National Technical University of Athens

PhD Dissertation

Cooperative and Interaction Control for Underwater Robotic Vehicles

Author:
Shahab Heshmati-alamdari

Advisor:
Prof. Kostas J. Kyriakopoulos

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Control Systems Lab
School of Mechanical Engineering

December, 2018

National Technical University of Athens

PhD Dissertation

Cooperative and Interaction Control for Underwater Robotic Vehicles

Author:
Shahab Heshmati-alamdari

Advisor:
Prof. Kostas J. Kyriakopoulos

Examination Committee:
Prof. K. J. Kyriakopoulos
Prof. E. Papadopoulos
Prof. I. Antoniadis
Prof. D. Dimarogonas
Prof. H. Sarimveis
Assist. Prof. G. Papalambrou
Associate Prof. C. Tzafestas

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Control Systems Lab
School of Mechanical Engineering

December, 2018

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Μεθοδολογίες Ελέγχου Συνεργασίας και Αλληλεπίδρασης για Υποβρύχια Ρομποτικά Οχήματα

Συγγραφέας:
Σαχαμπόντιν
Χεσματί-αλαμντάρι

Επιβλέπων καθηγητής:
Κώστας Κυριακόπουλος

Η διατριβή υποβάλλεται εκπληρώνοντας τις απαιτήσεις για την απονομή του Διδακτορικού Διπλώματος

σύμφωνα με το

Εργαστήριο Αυτομάτου Ελέγχου
Σχολή Μηχανολόγων Μηχανικών

Δεκέμβριος, 2018

Athens, December 2018.

NATIONAL TECHNICAL UNIVERSITY OF ATHENS

Abstract

School of Mechanical Engineering

Doctor of Philosophy

Cooperative and Interaction Control for Underwater Robotic Vehicles

by Shahab Heshmati-alamdari

In this dissertation we address the problem of robust control for underwater robotic vehicles under resource constraints and inspired by practical applications in the field of marine robotics. By the term "resource constraints" we refer to systems with constraints on communication, sensing and energy resources. Within this context, the ultimate objective of this dissertation lies in the development and implementation of efficient control strategies for autonomous single and multiple underwater robotic systems considering significant issues such as: external disturbances, limited power resources, strict communication constraints along with underwater sensing and localization issues. Specifically, we focused on cooperative and interaction control methodologies for single and multiple Underwater Vehicle Manipulator Systems (UVMSs) considering the aforementioned issues and limitations, a topic of utmost challenging area of marine robotics. More precisely, the contributions of this thesis lie in the scope of three topics: i) Motion Control, ii) Visual servoing and iii) Interaction&Cooperative Transportation. In the first part, we formulated in a generic way the problem of Autonomous Underwater Vehicle (AUV) motion operating in a constrained environment including obstacles. Various constraints such as: obstacles, workspace boundaries, thruster saturation, system's sensing range and predefined upper bound of the vehicle velocity are considered during the control design. Moreover, the controller has been designed in a way that the vehicle exploits the ocean currents, which results in reduced energy consumption by the thrusters and consequently increases significantly the autonomy of the system. In the second part of the thesis, we have formulated a number of novel visual servoing control strategies in order to stabilize the robot (or robot's end-effector) close to the point of interest considering significant issues such as: camera Field of View (FoV), Camera Calibration uncertainties and the resolution of visual tracking algorithm. In the third part of the thesis, regarding the interaction task, we present a robust interaction control scheme for a UVMS in contact with the environment, with great applications in underwater robotics (e.g. sampling of the sea organisms, underwater welding, object handling). The proposed control scheme does not required any a priori knowledge of the UVMS dynamical parameters

or the stiffness model. It guarantees a predefined behavior in terms of desired overshoot, transient and steady state response and it is robust with respect to external disturbances and measurement noises. Moreover, we have addressed the problem of cooperative object transportation for a team of UVMSs in a constrained workspace involving static obstacles. First, for case when the robots are equipped with appropriate force/torque sensors at its end effector we have proposed a decentralized impedance control scheme with the coordination relying solely on implicit communication arising from the physical interaction of the robots with the commonly grasped object. Second, for case when the robots are not equipped with force/torque sensor at it end effector, we have proposed a decentralized predictive control approach which takes into account constraints that emanate from control input saturation as well kinematic and representation singularities. Finally, numerical simulations performed in MATLAB and ROS environments, along with extensive real-time experiments conducted with available Control Systems Lab (CSL) robotic equipment, demonstrate and verify the effectiveness of the claimed results.

ΕΘΝΙΚΟ ΜΕΤΣΟΒΕΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Περίληψη

Σχολή Μηχανολόγων Μηχανικών

Διδακτορική Διατριβή

Μεθοδολογίες ελέγχου συνεργασίας και Αλληλεπίδρασης για υποβρύχια ρομποτικά οχήματα

Σαχαμπόντιν Χεσματί-αλαμντάρι

Σε αυτή τη διατριβή αντιμετωπίζουμε το πρόβλημα του ευρωστου ελέγχου υποβρύχιων ρομποτικών οχημάτων υπό περιορισμούς πόρων, εμπνευσμένα από πρακτικές εφαρμογές στον τομέα της θαλάσσιας ρομποτικής. Με τον όρο «περιορισμούς των πόρων» αναφέρομαστε σε συστήματα με περιορισμούς στην επικοινωνία, την ανίχνευση και τους ενεργειακούς πόρους. Στο πλαίσιο αυτό, ο απώτερος στόχος της παρούσας διατριβής έγκειται στην ανάπτυξη και υλοποίηση αποτελεσματικών στρατηγικών ελέγχου για αυτόνομα ενιαία και πολλαπλά υποβρύχια ρομποτικά συστήματα, λαμβάνοντας υπόψη σημαντικά ζητήματα όπως: εξωτερικές διαταραχές, περιορισμένες πηγές ενέργειας, αυστηρούς περιορισμούς επικοινωνίας μαζί με υποβρύχια ανίχνευση και εντοπισμό θέματα. Συγκεκριμένα, εστιάσαμε στις μεθοδολογίες ελέγχου αλληλεπίδρασης για μονά και πολλαπλά υποβρύχια οχήματα με ρομποτικό βραχίονα (UVMS), εξετάζοντας τα προαναφερθέντα ζητήματα και περιορισμούς, ένα θέμα εξαιρετικά προκλητικής περιοχής της θαλάσσιας ρομποτικής. Πιο συγκεκριμένα, οι συνεισφορές αυτής της διατριβής βρίσκονται στο πεδίο εφαρμογής τριών θεμάτων: i) Ελέγχου κίνησης, ii) οπτική ανατροφοδότηση και iii) Αλληλεπίδραση & Συνεργατική μεταφορά. Στο πρώτο μέρος, διατυπώσαμε με γενικό τρόπο το πρόβλημα της κίνησης ενος αυτόνομου υποβρυχίου οχήματος (AUV) που λειτουργεί σε περιορισμένο περιβάλλον, συμπεριλαμβανομένων των εμποδίων. Διάφοροι περιορισμοί όπως: εμπόδια, τα όρια του χώρου εργασίας, ανω όριο κορεσμού του προωθητήρα, το εύρος ανίχνευσης του συστήματος και το προκαθορισμένο ανώτερο όριο της ταχύτητας του οχήματος, λαμβάνονται υπόψη κατά τη διάρκεια του σχεδιασμού ελεγκτή. Επιπλέον, ο ελεγκτής έχει σχεδιαστεί με τέτοιο τρόπο έτσι ώστε το όχημα να εκμεταλλεύεται τα θαλάσσια ρεύματα, γεγονός που έχει ως αποτέλεσμα τη μειωμένη κατανάλωση ενέργειας από τους προωθητήρες και συνεπώς αυξάνει σημαντικά την αυτονομία του συστήματος. Στο δεύτερο μέρος της διατριβής, διατυπώσαμε μια σειρά από νέες στρατηγικές ελέγχου μέσω οπτικής ανατροφοδοτησής, προκειμένου

να σταθεροποιήσουμε το ρομπότ (ή τον τελικό τελεστή του ρομπότ) κοντά στο σημείο ενδιαφέροντος λαμβάνοντας υπόψη σημαντικά θέματα όπως: κάμερα οπτικής απεικόνισης (FoV), αβεβαιότητες βαθμονόμησης κάμερας και ανάλυση αλγορίθμου οπτικής παρακολούθησης. Στο τρίτο μέρος της διατριβής, σχετικά με την αλληλεπίδραση, παρουσιάζουμε ένα έυρωστο σχήμα ελέγχου αλληλεπίδρασης για ένα UVMS σε επαφή με το περιβάλλον, με σπουδαίες εφαρμογές στην υποβρύχια ρομποτική (π.χ. δειγματοληψία των θαλάσσιων οργανισμών, υποβρύχια συγκόλληση, χειρισμός αντικειμένων). Το προτεινόμενο σχήμα ελέγχου δεν απαιτεί εκ των προτέρων γνώση των δυναμικών παραμέτρων του UVMS ή του μοντέλου ακαμψίας. Εξασφαλίζει μια προκαθορισμένη συμπεριφορά όσον αφορά την επιθυμητή υπέρβαση, παροδική και σταθερή αντίδραση και είναι έυρωστη σε σχέση με τις εξωτερικές διαταραχές και τους θορύβους μέτρησης. Επιπλέον, αντιμετωπίσαμε το πρόβλημα της μεταφοράς συνεταιριστικών αντικειμένων για μια ομάδα UVMS σε ένα περιορισμένο χώρο εργασίας που περιλαμβάνει στατικά εμπόδια. Πρώτον, για την περίπτωση που τα ρομπότ είναι εφοδιασμένα με τους κατάλληλους αισθητήρες δύναμης / ροπής στο τελικό στοιχείο δράσης τους, έχουμε προτείνει ένα αποκεντρωμένο σχέδιο ελέγχου σύνθετης αντίστασης με τον συντονισμό να στηρίζεται αποκλειστικά σε σιωπηρή επικοινωνία που προκύπτει από τη φυσική αλληλεπίδραση των ρομπότ με το αντικείμενο που πιάστηκε συνήθως. Δεύτερον, για περιπτώσεις όπου τα ρομπότ δεν είναι εφοδιασμένα με αισθητήρα δύναμης / ροπής στο τελικό στοιχείο δράσης τους, έχουμε προτείνει μια προσέγγιση αποκεντρωμένης πρόβλεπτκού ελέγχου, η οποία λαμβάνει υπόψη τους περιορισμούς που απορρέουν από τον κορεσμό εισόδου ελέγχου καθώς και από τις κινηματικές περιορισμούς και ιδιότητες. Τέλος, πολλαπλές αριθμητικές προσομοιώσεις που διεξάγονται σε περιβάλλοντα MATLAB και ROS, μαζί με εκτεταμένα πειράματα σε πραγματικό χρόνο που διεξάγονται με τον διαθέσιμο ρομποτικό εξοπλισμό του εργαστηριού αυτομάτου έλεγχου (Control Systems Lab, CSL), αποδεικνύουν και επαληθεύουν την αποτελεσματικότητα των ισχυρισθέντων αποτελεσμάτων.

Acknowledgements

This Ph.D. Dissertation could not have been completed without the help and support of a lot of people.

Foremost, I would like to truly thank my advisor Prof. Kostas J. Kyriakopoulos for entrusting me and giving me the opportunity to work with him as well as within a number of European projects to contact research with various research groups around the European Union. His advices and guidance, on my research and my career, have been priceless and I will always remember his patience and support until the completion of this thesis. I would also like to thank my committee members, Prof. Evangelos Papadopoulos, and Prof. Ioannis Antoniadis for serving as my committee members and for their brilliant comments and suggestions.

I am especially grateful to my colleagues Dr. Alina Eqtami, Dr. George C. Karras and Dr. Charalampos P. Bechlioulis for their exceptional contribution to my thesis. Alina helped me a lot during my first steps on control design. George helped me greatly during all of my experiments at a level that without him they could not have been completed. Charalambos, my "mentor in control," gave me a lot of advice and guidance in addressing the theoretical challenges in nonlinear control design. Their valuable suggestions and unwavering support during this dissertation were of utmost importance at a level that could not be completed without their support.

I would like to thank to Professor Dimos V. Dimarogonas for his invitation to visit the KTH Royal Institute of Technology in Stockholm and work with him on the needs of a European research program at the Winter of 2016, but also on our collaboration so far.

I would like to express my sincere thanks to my beloved girlfriend Georgia Palaiologou for her unlimited faith, support and encouragement throughout the last difficult stages of this memorable experience. I would like also to show my gratitude to Lucita, for bringing me good luck and for accompanying me throughout these years.

As far as my family is concerned, I deeply thank my father Abbas, my mother Nayer my brother Hesam and my sister Atie for their psycological support on this journey from far away, even in my most difficult moments.

Last but not least, I will never forget the other past and current members of the Control System Lab, Minas Liarokapis, Panos Marantos, Yannis Roussos, Panagiotis Vlantis, George Zogopoulos-Papaliakos, Michalis Logothetis, Constantinos Vrohidis for the great moments we had together. A special thank to Michael Drossakis for his efforts in facilitating the research conducted in CSL.

