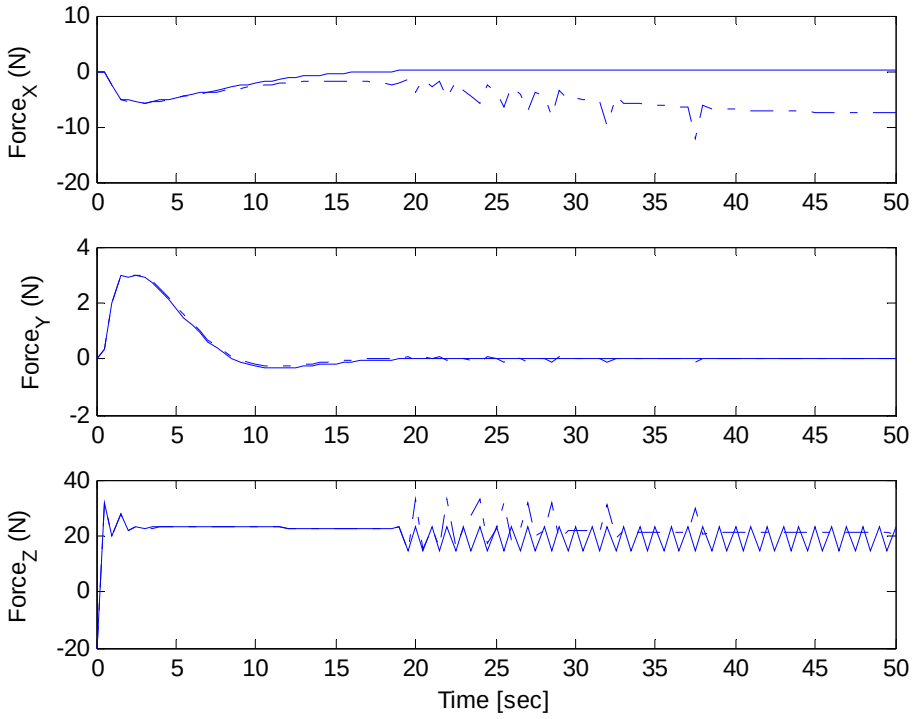


Figure 5.8: Required forces in Simulation 1 (dash-dot lines) and Simulation 2 (solid lines)

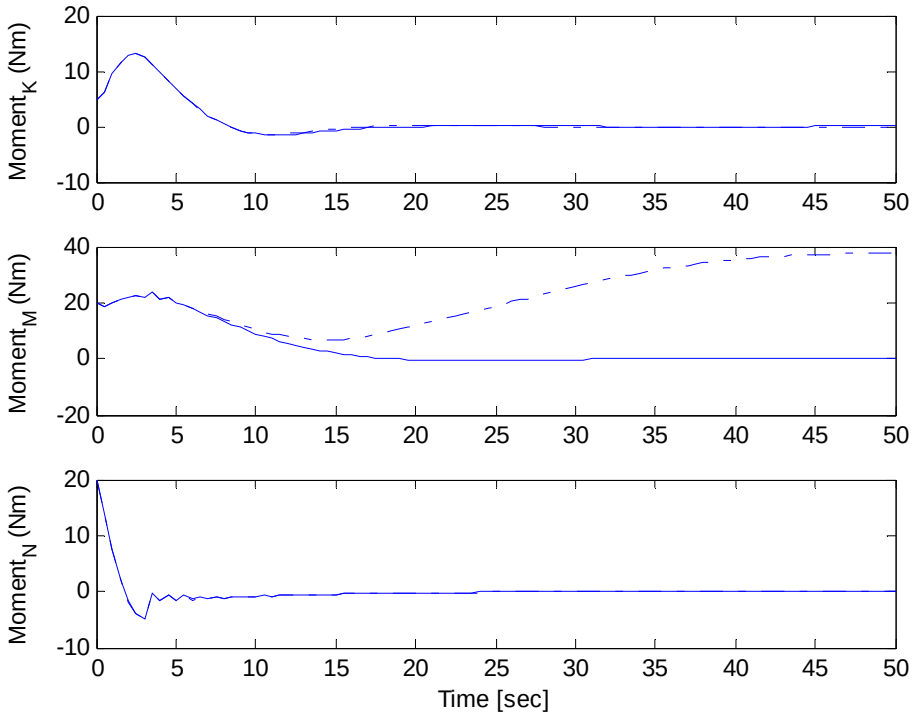


Figure 5.9: Required moments in Simulation 1 (dash-dot lines) and Simulation 2 (solid lines)

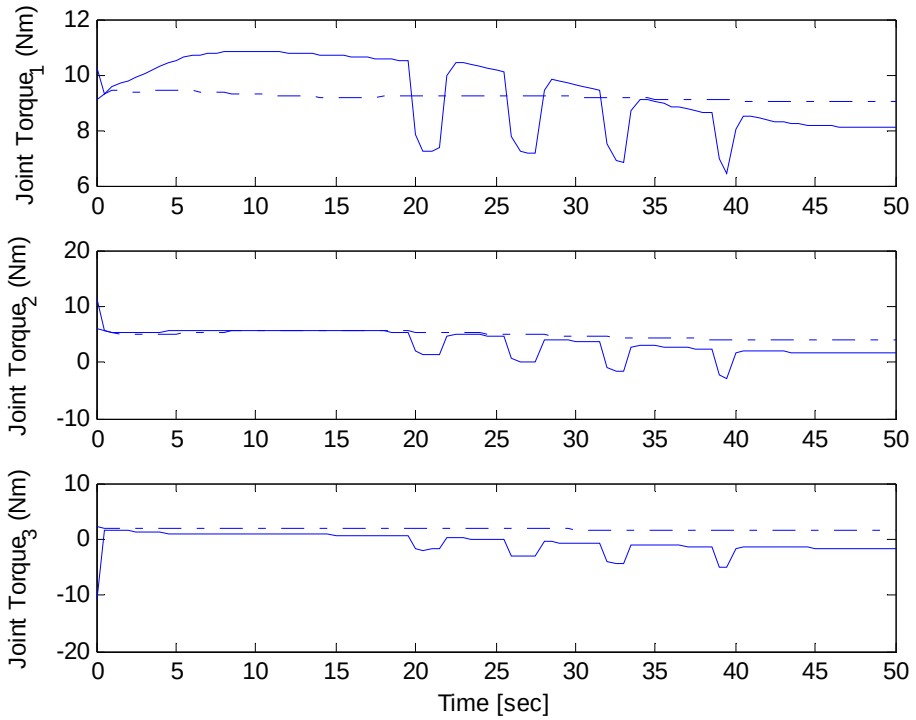
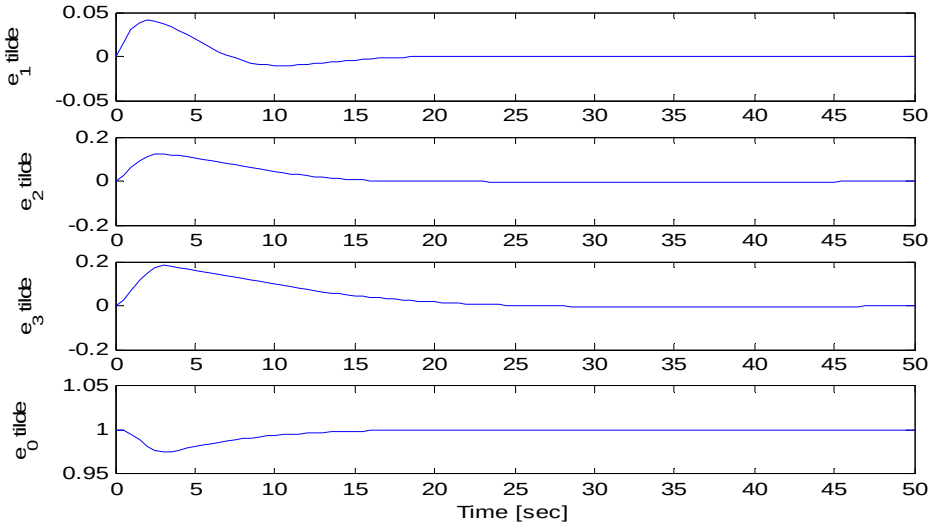
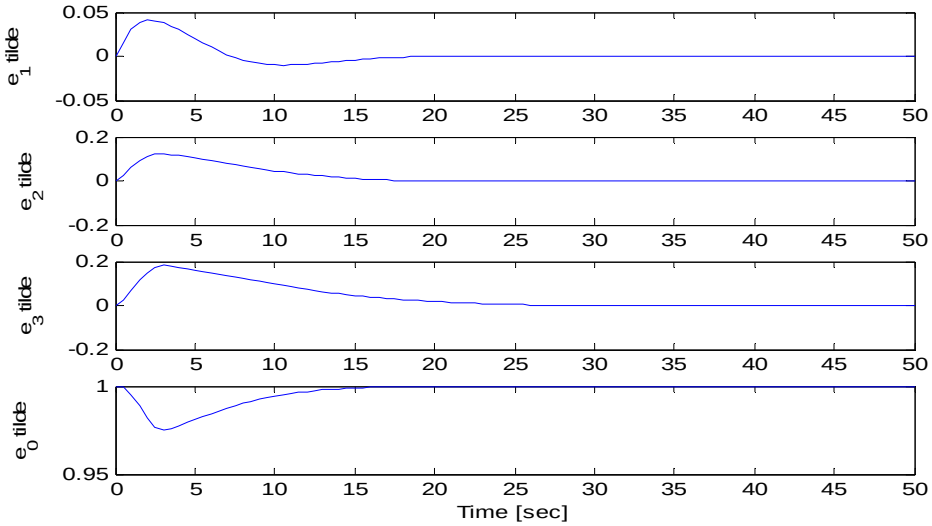


Figure 5.10: Required torques in Simulation 1 (dash-dot lines) and Simulation 2 (solid lines)

Figure 5.11 shows the quaternion based vehicle orientation tracking error with no sub-task in Figure 5.11(a) and with the manipulability measure in Figure 5.11(b). The quaternion tracking errors are asymptotically stable with the real parts of error $\tilde{e}_{ov}$ tending to unity and the imaginary parts of $\tilde{e}_{\varepsilon v}$ tending to zero in parallel with the controller properties, even for different forms of vehicle orientation.



(a) Simulation 1



(b) Simulation 2

Figure 5.11: Quaternion based vehicle orientation tracking errors for both simulations

5.4. Adaptive Robust Tracking Control for a UVMS

Based on a generalised pseudo-inverse formulation, the control input presented previously allows the tracking of a single sub-region and sub-task objective. However, in some UVMS applications, it involves the use of multiple sub-regions and sub-task criteria. For example, tracking a pipeline at a very specific region as illustrated in Figure 5.12 where the desired path is composed of an intersection of multiple sub-regions. The local vehicle or end-effector sub-regions consist of at least two local sub-regions, namely an inner sub-region and outer sub-region. These sub-regions intersect one another to yield an absolute sub-region for the tracking control. This approach is beneficial for the system when trying to avoid any obstacles in its tracking region. The local sub-region intersection can be exploited in various ways; spherical intersection or cubic intersection as discussed in [5.1, 5.9]. Moreover, the availability of the system's redundancy can be used to perform multiple sub-task criteria (i.e. drag minimisation and manipulability) instead of a single sub-task. The formulation of multiple sub-task criteria is briefly explained in the subsequent sections.

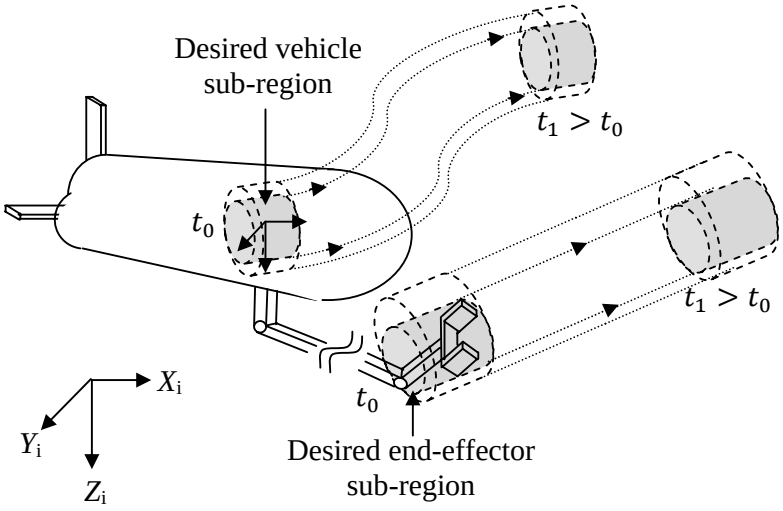


Figure 5.12: Intersection of multiple local sub-region tracking objectives of a UVMS

In an underwater environment, the tracking control task is challenging due to parametric uncertainties and an additive bounded disturbance. The presence of variable ocean currents creates hydrodynamic forces and moments that are not well-known or predictable, even though they are bounded. Therefore, an efficient controller for accurate prescribed trajectory tracking should incorporate the robot dynamics whilst being robust to parametric uncertainties associated with the dynamics and external

disturbances. In this section, the previously presented tracking controller is extended to an adaptive robust controller and also enables the use of multiple objectives. When an adaptive and robust control algorithm is employed, the adaptive approach satisfies the learning ability, while the employed robust controller is capable of rejecting bounded disturbances and increases robustness to the uncertainties.