Athens, December 2018 *Shahab Heshmati-alamdari*

Table of Contents

List of Figures

List of Tables

List of Acronyms

AHRS Attitude and Heading Reference System

AUV Autonomous Underwater Vehicle

CF Complementary Filter

DOF Degrees-of-Freedom

DVL Doppler Velocity Log

EKF Extended Kalman Filter

FHOCP Finite Horizon Optimal Control Problem

FoV Field of View

FPS Frames Per Second

GPS Global Positioning System

IBVS Image Based Visual Servoing

IMU Inertial Measurement Unit

INS Inertial Navigation System

LVS Laser Vision System

MPC Model Predictive Control

NED North-East-Down

OCP Optimal Control Problem

PBVS Position Based Visual Servoing

PPC Prescribed Performance Control

ROS Robotic Operation System

RRTs Exploring Random Trees

SVD Singular Value Decomposition

USBL Ultra Short Baseline

UUV Unmanned Underwater Vehicle

UVMS Underwater Vehicle Manipulator System

VTA Visual Tracking Algorithm

To my beloved parents for their love,
endless support,
encouragement and sacrifices

Part I - Introduction and Problem Statement

Chapter 1

Introduction

1.1 Motivation

During the last decades, considerable progress has been made in the field of unmanned marine vehicles, with a significant number of results in a variety of marine activities [3]. Applications such as marine science (e.g., biology, oceanography, archeology,ocean forecasting, pollution management, ecosystem monitoring) and offshore industry (e.g., ship maintenance, inspection of oil/gas facilities, cable burial, mating of underwater connectors, underwater welding) are indicative examples of applications that require the underwater robots to operate under various constraints and increased level of autonomy and endurance [4–7]. Moreover, a vast number of the aforementioned applications demand the underwater vehicle to be enhanced with intervention capabilities as well [8, 9], thus raising increasing interest on Underwater Vehicle Manipulator Systems (UVMSs) [10, 11]. A UVMS consists of an underwater robotic vehicle equipped with robotic manipulators. Currently, underwater intervention is carried out by Remotely Operated Vehicles (ROVs) equipped with one or multiple manipulators, which allow it to grasp, transport and manipulate objects. However, most ROVs are controlled by a human pilot on a surface ship, via a master-slave tele-operation scheme [12–14]. Therefore, the well-known disadvantages of human-robot tele-operation (e.g., time delays and increase of human fatigue over time) necessitate automated and efficient solutions to reduce costs and safety risks. This inevitably leads to the newal of ROVs with Autonomous Underwater Vehicles (AUVs) as well as to the development of autonomous intervention control platforms like the UVMSs that have attracted significant scientific interest during the last years [10, 15, 16]. Specifically, during the late 90s, efforts on the design of UVMSs were made within the pioneering AMADEUS project [17], which were later exploited in UNION [18] and SAUVIM projects [15], where autonomous underwater intervention

was carried out for the first time. A more recent European project which has boosted the underwater robotic interaction with relevant results was TRIDENT [19–24], where a vehicle-arm system was controlled in a coordinated manner. Another important milestone was achieved in the PANDORA project [25–27], where a strong emphasis was given on the issue of persistent autonomy. The latest related project in the domain of underwater intervention is the on-going European project DexROV [28], which focuses on inspection and maintenance tasks in the presence of communication latencies.

Within this context and motivated by the above, the objective and main contribution of this dissertation lies in the development and implementation of efficient control strategies for autonomous single and multiple underwater robotic systems, by also considering significant issues such as: external disturbances, limited power resources, various system constraints, strict communication constraints along with underwater sensing and localization issues. Specifically, we focused on cooperative and interaction control methodologies for single and multiple Underwater Vehicle Manipulator Systems (UVMSs) considering the aforementioned issues and limitations, a topic of utmost challenging area of marine robotics. Initially we divided the problem of control and cooperation between UVMSs in *three main parts*: **i) Motion Control**, **ii) Visual servoing** and **iii) Interaction/-Transportation**. Within each parts, we formulate and present methodologies and control algorithms for solving the corresponding problems and aim to achieve and satisfy the aforementioned constraints and limitation introduced from the nature of underwater environment.

1.2 Thesis Outline and Contributions

In this Section, we provide the outline of the thesis and indicate the contributions of each chapter.

Chapter 2 describes the general form of the kinematic and dynamic equations of a robotic underwater vehicle and an underwater vehicle manipulator system that will be adopted in this thesis.

1.2.1 Motion Control part:

As mentioned previously, applications such as ocean forecasting, pollution management, ecosystem monitoring, underwater inspection and surveillance, marine resource utilization and oceanography are examples of applications that require the underwater robots to operate under various constraints and increased level of autonomy and endurance [4–7].

On the design side, the endurance of an underwater system can be increased by improving the energy storage units (e.g., larger capacity batteries) [29–31] or reducing the vehicle's drag by design [32]. However, such approaches may solve partially the problem, due to current technology limitations, design constraints and reduced payload capabilities of the vehicle. Therefore, alternative approaches for energy reduction may considered. In general, the energy intake of an underwater vehicle is divided in two parts: i) the hotel load which is defined as the energy consumption of the on-board computers, processing effort, instrumentation and communication devices, and ii) the energy part used by the propulsion system (e.g thrusters) [6]. The hotel load reduction can be achieved by employing low power devices and lean algorithms that do not require significant processing effort. On the other hand, the optimization of the thrust energy consumption, is mainly a path planning problem where the vehicle must reach the desired goal in an energy optimal manner.

Energy minimization via mission planning has been studied in the past and still remains an open research issue for the underwater robotics community. The ocean currents may significantly affect the vehicle motion and must be taken under consideration in the control design [33]. The importance of utilizing ocean currents in underwater vehicle operation was emphasized in [34], where a genetic algorithm planner was proposed for the design of a path with minimum energy requirements. In [35], the authors proposed a path planning method for an AUV based on the A^* notion. In [36], an A^* energy efficient framework was proposed under the consideration of quasi-static ocean current information and constant thrust power. In [37], an A^* algorithm with energy based cost function was presented. Moreover, in [38], an A^* search was employed in order to design a continuous path, where the ocean currents were incorporated as quadratic drag force terms.

Alternative methods for the path planning of underwater vehicles are based on Rapidly Exploring Random Trees (RRTs) [39, 40], where the workspace is explored in order to navigate the robot uniformly. A similar method to obtain an obstacle free path for an AUV was employed in [41]. However, no sea currents were taken under consideration during the design of the algorithm. In [42], RRTs were combined with A^* in order to generate feasible and energy optimized paths for gliders. The majority of the aforementioned planning techniques are based on off–line optimization schemes, which consider static or quasi–static operational environments. Their output is often a set of way-points or trajectories, which are optimal with respect to the energy consumption, while satisfy certain environmental constraints.

However, in real–time missions, the vehicle operates in a partially known and dynamic

environment where the knowledge of the operating workspace is constantly updated on–line via the vehicle's on–board sensors (e.g multi beam imaging sonars, on-line ocean current estimators). In these cases, the underwater vehicle must re-calculate its path on–line according to possible environmental changes (i.e., new obstacles, other vehicles or humans operating in proximity etc.). Generally, this conclusion can be drawn, the motion control of underwater robotic vehicles is a highly demanding task with great challenges imposed by external disturbances, model uncertainties, energy consumption and constraints of the operating workspace. Thus, robust motion control is still an open issue for the underwater robotics community.More details regarding the open problems on underwater robotics path planning, can be found in [6] and [33] and the papers quoted therein.

On the other hand, motion control of underwater vehicles is a highly nonlinear problem, where multiple input and state constraints are imposed to the system. Various strategies, such as PID [43], feedback and linearization [44], [45], sliding mode [46–48], adaptive control [49–52], learning [53, 54], fuzzy logic [55], hybrid [56] and prescribed performance [57] have been proposed in literature for the motion control of underwater vehicles. However, by employing the aforementioned strategies, it is not always feasible or straightforward to incorporate input (generalized body forces/torques or thrust) and state (3D obstacles, velocities) constraints into the vehicle's closed-loop motion. In that sense, the motion control problem of underwater robots continues to pose considerable challenges to system designers, especially in view of the high-demanding missions envisioned by the marine industry (e.g., ship hull inspection, surveillance of oil platforms, cable installation and tracking, etc.). In this context, Nonlinear Model Predictive Control (NMPC) [58], can be considered an ideal approach for complex underwater missions, as it is able to combine motion planning, obstacle avoidance and workspace restrictions, while handling efficiently input and state constraints.

A sampling based Model Predictive Control scheme was proposed in [59] for motion control of underwater vehicles in presence of constraints. A depth control strategy for an over-actuated hover-capable AUV based on Linear Model Predictive Control was presented in [60]. However, actuator limits were the only considered constraints of the system. In the same paper, the authors conclude that MPC is applicable for AUV control, while there are still open issues in topics such as robustness and estimation of the safe operating region (i.e., state constraints). In [61], the authors propose an MPC scheme in order to design an energy efficient path for a glider, by minimizing a cost function based on the consumed energy. However, only the kinematic model of the vehicle was considered, without taking into account any disturbances or noise of the ocean current profile. In order to optimize sawtooth paths for an AUV, an MPC scheme with least squares cost function was presented in [62]. Interesting results including estimated ocean wave

profiles into an NMPC scheme, with an emphasis on real-time execution, were presented in [63]. However, the effect of noise and disturbance were not theoretically considered, but instead were presented through simulation testing. An MPC scheme with reduced dynamic model was presented in [64], where in order to avoid computational complexity, simplified linear models were considered for the vertical and horizontal control of the vehicle. In the aforementioned studies, the validation of the proposed strategies was conducted via simulation tests. An experimental validation of a visually aided NMPC scheme for an underwater robotic system was presented in [65], where simple kinematic equations of the system were considered.

In that sense, motivated by the aforementioned considerations, in this chapter, we formulated in a generic way the problem of Autonomous Underwater Vehicle (AUV) motion operating in a partially known constrained environment including obstacles. Specifically, during the control design, we considered strictly significant limitations such as: robustness with respect to external disturbances, dynamic parameter uncertainties, thruster saturations, obstacles, workspace boundary, predefined upper bound of the vehicle velocity. The obstacle avoidance has been designed based on the system‘s sensing range. This has allowed the AUV to compute or update its path in real-time based on the detected obstacles as the AUV moves through the workspace. Moreover, the controller has been designed in a way that the vehicle exploits the ocean currents, which results in reduced energy consumption by the thrusters and consequently increases significantly the autonomy of the system.

Chapter 3:

In this chapter, we present a novel Model predictive Control strategy for underwater robotic vehicles operating in a constrained workspace including obstacles. The purpose of this control scheme is to guide the vehicle towards specific way points. Various constraints such as: obstacles, workspace boundaries, thruster saturation and predefined upper bound of the vehicle velocity (requirements for various underwater tasks such as seabed inspection, mosaicking etc.) are considered during the control design. Moreover, the proposed scheme incorporates the full dynamics of the vehicle in which the ocean currents are also involved. Hence, the control inputs are formulated accordingly, in order to find optimal thrusts required for minimizing the way point tracking error, so that the vehicle may exploit the ocean currents, when these are in favor of the way-point tracking mission, resulting in reduced energy consumption by the thrusters. The closed-loop system has analytically guaranteed stability and convergence properties. The performance of the proposed control strategy is experimentally verified using a 4 Degrees of Freedom (DoF) underwater robotic vehicle inside a constrained test tank with sparse static obstacles. The covered material is based on the following contributions [66, 67]:

- [C11][1] **Shahab Heshmati-alamdari**, George C. Karras, Panos Marantos, and Kostas J. Kyriakopoulos, "A Robust Model Predictive Control Approach for Underwater Robotic Vehicles Operating in a Constrained workspace", IEEE International Conference on Robotics and Automation (ICRA), 2018.

- [J3] **Shahab Heshmati-alamdari**, George C. Karras, Panos Marantos, and Kostas J. Kyriakopoulos, "A Robust Predictive Control Approach for Underwater Robotic Vehicles", IEEE Transactions on Control Systems Technology (Under revision)

1.2.2 Visual Servoing part:

The *Visual Servoing part* consists of Chapters 4, 5,6. During this part (i.e., for cases that the robot is close to the point of interest), we consider visual feedback (i.e., imaging sonar or usual camera) as an appropriate feedback for designing of efficient controllers. This is motivated by the delay and inaccuracy of the measurements provided by common underwater localization systems e.g., Doppler Velocity Log (DVL) or Ultra Short Baseline (USBL). Generally, over the last decades, visual servoing has gained a lot of research interest in motion control systems. In general, it employs the visual information of a camera as feedback to determine the required motion control signal. Visual Servo Control has been extensively used in the past for the autonomous operation of underwater robotic vehicles. Complex missions such as ship hull inspection, surveillance of underwater facilities (for example oil platforms) and handling of underwater equipment (for example control panels, valves) require detailed and continuous visual feedback which can be obtained from either monocular or stereo vision systems. Structurally, visual servoing can be classified into three main categories: (i) Position-Based Visual Servoing (PBVS), where the visual features extracted along with the help of the visual tracking algorithm are used for the estimation of the 3D pose of the camera with respect to the target frame (ii) Image-Based Visual Servoing (IBVS), where the error function that is going to be minimized is based on the position of the image features in the image plane between the current and desired images and (iii) 2-1/2 Visual Servoing, where the control error function is defined in part in the Cartesian and in part in the image space [68–71]. Regarding visual servo control in underwater robotics, some previous work for the pipe inspection task (for example oil platforms) were realized in [72] and [73]. In [74] visual servoing using Laser Vision System (LVS) combined with an on-line identification mechanism has been proposed where the visibility constraints are formulated as state constraints. Some stereo vision approaches can be found at [75], [76]. In [77] and [78]

[1]The notations C, J stands for conference and journal publications, respectively, enumerated as appeared in author's web page: http://www.shahabheshmati.com

docking of an underwater robot using visual servoing has been presented. Some applications of visual servoing for station keeping of autonomous underwater vehicles are given in [79], [80] and in [81].