5.4.1 Formulation of Multiple Sub-Regions Objectives

The control objective is to design the input command signal such that the vehicle and the end-effector can track their desired local sub-regions and orientation references as closely as possible. Note that, the desired local sub-region is obtained from the intersection of multiple local sub-regions. To find this, the vehicle region tracking error $\tilde{\boldsymbol{\varsigma}}_{pv}$ and end-effector region tracking error $\tilde{\boldsymbol{\varsigma}}_{pm}$ are defined as follows [5.3]

$$\tilde{\boldsymbol{\varsigma}}_{pv} \triangleq \sum_{j=1}^{N_1} \frac{\beta_{vj}}{k_{v1}} \max\left(0, f_{vj}\left(\delta \boldsymbol{p}_{vj}\right)\right) \left(\frac{\partial f_{vj}\left(\delta \boldsymbol{p}_{vj}\right)}{\partial\left(\delta \boldsymbol{p}_{vj}\right)}\right)^T \tag{5.39}$$

$$\tilde{\boldsymbol{\varsigma}}_{pm} \triangleq \sum_{i=1}^{N_2} \frac{\beta_{mi}}{k_{m1}} \max\left(0, f_{mi}\left(\delta\, {}^{v}\boldsymbol{p}_{v,mi}\right)\right) \left(\frac{\partial f_{mi}\left(\delta\, {}^{v}\boldsymbol{p}_{v,mi}\right)}{\delta\left(\delta\, {}^{v}\boldsymbol{p}_{v,mi}\right)}\right)^T \tag{5.40}$$

where β_{vj} and β_{mi} are positive constants. The subscripts v and m denote the vehicle and manipulator end-effector, respectively. After defining the proper intersection of multiple local sub-regions, the desired vehicle and end-effector sub-regions, represented by the scalar functions $f_{vj}\left(\delta \boldsymbol{p}_{vj}\right) \in \mathbb{R}$ and $f_{mi}\left(\delta\, {}^{v}\boldsymbol{p}_{v,mi}\right) \in \mathbb{R}$, respectively, can be specified as follows [5.3, 5.9]

$$f_{vj}\left(\delta \boldsymbol{p}_{vj}\right) = \begin{bmatrix} f_1(\delta \boldsymbol{p}_{v1}) \\ f_2(\delta \boldsymbol{p}_{v2}) \\ \vdots \\ f_{N_1}\left(\delta \boldsymbol{p}_{vN_1}\right) \end{bmatrix} \leq 0 \Leftrightarrow f_v(\delta \boldsymbol{p}_v) \leq 0 \tag{5.41}$$

$$f_{mi}\left(\delta\, {}^{v}\boldsymbol{p}_{v,mi}\right) = \begin{bmatrix} f_1\left(\delta\, {}^{v}\boldsymbol{p}_{v,m1}\right) \\ f_2\left(\delta\, {}^{v}\boldsymbol{p}_{v,m2}\right) \\ \vdots \\ f_{N_2}\left(\delta\, {}^{v}\boldsymbol{p}_{v,mN_2}\right) \end{bmatrix} \leq 0 \Leftrightarrow f_m\left(\delta\, {}^{v}\boldsymbol{p}_{v,m}\right) \leq 0 \tag{5.42}$$

where $\delta \boldsymbol{p}_{vj} = \left(\boldsymbol{p}_{v,dj} - \boldsymbol{p}_{v}\right) \in \mathbb{R}^3$ and $\delta\, {}^{v}\boldsymbol{p}_{v,mi} = \left({}^{v}\boldsymbol{p}_{v,m,di} - {}^{v}\boldsymbol{p}_{v,m}\right) \in \mathbb{R}^3$ denote the continuous first partial derivatives. $\boldsymbol{p}_{v,di}(t) \in \mathbb{R}^3$ is the reference point inside the j^{th} desired vehicle sub-region, $j = 1,2 \dots, N_1$ and ${}^{v}\boldsymbol{p}_{v,m,di}(t) \in \mathbb{R}^3$ is the reference point inside the i^{th} desired end-effector sub-region, $i = 1,2 \dots, N_2$; N_1 and N_2 are the maximum number of vehicle and end-effector local sub-regions respectively. $\boldsymbol{p}_{v,dj}(t)$ and ${}^{v}\boldsymbol{p}_{v,m,di}(t)$ can be obtained using a similar method to [5.4] such that end-effector tracking as the primary task is achieved. Note that, (5.41) and (5.42) imply the same boundedness of (5.18). Based on Remark 5.1 and the equivalent definition of (5.19), (5.20) and (5.22), a similar design procedure from the previous section can be followed to define the general filtered tracking error for the construction of an adaptive robust control law. In this section, the sub-task tracking error, the generalised orientation and filtered tracking errors are represented by $\tilde{\boldsymbol{\varsigma}}_{sub}$(t), $\tilde{\boldsymbol{\varsigma}} \triangleq [\tilde{\boldsymbol{\varsigma}}_0 \quad \tilde{\boldsymbol{\varsigma}}_{\varepsilon}^T]^T$ and $\bar{\boldsymbol{r}}(t) = [\bar{\boldsymbol{r}}_v^T(t) \quad \bar{\boldsymbol{r}}_m^T(t)]^T$, respectively.

Now, let the dynamic model of a UVMS given by (3.63) and (3.64) contain unknown bounded disturbances, $\boldsymbol{T}_d \in \mathbb{R}^{(6+n)}$, such that

$$M(\boldsymbol{q})\dot{\xi} + C(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + D(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + \boldsymbol{g}(\boldsymbol{\eta}_e,\boldsymbol{q}) + \boldsymbol{T}_d = \boldsymbol{\tau} \tag{5.43}$$

and

$$M(\boldsymbol{q})\dot{\xi} + C(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + D(\boldsymbol{q},\boldsymbol{\xi})\boldsymbol{\xi} + \boldsymbol{g}(\boldsymbol{\eta}_e,\boldsymbol{q}) + \boldsymbol{T}_d = Y\left(\boldsymbol{q},\boldsymbol{\eta}_e,\boldsymbol{\xi},\dot{\boldsymbol{\xi}}\right)\boldsymbol{\Phi} \tag{5.44}$$

Therefore, it is clear that the adaptive-robust sub-region control law in the form of

$$\boldsymbol{\tau} = K_0 K_p \tilde{\boldsymbol{\varsigma}}_T + K_r \bar{\boldsymbol{r}} + Y\hat{\boldsymbol{\Phi}} + \boldsymbol{v}_R \tag{5.45}$$

will ensure the convergence of tracking errors $\tilde{\boldsymbol{\varsigma}}_{pv}(t), \tilde{\boldsymbol{\varsigma}}_{pm}(t), \tilde{\boldsymbol{\varsigma}}_{\varepsilon v}(t), \tilde{\boldsymbol{\varsigma}}_{\varepsilon m}(t)$ and $\bar{\boldsymbol{r}}(t)$ to zero, given that

$$K_0 = diag\{E_1^T, I_{3\times3}, J_M^T\} \in \mathbb{R}^{(6+n)\times12},$$
$$K_p = diag\{K_{v1}, k_{v2}I_{3\times3}, K_{m1}, k_{m2}I_{3\times3}\} \in \mathbb{R}^{12\times12},$$