It is well known that visual servoing is solely based on visual information extracted from the position of the features of interest on the camera image, a significant issue that reasonably raises concerns the satisfaction of certain hard visibility constraints, imposed by the fact that the features of interest should constantly lie within the camera field of view [82]. Although dealing with hard constraints is a rather challenging control task, various methods have been presented in the related literature. In particular, decoupled control approaches have been proposed in [83–85]. In these approaches, the camera motion is controlled in part by a position based scheme, while an image based visual servoing part is employed in order to meet the hard visibility constraints. In [86], the efficiency and the performance of the aforementioned approaches were compared with respect to a conventional IBVS scheme. In the same vein, a set of visual servo controllers are combined under various switching policies in [87–90], where the satisfaction of the visibility constraints is guaranteed by the appropriate selection of the switching conditions.

Path-planning for image based visual servoing [91] has also been employed to tackle the aforementioned problem. The key idea behind this approach is to develop feasible image feature trajectories that meet the specific field of view constraints. In particular, path planning visual servoing strategies invariant to changes in camera intrinsic parameters are proposed in [91, 92], employing the projective transformation. In [91], based on a task function that is invariant to the camera intrinsic parameters, an image feature path is defined such that the camera follows a straight line in the Cartesian space. However, the visibility constraints are guaranteed only via zooming; an issue that was resolved later in [92] by modifying the solution provided initially in [93]. On the other hand, path planning strategies based on artificial potential fields have also been adopted. In [94], the authors suggested a novel potential field for eye-in-hand image based visual servoing that extends their previous work [95]. In particular, path planning in 3D space and projection onto the image plane are adopted to calculate a safe camera motion that does not violate either the field of view constraints or the joint limits. Similarly, obstacle avoidance is achieved in [88] as well. Finally, a navigation function was adopted in [96] to steer the camera from an initial to a desired configuration, while maintaining the visibility of the visual features. Nevertheless, the applicability of the aforementioned approaches is restricted only to very simple scenarios.

Alternatively, other researchers have adopted optimization techniques that aim at finding an optimal path with respect to various metrics such as the distance from image boundary, the length of the path and the energy[97]. In particular, several approaches based on

convex optimization techniques [98–100] and the perceptual control manifold [101] have been presented. On the other hand, employing the Lagrange multipliers approach, the optimization algorithm proposed in [102] aims at finding optimal image feature paths with respect to visibility constraints as well as various robot motion constraints. Similarly, the optimization algorithm proposed in [103], which is based on a polynomial parameterization of the camera path, guarantees visibility constraints while the camera is following a straight line in the Euclidean space. The former was extended later in [104] and [99], where an optimal path is obtained that satisfies a set of predefined constraints, by exploiting a parameterization of the camera trajectory in 3D space. Furthermore, the control strategy presented in [105], calculates an optimal camera trajectory by extending the 3D curves algorithm initially proposed in [106].

Recently, nonlinear model predictive control [58] has been adopted in IBVS [107–114] to handle the visibility issue by formulating it as state inequality constraints. The Visual Predictive Control (VPC) was originally presented in [107], while a robustness analysis with respect to noises and disturbances on the image features detection, was included in [108]. Based on these results, a self-triggering IBVS scheme was proposed in [114] to reduce the required computational effort. Several other applications of VPC for navigation tasks and medical procedures have also been presented in [110–113, 115]. However, all aforementioned approaches are based on solving online a nonlinear constrained optimization problem. Thus, their applicability in fast robotic tasks is rather questionable, owing to the high processing requirements.

Motivated by the above considerations, in this part we thus have formulated a number of novel visual servoing control strategies (i.e., Image based and Position based) in order to reach the robot's end effector close to or stabilize to the point of interest. Within this framework, we have considered significant issues in visual servoing control such as: camera Field of View (FoV) constraint, robustness with respect to Camera Calibration uncertainties and the resolution of visual tracking algorithm. Specifically, we first designed a Position Based Visual Servoing (PBVS) scheme for case when the relative position between the robot and the point of interest is available online (e.g., by detection of a known marker on the object) based on Nonlinear Model Predictive Control (NMPC). Then we extended this work to an Image Based Visual Servoing (IBVS) scheme for case that the relative position is not available online. Both of the aforementioned schemes are combined with a self-triggering mechanism that decides when the Visual Tracking Algorithm needs to be triggered and new control signals must be calculated. Thus, the proposed schemes result in reduction of the CPU computational effort, energy consumption and increases the autonomy of the system. Furthermore, for case when neither the Camera Calibration Matrix of the supposed vision system is

available in priory, a model free IBVS control strategy is proposed which satisfies the visibility constraints.

Chapter 4:

This chapter presents a novel Vision-Based NMPC scheme for underwater robotic vehicles. In this scheme, the visual servoing controller is combined with a mechanism that decides when the Visual Tracking Algorithm (VTA) needs to be triggered and when new control inputs must be calculated. More specifically, in the proposed scheme, the control loop does not close periodically, but instead a self-triggering mechanism decides when to provide the next control update. Between two consecutive triggering instants, the control sequence computed by the NMPC is applied to the robot in an open-loop fashion, i.e, no visual measurements are required during that period. This results to a significant smaller number of requested measurements from the vision system, as well as less frequent computations of the control law. This way, the processing effort and energy consumption are reduced and the accuracy and autonomy of the robotic system are increased. These are of utmost importance in the case of Autonomous Underwater Vehicles (AUVs) which perform persistent inspection tasks. The inputs and visibility constraints (i.e preserving the target inside the camera's field of view), the external disturbances induced by currents and waves, as well as the vehicle's kinematic constraints due to under-actuation, are being considered during the control design. The closed-loop system has analytically guaranteed stability and convergence properties. The performance of the proposed control scheme is experimentally verified and compared with respect to the (conventional) classic NMPC scheme, using a small underactuated underwater vehicle. The covered material is based on the following contributions [65, 116]:

- [C1] Alina Eqtami,**Shahab Heshmati-alamdari**, Dimos V Dimarogonas, Kostas J Kyriakopoulos, "Self-triggered model predictive control for nonholonomic systems", European Control Conference (ECC), 2013.

- [C3] **Shahab Heshmati-alamdari**, Alina Eqtami, George C Karras, Dimos V Dimarogonas, Kostas J Kyriakopoulos, "A self-triggered visual servoing model predictive control scheme for under-actuated underwater robotic vehicles", IEEE International Conference on Robotics and Automation (ICRA), 2014.

Chapter 5:

This chapter presents a novel Image-Based NMPC scheme for autonomous underwater robotic vehicles. Generally, a real-time visual servoing task which employs a Visual Tracking Algorithm (VTA) imposes high computational cost to robotic system, which

consequently results in higher energy consumption and lower autonomy. Motivated by this fact, this chapter presents a novel Image Based Visual Servoing-Model Predictive Control (IBVS-MPC) scheme which is combined with a mechanism that decides when the VTA needs to be triggered and new control inputs must be calculated. Between two consecutive triggering instants, the control input trajectory is applied to the robot in an openloop fashion, i.e, no visual measurements and calculation of the control inputs are required during that period. This results in the reduction of the computational effort, energy consumption and increases the autonomy of the system. These factors are of utmost importance in the case of small autonomous robotic systems which perform vision based tasks, such as surveillance and inspection of indoors and outdoors environments. The visibility and inputs constraints, optimality rate of the MPC, as well as the external disturbances, are being considered during the control design. The covered material is based on the following contributions [108, 117]:

- [C4] **Shahab Heshmati-alamdari**, George K Karavas, Alina Eqtami, Michael Drossakis, Kostas J Kyriakopoulos, "Robustness analysis of model predictive control for constrained image-based visual servoing", IEEE International Conference on Robotics and Automation (ICRA), 2014.

- [C7] **Shahab Heshmati-alamdari**, George C. Karras, Alina Eqtami and Kostas J. Kyriakopoulos,"A Robust Self Triggered Image Based Visual Servoing Model Predictive Control Scheme for Small Autonomous Robots", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015.

Chapter 6:

In this chapter, we propose a novel Image Based Visual Servoing scheme that imposes prescribed transient and steady state response on the image feature coordinate errors and satisfies the visibility constraints that inherently arise owing to the camera's limited field of view, despite the inevitable calibration and depth measurement errors. Visualizing the aforementioned performance specifications as error bounds, the key idea is to provide an error transformation that converts the original constrained problem into an equivalent unconstrained one, the stabilization of which proves sufficient to achieve prescribed performance guarantees and satisfy the visibility constraints. The performance of the developed scheme is a priori and explicitly imposed by certain designer specified performance functions, and is fully decoupled by the control gains selection, thus simplifying the control design. Moreover, its computational complexity proves significantly low. It is actually a static scheme involving very few and simple calculations to output the control signal, which enables easily its implementation on fast embedded control platforms. The covered material is based on the following contributions[118, 119]:

- [C6] **Shahab Heshmati-alamdari**, Charalampos P Bechlioulis, Minas V Liarokapis, Kostas J Kyriakopoulos, "Prescribed performance image based visual servoing under field of view constraints", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), 2014.

- [J5] Charalampos P Bechlioulis, **Shahab Heshmati-alamdari**, George C. Karras, Kostas J. Kyriakopoulos"Robust Image Based Visual Servoing with Prescribed Performance under Field of View Constraints", IEEE Transactions on Robotics, (Under revision).

1.2.3 Interaction and Cooperative Transportation part:

The *Interaction and Transportation part* consists of Chapters 7, 8, 9.

Interaction:
It is well known that underwater tasks are very challenging owing mainly to external disturbances (i.e., sea currents), the lack of appropriate and adequately accurate sensing/localization [120] and the unknown (or partially known) constrained environment (e.g., offshore industry, oil/gas facilities) [121, 122]. The aforementioned difficulties make the control of underwater manipulator systems a challenging problem that has already gained significant scientific attention within the marine robotic community during the last years [122]. Such control problems involve constrained high-dimensional nonlinear systems with significant complexity regarding the uncertainty of the robot dynamics, the redundancy of the system, the various operational constraints (e.g., visibility constraints and joint limits), the nonlinear coupled dynamics between the underwater vehicle and manipulator systems [123] as well as the gravity/buoyancy forces that affect the response of the manipulator. All aforementioned challenges should be taken into consideration when designing the control system of a UVMS. Hence, from a control perspective, achieving such goals highlights the need of employing appropriate methodologies from nonlinear robot control theory [124]. However, most control schemes that have been developed so far for static manipulators and space robots cannot be used directly on UVMS owing to the aforementioned specifics. Therefore, the control of UVMS remains still a challenging task.

Until the early 1990s, only a few research studies had appeared that dealt with the UVMS control [125]. Even deriving the dynamic model and developing an efficient dynamic simulation for underwater vehicles equipped with robotic manipulators were studied recently in [126–128]. However, during the last years, owing to the enormous demand for more dexterous and efficient underwater interaction robotic platforms, many research studies

have been conducted towards enhancing the efficacy of UVMSs [12]. In [129], a hybrid position/force control scheme was designed and implemented on a hydraulically actuated manipulator. That work was extended later into a hybrid position/force controller and was compared to a fixed-gain version for various contact stiffness models [130]. However, the dynamic coupling between the manipulator and the vehicle was not considered. In [131], a control scheme was proposed in order to achieve high maneuverability of an underwater manipulator mounted on a ROV. A force/torque sensor was assumed to be installed between the ROV and the manipulator to compensate the dynamic effects of the underwater manipulator motion on the vehicle. The control strategy proposed in [132] compensates for the force/moment exerted on the vehicle from the manipulator by utilizing the buoyant and gravitational forces, while simultaneously maintaining the contact between the robot end-effector and the environment. In the same direction, the force control strategy proposed in [133] aims at stabilizing the vehicle when the manipulator operates in a constrained workspace. Moreover, an impedance control technique was studied in [134], where the UVMS dynamic model was considered as an integrated system. That work was extended later in [135], where impedance control was combined with hybrid position/force control by means of fuzzy switching. Nevertheless, all aforementioned control strategies request accurate knowledge of the contact stiffness model.