$K_r \in \mathbb{R}^{(6+n)\times(6+n)}$ is a positive definite matrix and $\tilde{\boldsymbol{\zeta}}_T = \begin{bmatrix}\tilde{\boldsymbol{\zeta}}_{pv}^{\mathrm{T}} & \tilde{\boldsymbol{\zeta}}_{\varepsilon v}^{\mathrm{T}} & \tilde{\boldsymbol{\zeta}}_{pm}^{\mathrm{T}} & \tilde{\boldsymbol{\zeta}}_{\varepsilon m}^{\mathrm{T}}\end{bmatrix}^T$ is the composite vector. $\boldsymbol{v}_R = [\boldsymbol{v}_{R_v} \quad \boldsymbol{v}_{R_m}]^{\mathrm{T}} \in \mathbb{R}^{(6+n)}$ is a generalised vector representing an auxiliary robust controller defined by

$$\boldsymbol{v}_R = Y(\cdot)\boldsymbol{\varepsilon}(t) \tag{5.46}$$

where $\boldsymbol{\varepsilon}(t)$ is a generalised additional control input and $Y(\cdot)$ is defined in (5.44). For each subsystem, it is assumed that the positive scalar function $\rho \in \mathbb{R}$ is known *a priori* and bounded by

$$\rho \geq \left\|\tilde{\boldsymbol{\Phi}}\right\| = \left\|\hat{\boldsymbol{\Phi}} - \boldsymbol{\Phi}\right\| \tag{5.47}$$

where the vector $\boldsymbol{\Phi}$ contains uncertain parameters. $\hat{\boldsymbol{\Phi}}$ is the estimate of the parameter vector which can be computed as follows

$$\hat{\boldsymbol{\Phi}} = -\frac{2}{\sigma_r}\mathrm{e}^{-\sigma_r t}Y^T(\cdot)\bar{\boldsymbol{r}} + \boldsymbol{\Phi} \tag{5.48}$$

Substituting (5.45) into (5.43) gives the closed-loop error of the system

$$M\dot{\bar{\boldsymbol{r}}} = -Y(\cdot)\tilde{\boldsymbol{\Phi}} - K_r\bar{\boldsymbol{r}} - K_0K_p\tilde{\boldsymbol{\zeta}}_T - C\bar{\boldsymbol{r}} - D\bar{\boldsymbol{r}} - \boldsymbol{v}_R \tag{5.49}$$

The stability of the sub-region, orientation and sub-task tracking control is specified by the following theorem:

Theorem 5.2: Let the general components of σ_r given in (5.48) be a positive constant; $\boldsymbol{\Phi}$ as defined in (5.48) is the lower bound of the uncertain parameter; and ρ is the upper uncertainty bound of (5.47) and are all assumed to be initially known. Thus, the adaptive-robust control law described by (5.45), (5.46) and (5.48) with the generalised additional control input defined by

$$\boldsymbol{\varepsilon}(t) = \boldsymbol{\rho}\mathrm{e}^{-(\sigma_r/2)t} \tag{5.50}$$

ensures asymptotic sub-region, orientation and sub-tasks tracking for the underwater vehicle with a kinematically redundant on-board manipulator given by (5.43) in the sense that

$$\lim_{t\to\infty}\left[\tilde{\boldsymbol{\varsigma}}_{pv}^{\mathrm{T}}\ \tilde{\boldsymbol{\varsigma}}_{pm}^{\mathrm{T}}\ \tilde{\boldsymbol{\varsigma}}_{\varepsilon v}^{\mathrm{T}}\ \tilde{\boldsymbol{\varsigma}}_{\varepsilon m}^{\mathrm{T}}\ \tilde{\boldsymbol{\varsigma}}_{sub}^{\mathrm{T}}\right]^{\mathrm{T}} = 0 \tag{5.51}$$

provided that the initial conditions are selected such that $\tilde{\varsigma}_{ov}(0) \neq 0$ and $\tilde{\varsigma}_{om}(0) \neq 0$.

Proof: See Appendix B.

5.4.2 Weighted-Sum Approach

The proposed controllers in the preceding sections can achieve sub-region tracking and still have the redundancy of the UVMS available to perform a sub-task. This means that the self-motion of the kinematically redundant system is available to perform at least one sub-task. For the UVMS, there are several sub-tasks that can be monitored during the motion, e.g. manipulator manipulability [5.7], drag minimisation [5.10], joint range limits to avoid mechanical damage, yaw angle control to exploit the vehicle shape in the presence of ocean currents [5.11] and optimisation of the restoring moments [5.12]. Recently, a new performance index for a UVMS's redundancy resolution scheme was proposed in [5.13] where the congruent buoyancy and gravity loading of the underwater vehicle and onboard manipulator, expressed in terms of generalised velocity components, was optimised using the system's redundant degrees-of-freedom.

In order to formulate the multiple optimisation criteria for the function $H(\boldsymbol{q})$ defined in Chapter 3, a formulation technique called the weighted-sum approach can be utilised. As reported in [5.14], the overall performance after selection of proper optimisation criteria can be obtained as follows

$$H(\boldsymbol{q}) = \sum_{k=1}^{l} \alpha_k H_k(\boldsymbol{q}) \tag{5.52}$$

where α_k is the weighting factor of the sub-task, H is the scalar function expressing the k^{th} desired performance criterion and l is the maximum number of the self-motion

(sub-task) criteria. From (5.52), multiple sub-task prioritisations related to the task can be achieved by adjusting the values of α_k.

5.4.3 Multiple Sub-Task Criteria

- *Manipulability Measure or Singularity Avoidance* - Referring to the previous section, the first sub-task objective is based on singularity avoidance for a kinematically redundant onboard manipulator. Once more, the manipulability measure is defined by [5.7]

$$\mathrm{H}(\boldsymbol{q}) = \sqrt{\det(JJ^T)} \tag{5.53}$$

  where $\det(\cdot)$ represents the determinant of the matrix and J is the Jacobian matrix. If this measure is maximised, then redundancy of the system is exploited to move away from singularities.

- *Joint Limit Avoidance* - Due to the manipulator's mechanical properties, the joint angles cannot be greater or lower respectively than a specified maximum angle $q_{i_{max}}$ and a minimum angle $q_{i_{min}}$. Therefore, it is important to consider the joint limit avoidance as a sub-task for a redundant manipulator. As in [5.15], the following performance function is selected

$$H(\boldsymbol{q}) = \sum_{i=1}^{n} \frac{1}{4} \frac{\left(q_{i_{max}} - q_{i_{min}}\right)^2}{C_i\left(q_{i_{max}} - q_i\right)\left(q_i - q_{i_{min}}\right)} \tag{5.54}$$

  where n is the number of robot joints and C_i is a positive number defining the degree of strictness of the constraint for the ith joint. Note that only the manipulator joints are weighted since there are no physical limits on the vehicle's DOF. This pertinent objective function automatically gives higher weight to the joints approaching their limits and reaches infinity at the joint bounds. Accordingly, each term of the summation (5.54) takes the value one when the robot is at the furthest angle from the associated upper and lower joint limits and reaches infinity at the limits. Moreover, this function offers normalisation on the variation in the motion ranges.

Since the redundant joint has a mechanical limit, then there is a certain position where it involves the vehicle movement. Figure 5.13 shows that in order to fulfil the end-effector trajectory up to region D, the vehicle is desired to move because the second joint is getting close to its mechanical limit ($q_2 = 120°$).

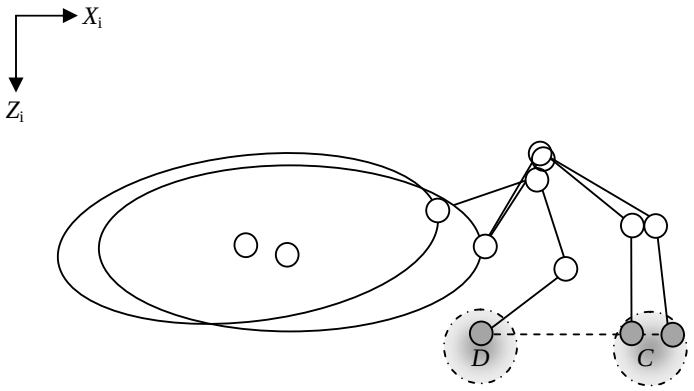


Figure 5.13: Planar view of a UVMS with the second joint approaching its limit ($q_2 = 120°$)

Using the multi-performance criteria method, each sub-task objective can be merged into a weighted sum to form an overall objective. The weight gain of each sub-task can be manipulated to change their importance in the robot's operation range. As a result, the two sub-task objectives of manipulability measure and joint limit avoidance can be merged via (5.52) to form the multi-task criteria. Assuming an equal importance for these two objectives, then (5.52) can be assigned as follows

$$H(\boldsymbol{q}) = 0.5\left(\sqrt{\det(JJ^T)}\right) + 0.5\left(\sum_{i=1}^{n}\frac{1}{4}\frac{\left(q_{i_{max}} - q_{i_{min}}\right)^2}{C_i\left(q_{i_{max}} - q_i\right)\left(q_i - q_{i_{min}}\right)}\right) \tag{5.55}$$

where α_k is chosen as 0.5; $l = 2$.

5.4.4 Simulation Results

Simulation studies have been considered for an adaptive-robust tracking controller with multiple sub-regions and sub-tasks criteria. In marine environments, it is difficult to determine the effect of ocean currents on the underwater system accurately and they are subject to change in different operating areas. These effects can be included in the

dynamic model (3.63) by considering the relative velocity $\boldsymbol{v}_{vr}$ between the vehicle and ocean current in the equation of motion [5.16]

$$\boldsymbol{v}_{vr} = \boldsymbol{v} - J_{v_e}\boldsymbol{v}_c \tag{5.56}$$

where J_{v_e} is the Jacobian matrix of the vehicle and $\boldsymbol{v}_c$ is the velocity of the ocean current expressed in the inertial-fixed frame. Note that in (5.56) only the vehicle motion is significantly affected by the presence of the ocean current due to its dimensions. Therefore, the equation of motion (3.63) can be written as

$$\boldsymbol{\tau} = M(\boldsymbol{q})\dot{\boldsymbol{\xi}}_r + C(\boldsymbol{q}, \boldsymbol{\xi}_r)\boldsymbol{\xi}_r + D(\boldsymbol{q}, \boldsymbol{\xi}_r)\boldsymbol{\xi}_r + \boldsymbol{g}(\boldsymbol{\eta}_e, \boldsymbol{q}) \tag{5.57}$$

where $\boldsymbol{\xi}_r = [\boldsymbol{v}_{vr} \quad \dot{\boldsymbol{q}}]^T$ denotes the generalised coordinates with the ocean current term. Since (5.57) requires the current measurement for dynamic compensation purposes, an alternative approach can be considered where the current effect acts as a time-varying additional disturbance $\boldsymbol{\tau}_c$ and can be included in (3.63) which leads to [5.8, 5.17]

$$\boldsymbol{\tau} - \boldsymbol{\tau}_c = M(\boldsymbol{q})\dot{\boldsymbol{\xi}} + C(\boldsymbol{q}, \boldsymbol{\xi})\boldsymbol{\xi} + D(\boldsymbol{q}, \boldsymbol{\xi})\boldsymbol{\xi} + \boldsymbol{g}(\boldsymbol{\eta}_e, \boldsymbol{q}) \tag{5.58}$$

where (5.58) is identical to the expression of the dynamic model (5.43) in view of the fact that $\boldsymbol{\tau}_c = \boldsymbol{T}_d$. To verify the robustness of the proposed adaptive robust controller, a unidirectional ocean current is defined as follows

$$\boldsymbol{v}_c = [0.3 + 0.03(\sin(0.5t)) \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T [\mathrm{m/s}] \tag{5.59}$$

Furthermore, $H(\boldsymbol{q})$ is selected as a combination of two different sub-task objectives which can be represented in the following function

$$H(\boldsymbol{q}) = 0.5(\det(J_M J_M^T)) + 0.5\left(\sum_{i=1}^{n} \frac{1}{4} \frac{\left(q_{i_{max}} - q_{i_{min}}\right)^2}{C_i\left(q_{i_{max}} - q_i\right)\left(q_i - q_{i_{min}}\right)}\right) \tag{5.60}$$

where the first term in (5.60) is chosen to maximise the manipulability and the second term attempts to ensure that the optimal link configuration is given by $\left(\frac{1}{4}\sum_{i=1}^{n}\left(q_{i_{max}} - q_{i_{min}}\right)^2 / \left(C_i\left(q_{i_{max}} - q_i\right)\left(q_i - q_{i_{min}}\right)\right)\right)$; $C_i = [0.25 \quad 0.5 \quad 0.25]^T$, $q_{i_{max}} = 2.10$ rad

and $q_{i_{min}} = -2.10$ rad; $n = 3$. Based on the multiple sub-regions approach, the vehicle is required to track the intersection of two spherical sub-regions with radii $\gamma_{v1} = 0.18$ m and $\gamma_{v2} = 0.20$ m defined by

$$f_{v1}(\delta \boldsymbol{p}_{v1}) = (x_{vd1} - x_v)^2 + (y_{vd1} - y_v)^2 + (z_{vd1} - z_v)^2 - \gamma_{v1}^2 \leq 0 \tag{5.61}$$

$$f_{v2}(\delta \boldsymbol{p}_{v2}) = (x_{vd2} - x_v)^2 + (y_{vd2} - y_v)^2 + (z_{vd2} - z_v)^2 - \gamma_{v2}^2 \leq 0 \tag{5.62}$$

The centres of both spherical sub-regions are assigned at the same point where a lower radius γ_{v1} acts as an inner sub-region while a higher radius γ_{v2} acts as an outer sub-region. Therefore, their trajectories expressed in the inertial-fixed frame can be specified as

$$\begin{bmatrix} x_{vd1} \\ y_{vd1} \\ z_{vd1} \end{bmatrix} = \begin{bmatrix} x_{vd2} \\ y_{vd2} \\ z_{vd2} \end{bmatrix} = \begin{bmatrix} 2.2 + \left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \\ 0 \\ -1.6 - \left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \end{bmatrix} \text{[m]} \tag{5.63}$$

where $t_f = 50$ s. Meanwhile, the end-effector is required to track the intersection of two circle sub-regions with radii of $\gamma_{m1} = 0.20$ m and $\gamma_{m2} = 0.25$ m defined by

$$f_{m1}\left(\delta\, {}^{v}\boldsymbol{p}_{v,m1}\right) = (x_{md1} - x_m)^2 + (z_{md1} - z_m)^2 - \gamma_{m1}^2 \leq 0 \tag{5.64}$$

$$f_{m2}\left(\delta\, {}^{v}\boldsymbol{p}_{v,m2}\right) = (x_{md2} - x_m)^2 + (z_{md2} - z_m)^2 - \gamma_{m2}^2 \leq 0 \tag{5.65}$$

The centres of the two circles move in a line expressed in the body-fixed frame and are specified as follows

$$\begin{bmatrix} x_{md1} \\ z_{md1} \end{bmatrix} = \begin{bmatrix} x_{md2} \\ z_{md2} \end{bmatrix} = \begin{bmatrix} 0.69 + 0.01\left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \\ -0.05 + 0.18\left(2\pi t/t_f - sin\, 2\pi t/t_f\right)/2\pi \end{bmatrix} \text{[m]} \tag{5.66}$$

where t_f is previously defined. During the entire simulation, the trajectory for vehicle orientation is kept constant. Initialisation of the system follows from the previous simulation.