On the other hand, treating a UVMS as an integrated system raises significant issues regarding the redundancy of the system, the uncertainties and the dynamic couplings, which all should be considered during the control design. It is well known that a UVMS is kinematically redundant owing to the vehicle's degrees of freedom [136]. The redundancy of a UVMS might be exploited in order to coordinate the system in such a way that the end-effector tracking accuracy is guaranteed, and this may be achieved by addressing various issues such as: i) the sluggish response of vehicle's thrusters [137–139], ii) the manipulator's joint limits, iii) the manipulability and iv) the configuration singularities [136]. In general, two strategies at the kinematic [140] and the dynamic [141] levels respectively have been developed for handling the robot's redundancy. The former was enhanced later in [142] for an arbitrary number of priority levels with inferior computational complexities. That framework, which is based on the least-squares minimization technique, solves a finite number of prioritized secondary tasks (e.g., maintaining manipulator's joint limits, increasing the manipulability) in a hierarchical structure. However, a compromise raises among the tasks corresponding to the inequality objectives (e.g., joint limits) [143]. In order to tackle this issue, a method was proposed based on quadratic programming in [144], which solves equality and inequality constraints at any priority level. However, that approach leads in high dimensional quadratic programming problems and cannot handle the activation or deactivation of tasks [145]. The aforementioned strategy was extended later in [146], where instead of a cascade of quadratic programming

problems, a single problem is solved that identifies the active set of all the constraints at the same time. Nevertheless, that strategy lacks the ability of enabling and disabling inequality tasks without causing discontinuities. Recently, a generic method that allows activating and deactivating tasks without incurring discontinuities was proposed in [145]. For the case of a UVMS, based on results initially proposed in [147], the redundancy resolution has been employed firstly in [136] to avoid the manipulator's singular configurations. Later, a classical gradient projection method [140] was adopted that addressed various practical criteria such as minimization of load [148, 149] and restoring moments [150]. Finally, a method based on a fuzzy switching technique was proposed in [151] to overcome the conflict among various secondary tasks.

As it is already mentioned, in underwater robotic interaction tasks various issues regarding the uncertainties and complexity of the robot dynamic model, the external disturbances (e.g., sea currents), the steady state performance as well as the overshooting/undershooting of the interaction force error, should be addressed during the control design. Motivated by the aforementioned considerations, this part presents a robust interaction control scheme for a UVMS in contact with the environment, with great applications in underwater robotics (e.g. sampling of the sea organisms, underwater welding, object handling). The proposed control scheme does not required any a priori knowledge of the UVMS dynamical parameters or the stiffness model. It guarantees a predefined behavior in terms of desired overshoot, transient and steady state response and it is robust with respect to external disturbances and measurement noises. Moreover, the proposed controller exhibits the following important characteristics: i) it is of low complexity and thus can be easily used in most UVMSs ii) the performance of the proposed scheme (e.g., desired overshoot, steady state response) is a priori and explicitly imposed by certain designer-specified performance functions, and is fully decoupled by the control gains selection, thus simplifying the control design.

Chapter 7:

This chapter presents a force/position tracking control protocol for an Underwater Vehicle Manipulator System (UVMS) in compliant contact with a planar surface, without incorporating any knowledge of the UVMS dynamic model, the exogenous disturbances or the contact stiffness model. Moreover, the proposed control framework guarantees: (i) certain predefined minimum speed of response, maximum steady state error as well as overshoot/undershoot concerning the force/position tracking errors, (ii) contact maintenance and (iii) bounded closed loop signals. Additionally, the achieved transient and steady state performance is solely determined by certain designer-specified performance

functions/parameters and is fully decoupled from the control gain selection and the initial conditions. Simulation and experimental studies clarify the proposed method and verify its efficiency. These results presented in this chapter are based on [152, 153]:

- [C10] **Shahab Heshmati-alamdari**, Alexandros Nikou, Kostas J. Kyriakopoulos, Dimos V. Dimarogonas, "A Robust Force Control Approach for Underwater Vehicle Manipulator Systems", The 20th World Congress of the International Federation of Automatic Control (IFAC 2017), 2017.

- [J4] **Shahab Heshmati-alamdari**, Charalampos P. Bechlioulis , George C. Karras, Alexandros Nikou, Dimos V Dimarogonas, Kostas J. Kyriakopoulos"A Robust Interaction Control Approach for Underwater Vehicle Manipulator Systems", IFAC Annual Reviews in Control" Journal, 2018.

Cooperative Transportation:

During the last decades, Unmanned Underwater Vehicles (UUVs) have been widely used in various applications such as marine science (e.g., biology, oceanography, archeology) and offshore industry (e.g., ship maintenance, inspection of oil/gas facilities) [154]. In particular, a vast number of the aforementioned applications, demand the underwater vehicle to be enhanced with intervention capabilities as well [8], thus raising increasing interest on Underwater Vehicle Manipulator System (UVMS)[11]. Nowadays, the underwater intervention tasks involve a Remotely Operated Vehicle (ROV), equipped with one or multiple manipulators that allow it to grasp, transport and manipulate objects while being controlled by a human pilot on a surface ship, via a master-slave tele-operation scheme [12–14]. However, the well-known disadvantages of human-robot tele-operation (e.g., time delays, increase of human fatigue over time) led inevitably to the development of autonomous intervention control schemes for UVMSs that have gained significant scientific attention during the last years [10, 15, 16].

Most of the underwater manipulation tasks can be carried out more efficiently, if multiple UVMSs are cooperatively involved. For instance, two or more UVMSs can transport bulky objects (see Fig. 1.1) more easily and safely than a single UVMS, owing to shape, actuation and payload constraints [2, 155]. In [155], the authors have studied the problem of modeling two UVMSs carrying a rigid object. The robot-object contact was considered rigid, thus the whole system configuration formed a singular system of differential equations [156]. The kinematic redundancy and manipulability of this system were examined in [157, 158]. Moreover, a centralized cooperative control scheme for multiple UVMSs holding commonly an object was proposed in [2]. However, in all

aforementioned works, the major requirements and constraints imposed by the nature of underwater environment have not been considered at all.

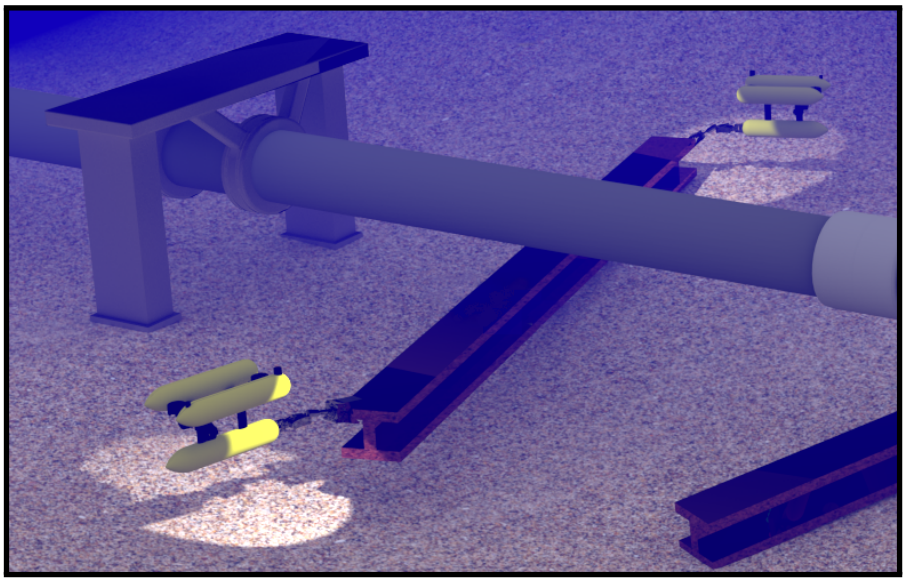


FIGURE 1.1: A cooperative transportation task conducted by UVMSs.

Underwater tasks are very demanding, with the most significant challenge being imposed by the strict communication constraints [15, 159]. In general, the communication of multi-robot systems can be classified in two major categories, namely explicit and implicit. The first one is designed solely to convey information such as control signals or sensory data directly to other robots [160], while the latter occurs as a side-effect of robot's interaction with the environment or other robots, either physically (e.g., the interaction forces between the object and the robot) or non-physically (e.g., visual observation). In the latter case, the required information is acquired by appropriate sensors installed on the robots (e.g., force/torque or vision sensors). The most investigated and frequently employed communication form in multi-robot systems is the explicit one. It usually leads to simpler theoretic analysis and renders teams more effective. However, even though the inter-robot communication is of utmost importance during cooperative manipulation tasks, employing explicit communication in underwater environment may result in severe performance problems owing to the limited bandwidth and update rate of underwater acoustic devices. Moreover, as the number of cooperating robots increases, communication protocols require complex design to deal with the crowed bandwidth [161]. Therefore, the number of operating underwater robots, involved in cooperative schemes that exploit explicit communication protocols, is strictly limited owing to the narrow bandwidth of acoustic communication devices. In order to overcome such limitations, implicit communication can be employed instead. Despite the increased difficulty of the theoretical analysis, it leads to simpler protocols and saves bandwidth as well as power, since no or very few data is explicitly exchanged.

Cooperative manipulation has been well-studied in the literature, especially the centralized schemes [162–165]. Despite its efficiency, centralized control is less robust, since all units rely on a central system, and its complexity increases rapidly as the number of participating robots becomes large. On the other hand, although decentralized cooperative manipulation schemes exhibit increased robustness and low complexity, they usually depend on either explicit communication interchange among the robots (e.g., online transmission of the desired trajectory [166, 167] or off–line knowledge of the objects' trajectory [168–170]). For instance, in order to achieve collision avoidance, either the desired object trajectory should be transmitted online between the cooperating underwater robots or all robots should agree mutually on a safe desired trajectory of the object in the workspace. This demands an accurate common global localization system for all participating robots [171], which either is difficult to be achieved in underwater environment or in the most optimistic case would raise the mission cost. Therefore, the design of decentralized cooperative manipulation algorithms for underwater tasks employing implicit and lean explicit communication becomes apparent. In recent studies [171, 172], potential fields methods were employed and a multi layer control structure was developed to manage the coordination of the robot swarm, the guidance and navigation of UVMSs and the manipulation tasks. To overcome localization and consensus problems, the authors have considered the object as the swarm reference frame. However, employing this strategy, requires each robot to communicate with the whole robot team, which consequently restricts the number of robots involved in the cooperative manipulation task owing to bandwidth limitations. Compelling results towards the same direction have been given in [173–175], based on priority control strategy [145]. In particular, a three-fold decentralized cooperative control strategy is proposed where initially, each robot individually finds out an optimal task space control velocity, which is transfered afterwards among the robots in order to obtain a commonly agreed velocity via a fusion policy. The commonly agreed task velocity then is extended to the joint space of each UVMS based again on a task priority technique [145], but this time with a higher priority. Various safety constraints (e.g., joint limits, manipulability) may also be considered in case of two cooperating UVMSs. However, the implementation of the aforementioned scheme for a large number of cooperative UVMSs, should also deal with crowed bandwidth issues. Moreover, if constrained workspace (i.e., obstacles within the workplace) is considered, achieving consensus on a mutually agreed safe trajectory would become questionable.

Motivated by the above considerations, in this part we have addressed the problem of cooperative object transportation for a team of UVMSs in a constrained workspace involving static obstacles. First, for case when the robots are equipped with appropriate force/torque sensors at its end effector we have proposed a decentralized impedance

control scheme with the coordination relying solely on implicit communication arising from the physical interaction of the robots with the commonly grasped object. Second, for case when the robots are not equipped with force/torque sensor at it end effector, we have proposed a decentralized predictive control approach which takes into account constraints that emanate from control input saturation as well kinematic and representation singularities.

Chapter 8:

This chapter addresses the problem of cooperative object transportation for multiple Underwater Vehicle Manipulator Systems (UVMSs) in a constrained workspace involving static obstacles, with the coordination relying solely on implicit communication arising from the physical interaction of the robots with the commonly grasped object. We propose a novel distributed leader-follower architecture, where the leading UVMS, which has knowledge of the object's desired trajectory, tries to achieve the desired tracking behavior via an impedance control law, navigating in this way, the overall formation towards the goal configuration while avoiding collisions with the obstacles. On the other hand, the following UVMSs estimate locally the object's desired trajectory via a novel prescribed performance estimation law and implement a similar impedance control law achieving in this way tracking of the desired trajectory despite the uncertainty and external disturbance in the object and the UVMS dynamics respectively. The feedback relies on each UVMS's force/torque measurements and no explicit data is exchanged online among the robots, thus reducing the required communication bandwidth and increasing robustness. Moreover, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities. These results presented in this chapter are based on [176, 177]:

- [C13] **Shahab Heshmati-alamdari**, Charalampos P. Bechlioulis, George C. Karras, Kostas J. Kyriakopoulos, "Decentralized Impedance Control for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems under Lean Communication", IEEE OES Autonomous Underwater Vehicle Symposium, 2018, Accepted.

- [J2] **Shahab Heshmati-alamdari**, Charalampos P. Bechlioulis, George C. Karras, Kostas J. Kyriakopoulos, "Cooperative Impedance Control for Multiple Underwater Vehicle Manipulator Systems under Lean Communication", IEEE Journal of Oceanic Engineering, submitted.