The controller gains are chosen to be: $K_p = diag\{200, 200, 200, 40, 90, 90, 55, 55, 55\}$ and $K_r = diag\{2.0, 2.0, 2.0, 1.33, 1.0, 1.0, 0.64, 0.64, 0.64\}$. The positive constants of (5.39) and (5.40) are set to 0.8 for the inner sub-regions and 0.2 for the outer sub-regions. Since an adaptive robust controller is enforced for the vehicle only to reject the disturbance of ocean current, σ_r is selected as 0.01 and the value of ρ is initialised to 30. All the estimates of the lower bound of vehicle parameters are set to zero.

The sub-region tracking paths of the vehicle and manipulator's end effector in the 3-D operational space are presented in Figure 5.14, while Figure 5.15 shows the effect of the multiple sub-tasks on the UVMS with a planar plane presentation. As can be seen in Figures 5.14 and 5.15, both subsystems start inside their individual inner sub-regions and they initially move out from the sub-region due to the effect of the unidirectional ocean current. However, they converge into their individual sub-region intersections at the steady-state regardless of the multiple sub-tasks. Note that, since the end-effector position is vertically controllable, the position in the sway direction follows the position of the vehicle. Figures 5.14 and 5.15 also show the consequence of the multiple sub-tasks objectives that are formulated in (5.60). The manipulator has to reconfigure itself in a dexterous posture in order to keep the second joint within a safe working configuration as can be clearly seen in Figure 5.16(a).

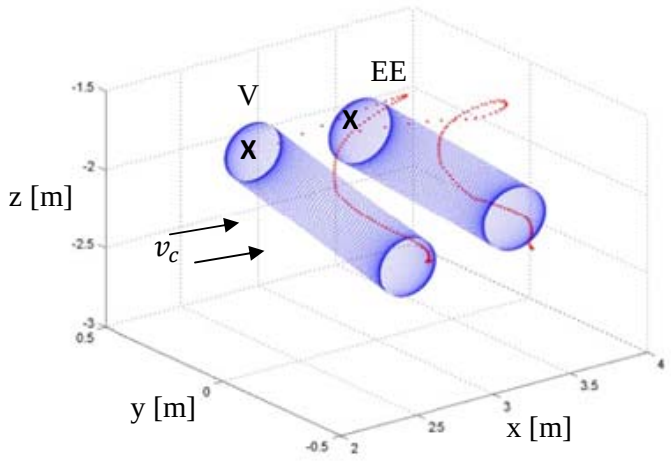


Figure 5.14: Desired sub-region and actual tracking trajectories

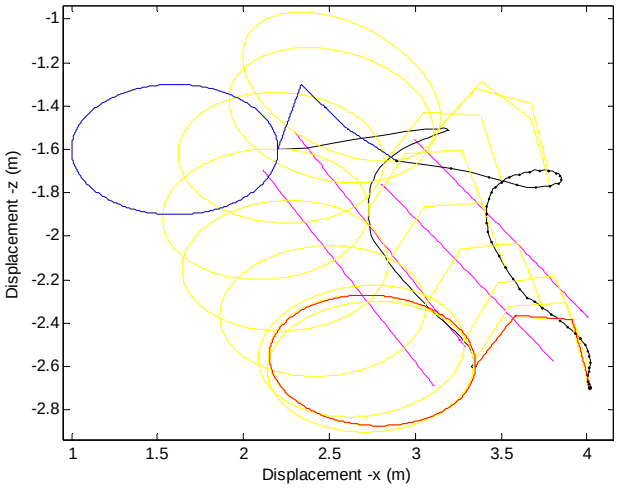
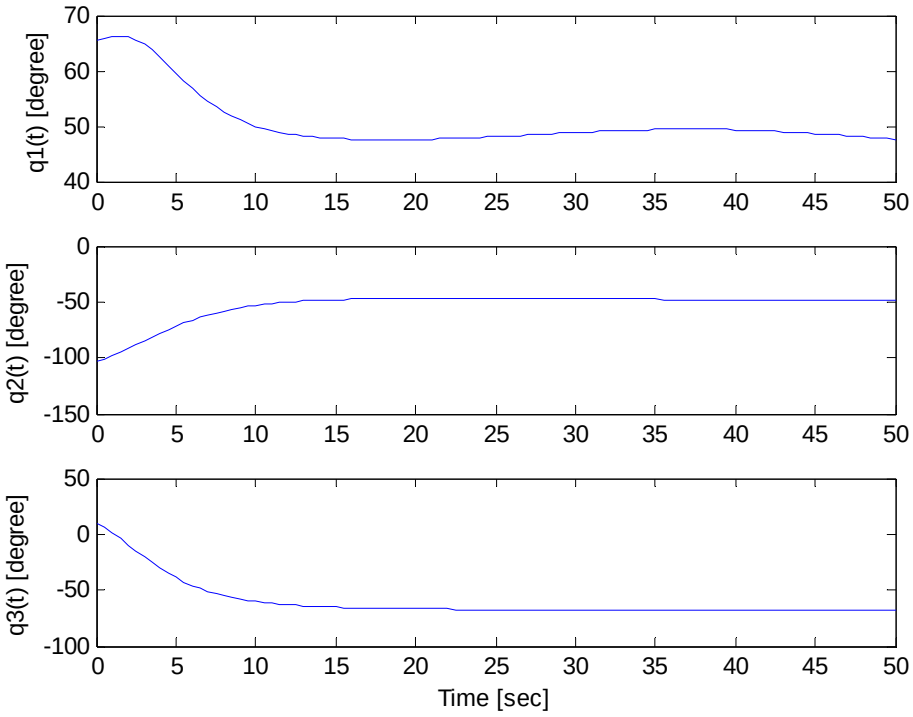
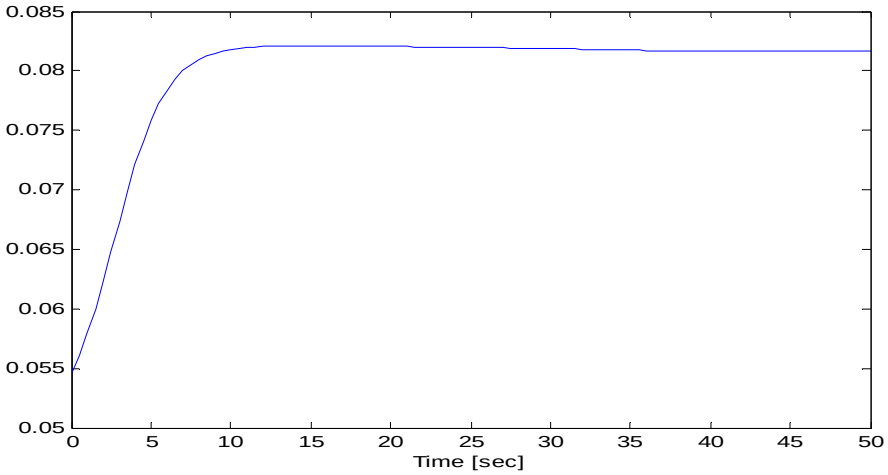