Chapter 9:

This chapter addresses the problem of cooperative object transportation for multiple Underwater Vehicle Manipulator Systems (UVMSs) in a constrained workspace involving static obstacles. We propose a decentralized Nonlinear Model Predictive Control (NMPC) approach for a team of UVMSs in order to transport an object while avoiding significant constraints and limitations such as: kinematic and representation singularities, obstacles within the workspace, joint limits and control input saturations. More precisely, by exploiting the coupled dynamics between the robots and the object, and using certain load sharing coefficients, we design a decentralized NMPC for each UVMS in order to cooperatively transport the object within the workspace's feasible region. Moreover, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities. Additionally, the feedback relies on each UVMS's locally measurements and no explicit data is exchanged online among the robots, thus reducing the required communication bandwidth and increasing robustness. These results presented in this chapter are based on [178, 179]:

- [C14] **Shahab Heshmati-alamdari**, George C. Karras, Kostas J. Kyriakopoulos, "A Decentralized Predictive Control Approach for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems", IEEE International Conference on Robotics and Automation (ICRA), 2019, Submitted.

- [J6] **Shahab Heshmati-alamdari**, George C. Karras, Kostas J. Kyriakopoulos"A Robust Decentralized Predictive Control Approach for Cooperative Transportation by Multiple Underwater Vehicle Manipulator Systems", IEEE Journal of Oceanic Engineering, (Under preparation).

Part II - Mathematical Modeling

Chapter 2

Underwater Vehicle Mathematical Modeling

2.1 Underwater Vehicle Kinematics & Dynamics

Let us define a body-fixed frame $\mathcal{B} = \{e_x, e_y, e_z\}$ attached to the vehicle's center of gravity, as shown in 2.1, and an inertial frame $\mathcal{I} = \{e_F, e_R, e_D\}$ located at a fix position $O_{\mathcal{I}}$ within the workspace of the vehicle. Following standard modeling techniques [154], the dynamic model of the vehicle in the body fixed frame may be derived from the general Newton-Euler motion equations of a 6-DoF rigid body subject to external forces and torques in a fluid medium, as follows:

$$\begin{aligned} \boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) &= \boldsymbol{\tau}_E + \boldsymbol{\tau} \\ \dot{\boldsymbol{\eta}} &= \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{v} \end{aligned} \tag{2.1}$$

where:

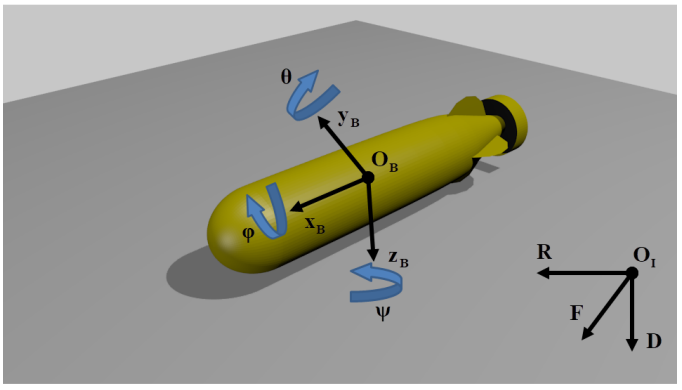


FIGURE 2.1: Underwater Vehicle Frames

- $\boldsymbol{\eta} \triangleq \left[\boldsymbol{\eta_1^T}\ \boldsymbol{\eta_2^T}\right]^T \in \Re^6$ is the pose vector expressed in $\mathcal{I}$, that involves the position (i.e., $\boldsymbol{\eta_1} \triangleq [x\ y\ z]^T$) and orientation (i.e., $\boldsymbol{\eta_2} \triangleq [\phi\ \theta\ \psi]^T$) vectors;
- $\boldsymbol{v} \triangleq \left[\boldsymbol{v_1^T}\ \boldsymbol{v_2^T}\right]^T \in \Re^6$ is the velocity vector expressed in $\mathcal{B}$, that involves the linear (i.e., $\boldsymbol{v_1} \triangleq [u\ v\ w]^T$) and angular (i.e., $\boldsymbol{v_2} \triangleq [p\ q\ r]^T$) velocity vectors;
- $\boldsymbol{\tau}_E \in \Re^6$ is the total environmental force/torque vector expressed in $\mathcal{B}$, that is applied on the vehicle;
- $\boldsymbol{\tau} \triangleq [\tau_X,\ \tau_Y,\ \tau_Z,\ \tau_K,\ \tau_M,\ \tau_N]^T \in \Re^6$ is the total propulsion vector (i.e., the body forces $\tau_X,\ \tau_Y,\ \tau_Z$ and torques $\tau_K,\ \tau_M,\ \tau_N$ generated by the actuators) applied on the vehicle and expressed in $\mathcal{B}$;
- $\boldsymbol{M} \triangleq \boldsymbol{M}_{RB} + \boldsymbol{M}_A$, where $\boldsymbol{M}_{RB} \in \Re^{6\times6}$ and $\boldsymbol{M}_A \in \Re^{6\times6}$ are the rigid body and added mass inertia matrices respectively;
- $\boldsymbol{C}(\boldsymbol{v}) \triangleq \boldsymbol{C}_{RB}(\boldsymbol{v}) + \boldsymbol{C}_A(\boldsymbol{v})$, where $\boldsymbol{C}_{RB}(\boldsymbol{v}) \in \Re^{6\times6}$ and $\boldsymbol{C}_A(\boldsymbol{v}) \in \Re^{6\times6}$ are the rigid body and added mass matrices that model the Coriolis and centrifugal effects respectively;
- $\boldsymbol{D}(\boldsymbol{v}) \triangleq \boldsymbol{D}_{quad}(\boldsymbol{v}) + \boldsymbol{D}_{lin}$, where $\boldsymbol{D}_{quad}(\boldsymbol{v}) \in \Re^{6\times6}$ and $\boldsymbol{D}_{lin} \in \Re^{6\times6}$ denote the quadratic and linear drag matrices respectively;
- $\boldsymbol{g}(\boldsymbol{\eta}) \in \Re^6$ is the hydrostatic restoring force vector;
- $\boldsymbol{J}(\boldsymbol{\eta}) \triangleq \begin{bmatrix} \boldsymbol{J}_1(\boldsymbol{\eta_2}) & \mathbf{O}_3 \\ \mathbf{O}_3 & \boldsymbol{J}_2(\boldsymbol{\eta_2}) \end{bmatrix}$ is the Jacobian matrix transforming the velocities from the body-fixed frame $\mathcal{B}$ to the inertial frame $\mathcal{I}$, in which $\boldsymbol{J}_1(\boldsymbol{\eta_2}) \in SO(3)$ stands for the rotation matrix and $\boldsymbol{J}_2(\boldsymbol{\eta_2}) \in \Re^{3\times3}$ denotes the lumped transformation matrix;

The expanded form of the matrices appear in Eq.2.1, for a generic $6-DoF$ underwater vehicle, are given as follows [154]:

- $\boldsymbol{M_{RB}} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & I_x & -I_{xy} & -I_{xz} \\ mz_G & 0 & -mx_G & -I_{yx} & I_y & -I_{yz} \\ -my_G & mx_G & 0 & -I_{zx} & -I_{zy} & I_z \end{bmatrix}$

- $\boldsymbol{M_A} = -\begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix}$

- $\boldsymbol{C_{RB}} = \begin{bmatrix} 0 & 0 & 0 & m(y_G q + z_G r) & -m(x_G q - w) & -m(x_G r + v) \\ 0 & 0 & 0 & -m(y_G p + w) & m(z_G r + x_G p) & -m(y_G r - u) \\ 0 & 0 & 0 & -m(z_G p - v) & -m(z_G q + u) & m(x_G p + y_G q) \\ -m(y_G q + z_G r) & m(y_G p + w) & m(z_G p - v) & 0 & -I_{yz}q - I_{xz}p + I_z r & I_{yz}r + I_{xy}p - I_y q \\ m(x_G q - w) & -m(z_G r + x_G p) & m(z_G q + u) & I_{yz}q + I_{xz}p - I_z r & 0 & -I_{xz}r - I_{xy}q + I_x p \\ m(x_G r + v) & m(y_G r - u) & -m(x_G p + y_G q) & -I_{yz}r - I_{xy}p + I_y q & I_{xz}r + I_{xy}q - I_x p & 0 \end{bmatrix}$

- $\boldsymbol{C_A(v)} = \begin{bmatrix} 0 & 0 & 0 & 0 & -\alpha_3 & \alpha_2 \\ 0 & 0 & 0 & \alpha_3 & 0 & -\alpha_1 \\ 0 & 0 & 0 & -\alpha_2 & \alpha_1 & 0 \\ 0 & -\alpha_3 & \alpha_2 & 0 & -b_3 & b_2 \\ \alpha_3 & 0 & -\alpha_1 & b_3 & 0 & -b_1 \\ -\alpha_2 & \alpha_1 & 0 & -b_2 & b_1 & 0 \end{bmatrix}$, where:

$$\alpha_1 = X_{\dot{u}}u + X_{\dot{v}}v + X_{\dot{w}}w + X_{\dot{p}}p + X_{\dot{q}}q + X_{\dot{r}}r$$
$$\alpha_2 = X_{\dot{v}}u + Y_{\dot{v}}v + Y_{\dot{w}}w + Y_{\dot{p}}p + Y_{\dot{q}}q + Y_{\dot{r}}r$$
$$\alpha_3 = X_{\dot{w}}u + Y_{\dot{w}}v + Z_{\dot{w}}w + Z_{\dot{p}}p + Z_{\dot{q}}q + Z_{\dot{r}}r$$
$$b_1 = X_{\dot{p}}u + Y_{\dot{p}}v + Z_{\dot{p}}w + K_{\dot{p}}p + K_{\dot{q}}q + K_{\dot{r}}r$$
$$b_2 = X_{\dot{q}}u + Y_{\dot{q}}v + Z_{\dot{q}}w + K_{\dot{q}}p + M_{\dot{q}}q + M_{\dot{r}}r$$
$$b_2 = X_{\dot{r}}u + Y_{\dot{r}}v + Z_{\dot{r}}w + K_{\dot{r}}p + M_{\dot{r}}q + N_{\dot{r}}r$$

- $\boldsymbol{D}(\boldsymbol{v}) \triangleq \boldsymbol{D}_{lin} + \boldsymbol{D}_{quad}(\boldsymbol{v})$, where:

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

- $\boldsymbol{g}(\boldsymbol{\eta}) = -\begin{bmatrix} \boldsymbol{J}_1^T(\boldsymbol{\eta_2})(m\boldsymbol{g}_{\mathcal{I}} + \boldsymbol{b}_{\mathcal{I}}) \\ \boldsymbol{r}_G \times \boldsymbol{J}_1^T(\boldsymbol{\eta_2}) m\boldsymbol{g}_{\mathcal{I}} + \boldsymbol{r}_B \times \boldsymbol{J}_1^T(\boldsymbol{\eta_2})\boldsymbol{b}_{\mathcal{I}} \end{bmatrix}$

- $\boldsymbol{J_1}(\boldsymbol{\eta_2}) = \begin{bmatrix} c\psi c\theta & -s\psi c\varphi + c\psi s\theta s\varphi & s\psi s\varphi + c\psi c\varphi s\theta \\ s\psi c\theta & c\psi c\varphi + s\varphi s\theta s\psi & -c\psi s\varphi + s\theta s\psi c\varphi \\ -s\theta & c\theta s\varphi & c\theta c\varphi \end{bmatrix}$

- $\boldsymbol{J_2}(\boldsymbol{\eta_2}) = \begin{bmatrix} 1 & s\varphi t\theta & c\varphi t\theta \\ 0 & c\varphi & -s\varphi \\ 0 & s\varphi/c\theta & c\varphi/c\theta \end{bmatrix}$

where:

- m is the mass, $\boldsymbol{r}_G \triangleq [x_G, y_G, z_G]^T$ and $\boldsymbol{r}_B \triangleq [x_B, y_B, z_B]^T$ denote the position of the gravity and buoyancy center with respect to the body frame $\mathcal{B}$.
- I_x, I_y, I_z, I_{xy},I_{yx},I_{xz}, I_{zx}, I_{yz}, I_{zy} denote the moments of inertia along the respective axes.
- $(x)_{\dot{y}}$, for $x \in \{X, Y, Z, K, M, N\}$ and $y \in \{u, v, w, p, q, r\}$ are the added mass terms according to [154]. Added mass should be understood as pressure-induced forces and moments due to a forced harmonic motion of the body which are proportional to the acceleration of the body. Consequently, the added mass forces and the acceleration will be 180 degrees out of phase to the forced harmonic motion [154]. More specific it stands: $(x)_{\dot{y}} \triangleq \frac{\partial(x)}{\partial \dot{y}}$.
- X_u, $X_{u|u|}$, Y_v, $Y_{v|v|}$, Z_w, $Z_{w|w|}$, K_p, $K_{p|p|}$, M_q, $M_{q|q|}$, N_r, $N_{r|r|}$ are the 1^{st} and 2^{nd} order drag parameters according to [154].
- $\boldsymbol{g}_{\mathcal{I}}$ and $\boldsymbol{b}_{\mathcal{I}}$ are the gravitational acceleration and buoyancy vectors expressed in $\mathcal{I}$.
- $s\,\cdot = \sin(\cdot)$, $c\,\cdot = \cos(\cdot)$, $t\,\cdot = \tan(\cdot)$

Remark 2.1. Let us define a constant rotation matrix $\boldsymbol{R_{IN}}$ transforming any vector from the standard North-East-Down (NED) frame $\mathcal{N}$ to the inertial frame $\mathcal{I}$. Then, $\boldsymbol{g}_{\mathcal{I}} = \boldsymbol{R_{IN}}\boldsymbol{g}_{\mathcal{N}} = \boldsymbol{R_{IN}}[0, 0, g]^T$ and $\boldsymbol{b}_{\mathcal{I}} = \boldsymbol{R_{IN}}[0, 0, -B]^T$, where g is the gravitational acceleration and B the buoyancy force.

Remark 2.2. In most cases, the underwater vehicles have three planes of symmetry. In addition to that, the z-axis of the inertial frame $\mathcal{I}$ is selected to be aligned with the z-axis of the NED frame $\mathcal{N}$. Thus, the representation of matrices $\boldsymbol{M_A}$, $\boldsymbol{C_A}(\boldsymbol{v})$ and $\boldsymbol{g}(\boldsymbol{\eta})$ can be further simplified:

- $\boldsymbol{M_A} = diag\{X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}\}$
- $\boldsymbol{C_A}(\boldsymbol{v}) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix}$

- $\boldsymbol{g}(\boldsymbol{\eta}) = \begin{bmatrix} (mg-B)\,s\theta \\ -(mg-B)\,c\theta s\varphi \\ -(mg-B)\,c\theta c\varphi \\ -(y_G mg - y_B B)\,c\theta c\varphi + (z_G mg - z_B B)\,c\theta s\varphi \\ (z_G mg - z_B B)\,s\theta + (x_G mg - x_B B)\,c\theta c\varphi \\ -(x_G mg - x_B B)\,c\theta s\varphi - (y_G mg - y_B B)\,s\theta \end{bmatrix}$

Next we describe the calculation of the total propulsion vector $\boldsymbol{\tau}$ for thruster actuated underwater vehicles. Consider a $6-DoF$ actuated vehicle with $i = 1...N$ thrusters. Each thruster produces the following vector of forces and torques with respect to the body fixed frame $\mathcal{B}$:

$$ {}^{\mathbf{i}}\boldsymbol{\tau} = \begin{bmatrix} {}^{\mathbf{i}}\mathbf{T} \\ {}^{\mathbf{i}}\mathbf{Q} \end{bmatrix} = \begin{bmatrix} {}^{i}T\,{}^{\mathbf{i}}\mathbf{e} \\ {}^{i}T\left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right) \end{bmatrix} = \begin{bmatrix} {}^{i}e_x \\ {}^{i}e_y \\ {}^{i}e_z \\ \left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right)_x \\ \left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right)_y \\ \left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right)_z \end{bmatrix} {}^{i}T \tag{2.2} $$

where ${}^{\mathbf{i}}\mathbf{e} = \begin{bmatrix} {}^{i}e_x & {}^{i}e_y & {}^{i}e_z \end{bmatrix}^T = \begin{bmatrix} \cos\psi_T^i \cos\theta_T^i \\ \sin\psi_T^i \cos\theta_T^i \\ -\sin\theta_T^i \end{bmatrix}$ is the orientation of the thruster i relatively to the body fixed frame $\mathcal{B}$ and ${}^{\mathbf{i}}\mathbf{L} = \begin{bmatrix} l_i^x & l_i^y & l_i^z \end{bmatrix}^T$ determines the position of the point of attack for ${}^{\mathbf{i}}\mathbf{T}$ relatively to $\mathcal{B}$. Superposition of the individual contributions ${}^{\mathbf{i}}\boldsymbol{\tau}$ leads to total vector of propulsion forces and torques:

$$ \boldsymbol{\tau} = \begin{bmatrix} X \\ Y \\ Z \\ K \\ M \\ N \end{bmatrix} = \sum_i^N {}^{\mathbf{i}}\tau = \sum_i^N \begin{bmatrix} {}^{\mathbf{i}}\mathbf{e} \\ \left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right) \end{bmatrix} {}^{i}T = \boldsymbol{B}_N \boldsymbol{T}_N = \begin{bmatrix} {}^{1}e_x & {}^{i}e_x & {}^{N}e_x \\ {}^{1}e_y & {}^{i}e_y & {}^{N}e_y \\ {}^{1}e_z & {}^{i}e_z & {}^{N}e_z \\ \left({}^{1}\mathbf{L} \times {}^{1}\mathbf{e}\right)_x & \cdots \left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right)_x \cdots & \left({}^{N}\mathbf{L} \times {}^{N}\mathbf{e}\right)_x \\ \left({}^{1}\mathbf{L} \times {}^{1}\mathbf{e}\right)_y & \left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right)_y & \left({}^{N}\mathbf{L} \times {}^{N}\mathbf{e}\right)_y \\ \left({}^{1}\mathbf{L} \times {}^{1}\mathbf{e}\right)_z & \left({}^{\mathbf{i}}\mathbf{L} \times {}^{\mathbf{i}}\mathbf{e}\right)_z & \left({}^{N}\mathbf{L} \times {}^{N}\mathbf{e}\right)_z \end{bmatrix} \begin{bmatrix} {}^{1}T \\ \cdot \\ \cdot \\ \cdot \\ {}^{i}T \\ \cdot \\ \cdot \\ \cdot \\ {}^{N}T \end{bmatrix} \tag{2.3} $$

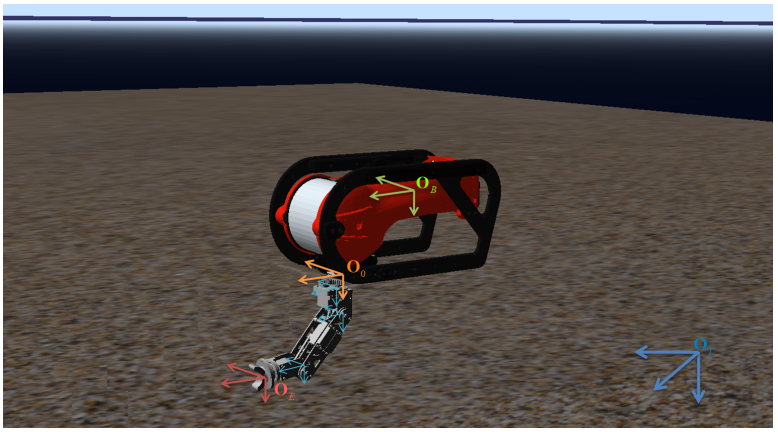


FIGURE 2.2: Underwater Vehicle Manipulator System Frames

2.2 Underwater Vehicle Manipulator System Kinematics

In order to analyze the motion of UVMS in 3D-space, several reference frames need to be defined. Consider an underwater robotic vehicle equipped with a robotic manipulator with n Degree of Freedom (DoF) (See Fig. 2.1). Let us define a body-fixed frame $\mathcal{B} = \{e_x, e_y, e_z\}$ attached to the vehicle's center of gravity, and an inertial frame $\mathcal{I} = \{e_F, e_R, e_D\}$ located at a fix position $O_{\mathcal{I}}$ within the workspace of the vehicle. Moreover, let us define the $\{0\}$-th frame, to be the reference frame of the manipulator system which is located at the manipulator base, and each i-th frame i is located at the i-th link along the D-H convention. The end-effector fixed frame is denoted by $\mathcal{E}$. In Figure 2.3 a sketch of an UVMS with reference frames is shown. In this section, the differential kinematic relationship between the end-effector velocities expressed in the earth-fixed frame and the body-fixed system velocity will be described.

Similarly to previous section-2.1, the vehicle pose vector of the vehicle $\boldsymbol{\eta}$ is defined as here as:

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\eta}_1^T & \boldsymbol{\eta}_2^T \end{bmatrix}^T \in \Re^6 \begin{cases} \nearrow \; \boldsymbol{\eta}_1 = \begin{bmatrix} x & y & z \end{bmatrix}^T \in \Re^3 \\ \searrow \; \boldsymbol{\eta}_2 = \begin{bmatrix} \varphi & \theta & \psi \end{bmatrix}^T \in \Re^3 \end{cases} \tag{2.4}$$

where, $\boldsymbol{\eta}_1$ is the position vector of the vehicle-fixed frame $\mathcal{B}$ and $\boldsymbol{\eta}_2$ is the vector of Euler-angles of the vehicle-fixed frame. The vectors $\dot{\boldsymbol{\eta}}_1$ and $\dot{\boldsymbol{\eta}}_2$ are the corresponding time derivatives expressed in the inertial frame $\mathcal{I}$ and:

$$\dot{\boldsymbol{\eta}} = \begin{bmatrix} \dot{\boldsymbol{\eta}}_1^\top & \dot{\boldsymbol{\eta}}_2^\top \end{bmatrix}^\top \in \Re^6$$

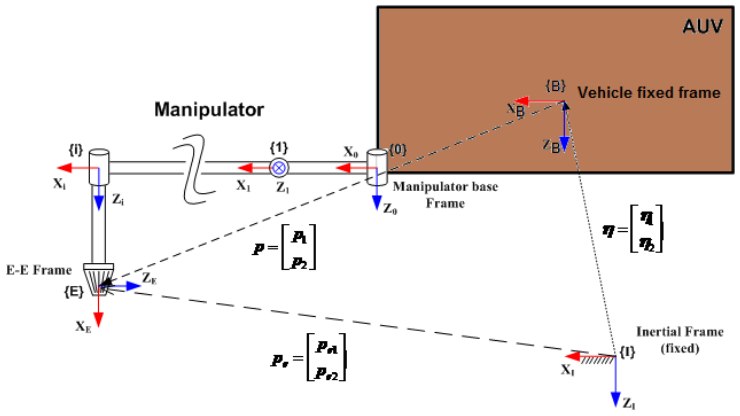


FIGURE 2.3: Schematic diagram of UVMS with coordinate frames attached.

Moreover, the vehicle-fixed velocities $\boldsymbol{v}$ is defined as:

$$\boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_1^\top & \boldsymbol{v}_2^\top \end{bmatrix}^\top \in \Re^6 \begin{cases} \nearrow \; \boldsymbol{v}_1 = \begin{bmatrix} u & v & w \end{bmatrix}^\top \in \Re^3 \\ \searrow \; \boldsymbol{v}_2 = \begin{bmatrix} p & q & r \end{bmatrix}^\top \in \Re^3 \end{cases}$$

The vector of vehicle-fixed velocities $\boldsymbol{v}$ is related to the time derivative of the vehicle pose vector $\dot{\boldsymbol{\eta}}$ by the vehicle Jacobian matrix:

$$\begin{bmatrix} \dot{\boldsymbol{\eta}}_1 \\ \dot{\boldsymbol{\eta}}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{J}_{v2}(\boldsymbol{\eta}_2) \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix} \Leftrightarrow \dot{\boldsymbol{\eta}} = \boldsymbol{J}_v(\boldsymbol{\eta}_2) \cdot \boldsymbol{v} \tag{2.5}$$

where:

$$\boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) = {}^I\boldsymbol{R}_{\mathcal{B}}(\boldsymbol{\eta}_2) = \begin{bmatrix} c\psi c\theta & -s\psi c\varphi + c\psi s\theta s\varphi & s\psi s\varphi + c\psi s\theta c\varphi \\ s\psi c\theta & c\psi c\varphi + s\psi s\theta s\varphi & -c\psi s\varphi + s\psi s\theta c\varphi \\ -s\theta & c\theta s\varphi & c\theta c\varphi \end{bmatrix} \tag{2.6}$$

$$\boldsymbol{J}_{v2}(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s\varphi t\theta & c\varphi t\theta \\ 0 & c\varphi & -s\varphi \\ 0 & s\varphi/c\theta & c\varphi/c\theta \end{bmatrix} \tag{2.7}$$

and ${}^I\boldsymbol{R}_{\mathcal{B}}(\boldsymbol{\eta}_2)$ is the rotation matrix expressing the transformation from the vehicle-fixed frame $\mathcal{B}$ to the inertial frame $\mathcal{I}$. The end-effector local pose vector $\boldsymbol{p}$ is defined as:

$$\boldsymbol{p} = \begin{bmatrix} \boldsymbol{p}_1^\top & \boldsymbol{p}_2^\top \end{bmatrix}^\top \in \Re^6 \begin{cases} \nearrow \; \boldsymbol{p}_1 = \begin{bmatrix} x_l & y_l & z_l \end{bmatrix}^T \in \Re^3 \\ \searrow \; \boldsymbol{p}_2 = \begin{bmatrix} \varphi_l & \theta_l & \psi_l \end{bmatrix}^T \in \Re^3 \end{cases}$$

where $\boldsymbol{p}_1$ is the local position vector of the e-e frame $\mathcal{E}$ expressed in the vehicle-fixed frame $\mathcal{B}$ and $\boldsymbol{p}_2$ is the local orientation (Euler-angles) vector of the e-e frame $\mathcal{E}$ expressed in the vehicle-fixed frame $\mathcal{B}$. The vectors $\dot{\boldsymbol{p}}_1$ and $\dot{\boldsymbol{p}}_2$ are the corresponding time derivatives expressed in the vehicle-fixed frame $\mathcal{B}$ and $\dot{\boldsymbol{p}} = \left[\dot{\boldsymbol{p}}_1^\top, \dot{\boldsymbol{p}}_2^\top\right]^\top \in \Re^6$. Accordingly, the end-effector pose vector $\boldsymbol{p}_e$ is defined as:

$$\boldsymbol{p}_e = \begin{bmatrix} \boldsymbol{p}_{e1}^\top & \boldsymbol{p}_{e2}^\top \end{bmatrix}^\top \in \Re^6 \begin{cases} \nearrow \; \boldsymbol{p}_{e1} = \begin{bmatrix} x_e & y_e & z_e \end{bmatrix}^\top \in \Re^3 \\ \searrow \; \boldsymbol{p}_{e2} = \begin{bmatrix} \varphi_e & \theta_e & \psi_e \end{bmatrix}^\top \in \Re^3 \end{cases}$$

where $\boldsymbol{p}_{e1}$ is the position vector of the e-e frame $\mathcal{B}$ expressed in the inertial frame $\mathcal{I}$ and $\boldsymbol{p}_{e2}$ is the orientation (Euler-angles) vector of the e-e frame $\mathcal{E}$ expressed in the inertial frame $\mathcal{I}$. The vectors $\dot{\boldsymbol{p}}_{e1}$ and $\dot{\boldsymbol{p}}_{e2}$ are the corresponding time derivatives expressed in the inertial frame $\mathcal{I}$ and $\dot{\boldsymbol{p}}_e = \begin{bmatrix} \dot{\boldsymbol{p}}_{e1}^\top & \dot{\boldsymbol{p}}_{e2}^\top \end{bmatrix}^\top \in \Re^6$. Assuming that matrix $\boldsymbol{J}_m \in \Re^{6\times n}$ represents the manipulator geometric Jacobian matrix with respect to manipulator base frame, we get:

$$\begin{bmatrix} {}^0\boldsymbol{v}_E \\ {}^0\boldsymbol{\omega}_E \end{bmatrix} = \boldsymbol{J}_m \cdot \dot{\boldsymbol{q}}_m = \begin{bmatrix} \boldsymbol{J}_{m1} \\ \boldsymbol{J}_{m2} \end{bmatrix} \cdot \dot{\boldsymbol{q}}_m \tag{2.8}$$

where $\boldsymbol{q}_m$ is the vector of the angular positions of the manipulator's joints. Moreover, the matrices $\boldsymbol{J}_{m1} \in \Re^{3\times n}$ and $\boldsymbol{J}_{m2} \in \Re^{3\times n}$ represent the position and orientation Jacobian matrices relating the contribution of the joint velocities $\dot{\boldsymbol{q}}_m$ to the e-e local linear velocity ${}^0\boldsymbol{v}_E$ and the e-e local angular velocity ${}^0\boldsymbol{\omega}_E$, respectively, both expressed in the manipulator base frame $\{0\}$. However, it is useful to express the e-e local velocity in the vehicle-fixed frame $\mathcal{B}$. The relationship between velocities in the two frames is:

$$\begin{bmatrix} \dot{\boldsymbol{p}}_1 \\ {}^{\mathcal{B}}\boldsymbol{\omega}_E \end{bmatrix} = \begin{bmatrix} {}^{\mathcal{B}}\boldsymbol{R}_0 & 0 \\ 0 & {}^{\mathcal{B}}\boldsymbol{R}_0 \end{bmatrix} \cdot \begin{bmatrix} {}^0\boldsymbol{v}_E \\ {}^0\boldsymbol{\omega}_E \end{bmatrix} \tag{2.9}$$

where the rotation matrix ${}^{\mathcal{B}}\boldsymbol{R}_0$ denotes the transformation from the manipulator base frame $\{0\}$ to the vehicle-fixed frame $\mathcal{B}$. Substituting Eq.2.8 into Eq. 2.9, the end-effector linear and angular velocity vector expressed in the vehicle-fixed frame $\mathcal{B}$ is given by:

$$\begin{bmatrix} \dot{\boldsymbol{p}}_1 \\ {}^{\mathcal{B}}\boldsymbol{\omega}_E \end{bmatrix} = \begin{bmatrix} {}^{\mathcal{B}}\boldsymbol{R}_0 & 0 \\ 0 & {}^{\mathcal{B}}\boldsymbol{R}_0 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{J}_{m1} \\ \boldsymbol{J}_{m2} \end{bmatrix} \cdot \dot{\boldsymbol{q}}_m \tag{2.10}$$

Furthermore, the following equation, relating the angular velocity ${}^{\mathcal{B}}\boldsymbol{\omega}_E$. to the time derivative of the Euler angles $\dot{\boldsymbol{p}}_2$, holds:

$${}^{\mathcal{B}}\boldsymbol{\omega}_E = \boldsymbol{E}_r\left(\boldsymbol{p}_2\right) \cdot \dot{\boldsymbol{p}}_2 \tag{2.11}$$

where

$$E_r\left(\boldsymbol{p}_2\right) = {}^{\mathcal{B}}\boldsymbol{R}_E\left(\boldsymbol{p}_2\right) \cdot \boldsymbol{J}_{v2}^{-1}\left(\boldsymbol{p}_2\right) \tag{2.12}$$

rotation matrix ${}^{\mathcal{B}}\boldsymbol{R}_E$ denotes the transformation from the e-e frame $\mathcal{E}$ to the vehicle-fixed frame $\mathcal{B}$ and $\boldsymbol{J}_{v2}\left(\boldsymbol{p}_2\right)$ is defined as in Eq.(2.7). Inverting Eq. (2.11) and considering Eq.(2.12) gives that:

$$\dot{\boldsymbol{p}}_2 = \boldsymbol{J}_{v2}\left(\boldsymbol{p}_2\right) \cdot {}^{E}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{\omega}_E \tag{2.13}$$

Combining Eq. (2.10) into Eq. (2.13), the relation of the e-e local velocity vector $\dot{\boldsymbol{p}}$ and the joint velocity vector $\dot{\boldsymbol{q}}_m$ is expressed by the following Jacobian matrix:

$$\dot{\boldsymbol{p}} = \begin{bmatrix} \dot{\boldsymbol{p}}_1 \\ \dot{\boldsymbol{p}}_2 \end{bmatrix} = \begin{bmatrix} {}^{\mathcal{B}}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m1} \\ \boldsymbol{J}_{v2}\left(\boldsymbol{p}_2\right) \cdot {}^{E}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m2} \end{bmatrix} \cdot \dot{\boldsymbol{q}}_m \tag{2.14}$$

Herein, we proceed to define the differential kinematic relationship between the end-effector velocities expressed in the earth-fixed frame $\mathcal{I}$ and the body-fixed system velocity. The coordinate transformation matrix from the end-effector frame $\mathcal{E}$ to the inertial frame $\mathcal{I}$ is expressed as below:

$${}^{I}\boldsymbol{T}_E = \begin{bmatrix} {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{R}_E & \boldsymbol{\eta}_1 + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{p}_1 \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix} \tag{2.15}$$

From the coordinate transformation Eq.(2.15), the position vector of the end-effector in the inertial frame $\mathcal{I}$ is written as:

$$\boldsymbol{p}_{e1} = \boldsymbol{\eta}_1 + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{p}_1 \tag{2.16}$$

Differentiating Eq. (2.16) with respect to time yields:

$$\dot{\boldsymbol{p}}_{e1} = \dot{\boldsymbol{\eta}}_1 + {}^{I}\dot{\boldsymbol{R}}_{\mathcal{B}} \cdot \boldsymbol{p}_1 + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \dot{\boldsymbol{p}}_1 \tag{2.17}$$

where considering that the derivative of the rotation matrix ${}^{I}\boldsymbol{R}_{\mathcal{B}}$ with respect the time is given as follows:

$${}^{I}\dot{\boldsymbol{R}}_{\mathcal{B}} = {}^{I}\boldsymbol{\omega}_{\mathcal{B}} \times {}^{I}\boldsymbol{R}_{\mathcal{B}} \tag{2.18}$$

we get the following relation:

$$\begin{aligned} &\dot{\boldsymbol{p}}_{e1} = \dot{\boldsymbol{\eta}}_1 + \left({}^{I}\boldsymbol{\omega}_{\mathcal{B}}\right) \times {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{p}_1 + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \dot{\boldsymbol{p}}_1 \;\Rightarrow \\ &\dot{\boldsymbol{p}}_{e1} = \dot{\boldsymbol{\eta}}_1 - \left({}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{p}_1\right) \times {}^{I}\boldsymbol{\omega}_{\mathcal{B}} + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \dot{p}_1 \end{aligned} \tag{2.19}$$

Considering that

$$ {}^{I}\boldsymbol{\omega}_{\mathcal{B}} = {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{v}_2 \tag{2.20} $$

and substituting Eq. (2.5) and Eq. (2.14) into Eq. (2.19), we get:

$$ \begin{aligned} \dot{\boldsymbol{p}}_{e1} = \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) \cdot \boldsymbol{v}_1 - \left({}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot p_1\right) \times {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{v}_2 \; + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m1} \cdot \dot{\boldsymbol{q}}_m \Rightarrow \\ \dot{\boldsymbol{p}}_{e1} = \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) \cdot \boldsymbol{v}_1 - \boldsymbol{S}\left({}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{p}_1\right) \cdot {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{v}_2 \; + {}^{I}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m1} \cdot \dot{\boldsymbol{q}}_m \end{aligned} \tag{2.21} $$

where $\boldsymbol{S}(\cdot)$ is the cross-product operator matrix, defined as $\boldsymbol{S}(\boldsymbol{a}) \cdot \boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$. Considering the following property:

$$ \boldsymbol{S}(\boldsymbol{R} \cdot \boldsymbol{a}) = \boldsymbol{R} \cdot \boldsymbol{S}(\boldsymbol{a}) \cdot \boldsymbol{R}^{\top} \; \forall \boldsymbol{R} \in SO(3), \; \boldsymbol{a} \in \Re^3 $$

the Eq. (2.21) is restated as:

$$ \dot{\boldsymbol{p}}_{e1} = \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) \cdot \boldsymbol{v}_1 - {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{S}(\boldsymbol{p}_1) \cdot \boldsymbol{v}_2 \; + {}^{I}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m1} \cdot \dot{\boldsymbol{q}}_m \tag{2.22} $$

As far as the angular velocities are concerned the following relation holds:

$$ {}^{I}\boldsymbol{\omega}_E = {}^{I}\boldsymbol{\omega}_{\mathcal{B}} + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{\omega}_E \tag{2.23} $$

Substituting Eq.(2.10) and Eq. (2.20) into Eq.(2.23), we get:

$$ {}^{I}\boldsymbol{\omega}_E = {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{v}_2 + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m2} \cdot \dot{q}_m \tag{2.24} $$

Moreover, as with Eq. (2.11), it holds:

$$ {}^{I}\boldsymbol{\omega}_E = \boldsymbol{E}_r(\boldsymbol{p}_{e2}) \cdot \dot{\boldsymbol{p}}_{e2} \;\Rightarrow\; {}^{I}\boldsymbol{\omega}_E = {}^{I}\boldsymbol{R}_E(\boldsymbol{p}_{e2}) \cdot \boldsymbol{J}_{v2}^{-1}(\boldsymbol{p}_{e2}) \cdot \dot{\boldsymbol{p}}_{e2} \tag{2.25} $$

Combining Eq. (2.24) with Eq. (2.25) and solving for $\dot{\boldsymbol{p}}_{e2}$ we get:

$$ \begin{aligned} {}^{I}\boldsymbol{R}_E(\boldsymbol{p}_{e2}) \cdot \boldsymbol{J}_{v2}^{-1}(\boldsymbol{p}_{e2}) \cdot \dot{\boldsymbol{p}}_{e2} = {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{v}_2 + {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m2} \cdot \dot{\boldsymbol{q}}_m \;\Rightarrow \\ \dot{\boldsymbol{p}}_{e2} = \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2}) \cdot {}^{E}\boldsymbol{R}_I \cdot {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{v}_2 + \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2}) \cdot {}^{E}\boldsymbol{R}_I \cdot {}^{I}\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m2} \cdot \dot{\boldsymbol{q}}_m \;\Rightarrow \\ \dot{\boldsymbol{p}}_{e2} = \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2}) \cdot {}^{E}\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{v}_2 + \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2}) \cdot {}^{E}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m2} \cdot \dot{\boldsymbol{q}}_m \end{aligned} \tag{2.26} $$

Now, let define the body-fixed system velocity vector $\boldsymbol{\zeta} = \left[\boldsymbol{v}^{\top}, \dot{\boldsymbol{q}}_m^{\top}\right]^{\top} \in \Re^{n+6}$. Finally, considering Eq.(2.22) and Eq.(2.26), it turns out that the e-e velocities $\dot{\boldsymbol{p}}_e = \left[\dot{\boldsymbol{p}}_{e1}^{\top}, \dot{\boldsymbol{p}}_{e2}^{\top}\right]^{\top}$ expressed in the inertial frame $\mathcal{I}$, are related to the body-fixed system velocity vector