Figure 5.15: A UVMS configuration with multiple sub-region and sub-task criteria in a planar plane

The manipulator is outstretched at $q_2 \approx 0$ [rad] and the vehicle is required to move to avoid this singular configuration. Similarly, the second joint reaches its mechanical joint limit at $q_2 = \pm\pi/1.5$ [rad] which also requires vehicle movement. Therefore, it is preferable for the manipulator not to move close to these critical situations when it can be achieved using the self-motion tracking. In Figure 5.16(b), the performance measure of multiple sub-task objectives is presented while Figure 5.17 depicts the vehicle orientation tracking errors with the quaternion representation converging to zero for $\tilde{\varsigma}_{\varepsilon v}$ and unity for $\tilde{\varsigma}_{ov}$ after a short initial transient period.



(a)



(b)

Figure 5.16: (a) The joint position for onboard manipulator when q_2 is kept away from its mechanical limit ($q_2 = -120°$) and kinematic singularity ($q_2 \approx 0°$); (b) The performance index of multiple sub-task objectives

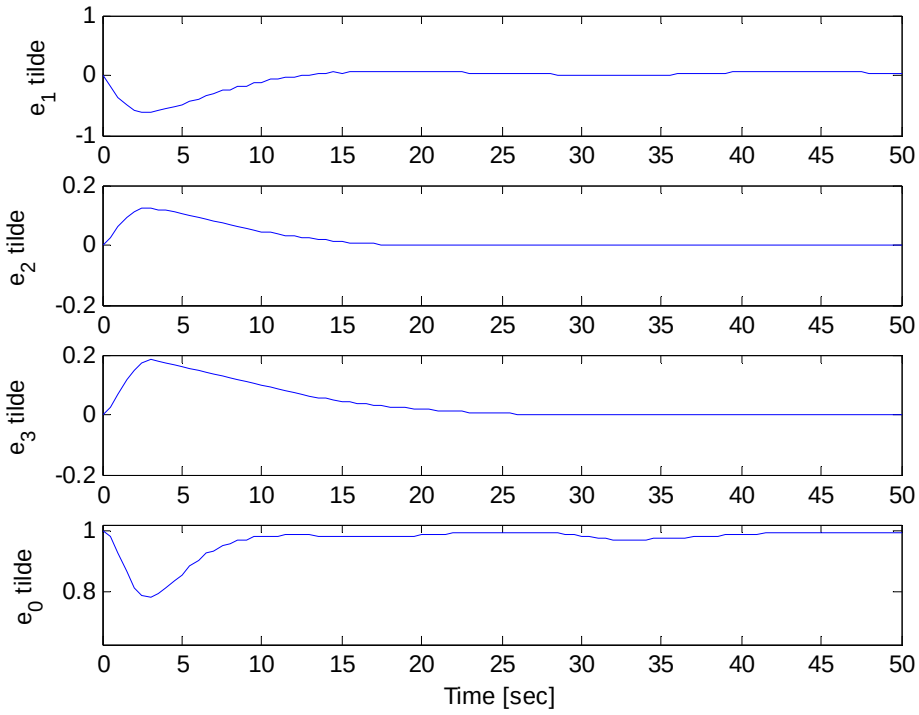


Figure 5.17: Quaternion based vehicle orientation tracking errors

5.5. Summary

The proposed approaches using sub-region and sub-task tracking control for redundant underwater robotic systems has been presented in this chapter. Various simulations for a six degrees-of-freedom vehicle equipped with a three degrees-of-freedom robotic arm have been performed to demonstrate the effectiveness of the proposed tracking controllers. Some conclusions can be stated:

- It was shown that the origin of the robot system in a closed-loop equation with the computed-torque controller is globally uniformly asymptotically stable. Since the closed-loop equation is linear autonomous, it was observed that this is equivalent to global exponential stability. This claim can be verified using Theorem A.5.

- By the use of a filtered tracking error-like term, all tracking objectives are ensured to be asymptotically stable using the proposed controller. The applied control strategy uses the pseudo-inverse of the manipulator Jacobian and does not require computation of the inverse kinematics.

- An adaptive-robust controller that achieves tracking under the influence of modelling uncertainties and an additive disturbance was also presented. Results from simulation studies were presented to verify the viability of the adaptive-robust controller. It was shown that the proposed control scheme guarantees the robustness against parametric uncertainty and the effect of a unidirectional current.

- The extra degrees-of-freedom of the onboard manipulator are used in the control law to perform a sub-task; i.e. drag minimisation, obstacle avoidance, manipulability, or avoidance of mechanical joint limits.

- The formulation of multiple criteria was performed when the system is subject to multiple sub-task tracking objectives.

5.6. References

[5.1] C. C. Cheah and Y. C. Sun, "A Region Reaching Control Scheme for Underwater Vehicle-Manipulator Systems," in *Proc. IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007, pp. 4576-4579.

[5.2] Y. C. Sun and C. C. Cheah, "Adaptive Control Schemes for Autonomous Underwater Vehicle," *Robotica*, vol. 27, pp. 119-129, 2009.

[5.3] S. P. Hou and C. C. Cheah, "Region Tracking Control for Robot Manipulators," in *IEEE International Conference on Control Applications*, Singapore, 2007, pp. 1438-1443.

[5.4] F. Lizarralde and J. T. Wen, "Attitude Control Without Angular Velocity Measurement: A Passivity Approach," in *IEEE Transactions on Automatic Control*, vol. 41, 1996, pp. 468-472.

[5.5] Y. Yamamoto and X. Yun, "Modeling and Compensation of the Dynamic Interaction of a Mobile Manipulator," in *Proc. IEEE Conf. Robotics and Automation*, vol. 1, 1994, pp. 1-6.

[5.6] P. Hsu, J. Hauser, and S. Sastry, "Dynamic Control of Redundant Manipulators," *Journal of Robotic Systems*, vol. 6, pp. 133-148, 1989.

[5.7] T. Yoshikawa, *Foundations of Robotics: Analysis and Control*.: MIT Press, 1990.

[5.8] G. Antonelli, "On the Use of Adaptive/Intergral Actions for 6-Degrees-of-Freedom Control of Autonomous Underwater Vehicles," *Journal of Oceanic Engineering*, vol. 32, no. 2, pp. 300-312, 2007.

[5.9] C. C. Cheah and D. Q. Wang, "Region Reaching Control of Robots: Theory and Experiments," in *Proc. of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 974-979.

[5.10] N. Sarkar and T. K. Podder, "Coordinate Motion Planning and Control of Autonomous Underwater Vehicle-Manipulator Systems Subject to Drag Optimization," *IEEE Journal Oceanic Engineering*, vol. 26, no. 2, pp. 228-239, April 2001.

[5.11] G. Antonelli, *Underwater Robots: Motion and Force Control of Vehicle-Manipulator Systems*. Germany: Springer-Verlag Berlin, 2003.

[5.12] J. Han and W. K. Chung, "Redundancy Resolution for Underwater Vehicle-Manipulator Systems with Minimizing Restoring Moments," in *Proc. IEEE/RJS International Conference on Intelligent Robots and Systems*, San Diego, CA, 2007, pp. 3522-3527.

[5.13] Z. H. Ismail and M. W. Dunnigan , "Redundancy Resolution for Underwater Vehicle-Manipulator Systems with Congruent Gravity and Buoyancy Loading Optimization," in *Proc. of the IEEE Int. Conference on Robotics and Biomimetic*, Guilin, China, 2009, pp. 1393-1399.

[5.14] K. Cleary and D. Tesar, "Incorporating Multiple Criteria in the Operation of Redundant Manipulators," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Cincinnati, Ohio, May 1990, pp. 618-624.

[5.15] T. F. Chan and R. V. Dubey, "A Weighted Least-Norm Solution Based Scheme for Avoiding Joint Limits for Redundant Manipulators," in *IEEE Transaction on Robotics and Automation*, 11(2), 1995, pp. 286-292.

[5.16] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, 1st ed. New York: John Wiley and Sons, 1994.

[5.17] G. Antonelli, S. Chiaverini, N. Sarkar and M. West, "Adaptive Control of an Autonomous Underwater Vehicle: Experimental Results on ODIN," in *IEEE Transaction on Control Systems and Technology*, vol. 9, 2001, pp. 756-765.

CHAPTER 6
CONCLUSIONS

6.1. General Conclusions

In this thesis, the development of new control laws for regulation tracking of underwater robotic systems has been presented. The kinematic and dynamic models representing the underwater robotic systems were initially reviewed to enable the proposed controllers to be tested through simulation. This also provides a thorough understanding of the dynamic model and its fundamental properties which are essential for stability analysis of the controllers via Lyapunov's method.

The task-space regulation problem of an underwater vehicle was then investigated. It is interesting to note that the desired target for the regulation problem is commonly specified as a point. The multiple targets are defined as sub-regions and a new sub-region priority reaching approach has been proposed for an underwater vehicle. Using the region-decomposition method, the operational space is broken down into multiple sub-regions with a priority order. The multiple sub-region criteria are: depth constraint, avoiding the region with an obstacle inside it, ensuring visibility of the feature during visual servoing and complexity region constraint. The fuzzy technique is used to handle these multiple sub-region criteria effectively. Due to the unknown gravitational and buoyancy forces, an adaptive term was also used in the proposed sub-region priority reaching controller.

An extension to a region boundary-based control law was then proposed for an underwater vehicle in order to illustrate the flexibility of the sub-region reaching concept. In this novel control law, a desired target is defined as a boundary instead of a point or a region. A region boundary-based control concept is regarded as a generalised region or set-point control problem where the system is regulated to move to the region boundary rather than into a region or a point. Therefore, the initial position of a vehicle can be either inside or outside of a region prior to its convergence into the boundary area within a specific time. In addition, the proposed control laws are also applicable for a UVMS where a sub-region boundary is defined for each sub-system. For a mapping of the uncertain restoring forces, a least-squares estimation algorithm and the inverse Jacobian matrix are utilised in the adaptive control law.

A sub-region tracking control scheme with a sub-task objective was developed for a UVMS with the purpose of developing a new tracking control concept for a kinematically redundant underwater robot. Using this concept, the desired objective is specified as a moving sub-region instead of a trajectory. In addition, due to the system being kinematically redundant, the controller also enables the use of self-motion of the system to perform sub-tasks, for instance, drag minimisation, obstacle avoidance, manipulability and avoidance of mechanical joint limits. Therefore, it can be ensured from the control formulation that the sub-task tracking is achievable without affecting the sub-region trajectory tracking objective.

An adaptive and robust sub-region tracking control scheme was then applied to overcome the influence of modelling uncertainties and additive bounded disturbances. The control task of tracking a prescribed sub-region trajectory is challenging due to the presence of variable ocean currents creating hydrodynamic forces and moments that are unknown and unpredictable. When the UVMS was subject to multiple sub-task tracking objectives, a weighted sum approach was used to formulate the performance optimisation criteria. Simulation results were presented to illustrate the effectiveness of all developed controllers.

To summarise, the contributions of this thesis to the subject of underwater robot control are:

Several new control approaches can be effectively applied to achieve the regulation and trajectory tracking objectives of underwater robotic systems. The controllers are designed to allow the underwater robots to operate in unknown and variable environments, and also to extend the capabilities of the robots by realising position and attitude control in Cartesian space under the influence of bounded disturbances.

The classes of controllers developed in this thesis are novel in both their application and their design formulation. The application of sub-region priority and region boundary based controllers as an alternative method for the set-point control problem has not been reported before. Moreover, this is the first instance of such a sub-region and sub-task tracking control scheme being used for an underwater vehicle-manipulator system. The use of an adaptive and robust control element in the design formulation of the proposed

controllers also has a degree of originality. The controllers developed in this thesis were placed in the context of previously proposed set-point and trajectory tracking underwater robot controllers. Chapters 4 and 5 give the details of how they complement previously reported work and their novel features.

6.2. Suggestions for Future Work

There are several areas where the work presented in this thesis can be usefully extended. Firstly, it would be interesting to investigate whether the developed controllers can solve the control synthesis problem for an underactuated AUV, i.e. an AUV with fewer control actuators than the number of desired degrees-of-freedom. For example, a three-dimensional attitude control problem for an AUV with only two actuators. Consequently, an AUV can perform its motion with a reduced number of actuators. This might be the result of a failure of an actuator or a deliberate decision to limit the number and choice of actuators in use, e.g. for cost effectiveness.

Secondly, for an underwater robotic system to perform a completely autonomous mission, it is desirable to allow the system to interact with the environment by using an onboard manipulator. In order to achieve this, an extension to an interaction control technique needs to be properly investigated. In Chapter 6, the control formulation is only useful when there is no contact between the robot's end-effector and environment. When in contact with the environment, interaction control, i.e. an impedance control scheme, can be adopted to achieve the behaviour of a desired mechanical impedance at the end-effector.

A region boundary-based shape controller could be proposed for coordinated control of multiple AUVs. The use of several AUVs to achieve a desired task could be beneficial in several examples such as ocean exploration or mine countermeasure. It would be interesting to investigate a new cooperative controller for this multi-robot system. Within this control framework, AUVs could be allowed to move as a group inside a desired region boundary while maintaining a minimum distance between themselves. Various desired region shapes could also be formed by choosing the appropriate objective functions.

Finally, it would be very interesting to investigate the effectiveness of the developed controllers if they were implemented in real underwater robotic systems. As presented in this thesis, the implementation of the proposed control approaches is feasible through the simulation studies in which the mathematical expression of the kinematic and dynamic models is used to obtain the system's output responses. However, it would be more challenging to examine the performance of the controllers in a practical environment due to various factors including the need for a variety of sensory systems.