$\boldsymbol{\zeta} = \left[\boldsymbol{v}^\top, \dot{\boldsymbol{q}}_m^\top\right]^\top$ by the following analytical Jacobian matrix:

$$\begin{bmatrix} \dot{\boldsymbol{p}}_{e1} \\ \dot{\boldsymbol{p}}_{e2} \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) & -{}^I\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{S}(\boldsymbol{p}_1) & {}^I\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m1} \\ 0 & \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2}) \cdot {}^E\boldsymbol{R}_{\mathcal{B}} & \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2}) \cdot {}^E\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m2} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \dot{\boldsymbol{q}}_m \end{bmatrix} \Rightarrow$$

$$\dot{\boldsymbol{p}}_e = \boldsymbol{J}_a(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} \tag{2.27}$$

which is also known as the UVMS differential kinematics equation [10]. In the same vein, the geometric Jacobian matrix of the UVMS can be given by:

$$\begin{bmatrix} \dot{\boldsymbol{p}}_{e1} \\ \boldsymbol{\omega}_e \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) & -{}^I\boldsymbol{R}_{\mathcal{B}} \cdot \boldsymbol{S}(\boldsymbol{p}_1) & {}^I\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m1} \\ 0 & {}^I\boldsymbol{R}_{\mathcal{B}} & {}^I\boldsymbol{R}_{\mathcal{B}} \cdot {}^{\mathcal{B}}\boldsymbol{R}_0 \cdot \boldsymbol{J}_{m2} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \dot{\boldsymbol{q}}_m \end{bmatrix} \Rightarrow$$

$$\begin{bmatrix} \dot{\boldsymbol{p}}_{e1} \\ \boldsymbol{\omega}_e \end{bmatrix} = \boldsymbol{J}_g(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} \tag{2.28}$$

2.3 Underwater Vehicle Manipulator System Dynamics

In this section, following approach given in [126] and [10] based on the Newton-Euler approach we aim to explain briefly the dynamic equations of motion of an UVMS[1]. It is well known that by knowing the forces acting on a body moving in a fluid it is possible to easily obtain the dynamics of a serial chain of rigid bodies moving in a fluid [10]. The inertial force and moments acting on the generic body are given by:

$$\boldsymbol{F}_i^i = \boldsymbol{M}_i\left[\boldsymbol{\alpha}_i^i + \dot{\boldsymbol{\omega}}_i^i \times \boldsymbol{r}_{i,c}^i + \boldsymbol{\omega}_i^i \times (\boldsymbol{\omega}_i^i \times \boldsymbol{r}_{i,c}^i)\right] \tag{2.29}$$

$$\boldsymbol{T}_i^i = \boldsymbol{I}_i^i \dot{\boldsymbol{\omega}}_i^i + \boldsymbol{\omega}_i^i \times (\boldsymbol{I}_i^i \boldsymbol{\omega}_i^i) \tag{2.29}$$

where:
$\boldsymbol{M}_i \in \Re^{3\times3}$ is the mass and added mass of link i located at the center of mass,
$\boldsymbol{I}_i^i \in \Re^{3\times3}$ is the inertia matrix plus added inertia with respect to the center of mass,
$\boldsymbol{r}_{i,c}^i$ is the vector from the origin of frame i toward the center of mass of link i expressed in frame i,
ω_i^i and $\dot{\boldsymbol{\omega}}_i^i$ are the angular velocity and angular acceleration og link i respectively,
$\boldsymbol{\alpha}_i^i$ is the linear acceleration of link i.
The forces $\boldsymbol{f}_i^i$ and moments $\boldsymbol{\mu}_i^i$ acting on the generic body of serial chain are given by

[1] For more details for analyzing the dynamics of robot manipulators based on Newton-Euler formulation, the reader is referred to [10, 126, 180]

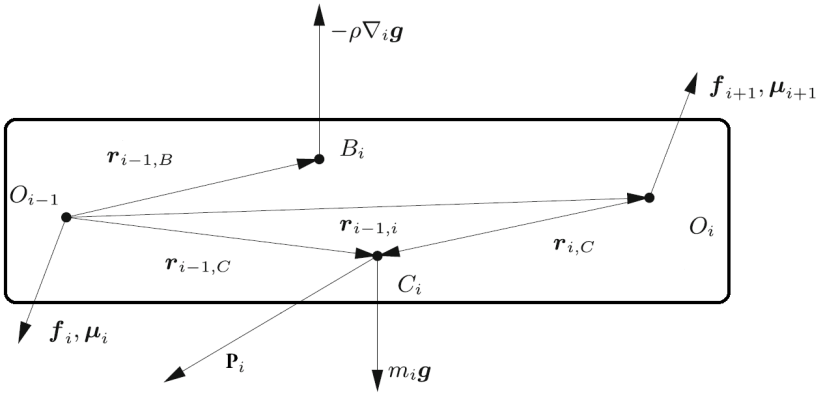


FIGURE 2.4: Force and moment acting on link i

(See Fig. 2.4):

$$\boldsymbol{f}_i^i = \boldsymbol{R}_{i+1}^i \boldsymbol{f}_{i+1}^{i+1} + \boldsymbol{F}_i^i - m_i \boldsymbol{g}^i + \rho \nabla_i \boldsymbol{g}^i + \boldsymbol{p}_i \tag{2.30}$$

$$\begin{aligned}\boldsymbol{\mu}_i^i = {} & \boldsymbol{R}_{i+1}^i \boldsymbol{\mu}_{i+1}^{i+1} + \boldsymbol{R}_{i+1}^i \boldsymbol{r}_{i-1,i}^{i+1} \times \boldsymbol{R}_{i+1}^i \boldsymbol{f}_{i+1}^{i+1} + \boldsymbol{r}_{i-1,c}^i \times \boldsymbol{F}_i^i + \boldsymbol{T}_i^i \\ & + \boldsymbol{r}_{i-1,c}^i \times (-m_i \boldsymbol{g}^i + \boldsymbol{p}_i) + \boldsymbol{r}_{i-1,b}^i \times \rho \nabla_i \boldsymbol{g}^i\end{aligned} \tag{2.30}$$

where:
$\boldsymbol{R}_{i+1}^i$ the rotation matrix between frame $i+1$ and i,
$\boldsymbol{p}_i$ linear and quadratic hydrodynamic friction forces,
$\boldsymbol{r}_{i-1,i}^i$ the vector from the origin of frame $i-1$ to the origin of frame i expressed in i,
$\boldsymbol{r}_{i-1,c}^i$ the vector from the origin of frame $i-1$ to the center of mass of link i expressed in frame i,
$\boldsymbol{r}_{i-1,b}^i$ the vector from the origin of frame $i-1$ to the center of buoyancy of link i expressed in frame i,
$\boldsymbol{g}^i = \boldsymbol{R}_I^i \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} \frac{m}{s^2}$,
ρ water density
$\rho \nabla_i \boldsymbol{g}^i$ the buoyant force.

The torque acting on joint i is finally given by:

$$\tau_{q,i} = \boldsymbol{\mu}_i^{i\top} \boldsymbol{z}_{i-1}^i \tag{2.31}$$

where the $\boldsymbol{z}_{i-1}^i$ is the unit vector along the z-xis of the link i. Let us define as $\boldsymbol{\tau}_m$ the

vector of force and moment acting on the manipulator joint torques:

$$\boldsymbol{\tau}_m = \begin{bmatrix} \tau_{m_1} \\ \vdots \\ \tau_{m_n} \end{bmatrix} \tag{2.32}$$

and the vector of force and moment acting on the vehicle as well as manipulator joint torques as:

$$\boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{\tau}_v \\ \boldsymbol{\tau}_m \end{bmatrix} \tag{2.33}$$

Now it is possible to write the equations of motion of the UVMS derived from the recursive Newton-Euler algorithm of (2.29)-(2.30) in a matrix form as:

$$\boldsymbol{M}(\boldsymbol{q}) \cdot \dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{\zeta}, \boldsymbol{q}) \cdot \boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{\zeta}, \boldsymbol{q}) \cdot \boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) = \boldsymbol{\tau} \tag{2.34}$$

where:
$\boldsymbol{M}(\boldsymbol{q}) \in \Re^{(n+6)\times(n+6)}$ is the inertia matrix including added mass,
$\boldsymbol{C}(\boldsymbol{\zeta}, \boldsymbol{q}) \in \Re^{n+6}$ is the vector of Coriolis and centripetal terms,
$\boldsymbol{D}(\boldsymbol{\zeta}, \boldsymbol{q}) \in \Re^{n_m+6}$ is the vector of dissipative effects,
$\boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \in \Re^{n+6}$ is the vector of gravity and buoyancy effects,
$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{v}^\top & \dot{\boldsymbol{q}}_m^\top \end{bmatrix}^\top \in \Re^{n+6}$ is the body-fixed system velocity vector,
$\boldsymbol{q}_m$ the vector of joint state of the manipulator.

Finally, in view of the Thruster Control Matrix of (2.3), the relationship between the generalized force $\boldsymbol{\tau}$ and the control input for an UVMS with N thruster and a manipulator with n degree of freedom is given by:

$$\boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{\tau}_v \\ \boldsymbol{\tau}_m \end{bmatrix} = \text{diag}\begin{bmatrix} \boldsymbol{B}_N, \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{T}_N \\ \dot{\boldsymbol{q}}_m \end{bmatrix} = \boldsymbol{B}\boldsymbol{u} \tag{2.35}$$

where the $\boldsymbol{u} \in \Re^{n+N}$ is the vector of the control inputs. If the end effector of the UVMS is in contact with the environment, the force/moment at the tip of the manipulator acts on the whole system according to the equation [10]:

$$\boldsymbol{M}(\boldsymbol{q}) \cdot \dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{\zeta}, \boldsymbol{q}) \cdot \boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{\zeta}, \boldsymbol{q}) \cdot \boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) + \boldsymbol{J}_g^\top(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{h}_e = \boldsymbol{\tau} \tag{2.36}$$

where $\boldsymbol{J}_g \in \Re^{6\times(n+6)}$ is the Jacobian matrix defined in Eq.(2.28) and $\boldsymbol{h}_e = [\boldsymbol{f}_e^\top, \boldsymbol{\mu}_e^\top] \in \Re^6$ is the vector of force/moment at the end effector.

2.4 Underwater Vehicle Manipulator System: Dynamic simulator

This section presents a dynamic simulation environment built in MATLAB®, following the previous analysis given in Section 2.1, Section 2.2, Section 2.3. The UVMS model considered in the simulation environment is an AUV equipped with a small 4 DoFs manipulator attached at the bow of the vehicle (see Fig.2.5). The design parameters of the AUV and the robotic manipulator are given in Tables 2.1-2.3. The dynamic equations of UVMS were derived based on the Newton-Euler approach presented in Section 2.2 and Section 2.3. The sampling time for running of the simulation is 0.1 *sec*, which is common in a real time operation with an underwater robotic system. This simulation

TABLE 2.1: Vehicle parameters

Parameter	Value	Unit
Degree of freedoms	6	
Length	0.64	m
Height	0.24	m
Width	0.25	m
Mass in air	12	kg

TABLE 2.2: Denavit-Hartenberg parameters of the robotic arm

Link	d_i	θ_i	a_i	α_i
1	L_1	q_1	0	$-\frac{\pi}{2}$
2	0	$q_2 - \frac{\pi}{2}$	L_2	0
3	0	$q_3 + \frac{\pi}{2}$	$-L_3$	$\frac{\pi}{2}$
4	L_4	q_4	0	0

TABLE 2.3: parameters of the robotic arm

Parameter	Value	Unit
Link 1 Length(L_1)	7.7	cm
Link 2 Length(L_2)	14.7	cm
Link 3 Length(L_3)	2.8	cm
Link 4 Length(L_4)	7.5	cm
Link 1 Mass	0.1	kg
Link 2 Mass	0.2	kg
Link 3 Mass	0.1	kg
Link 4 Mass	0.12	kg
Link Diameter	6	cm

environment is used in the following sections to verify the theoretical results of this dissertation where, due to lack of equipment, the implementation of a real experiment was impossible.

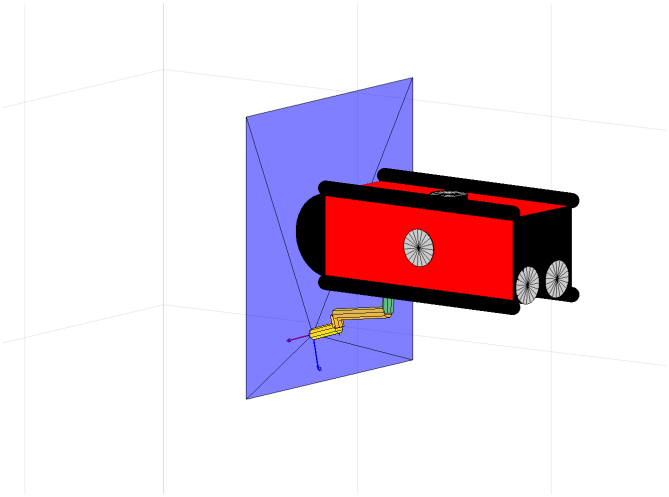


FIGURE 2.5: The dynamic simulation environment built in MATLAB®.

2.5 Concluding Remarks

In this chapter we presented the kinematic and dynamic equations of the individual robotic underwater vehicle and the underwater vehicle manipulator system.

Part III - Motion Control

Chapter 3

A Robust Predictive Control Approach for Underwater Robotic Vehicles

3.1 Introduction

In this chapter, a robust NMPC scheme for underwater robotic vehicles is presented. The purpose of the controller is to guide the vehicle towards specific way points (See Fig.3.1). Various constraints such as: sparse obstacles, workspace boundaries, control input saturation as well as predefined upper bound of the vehicle velocity (requirements for several underwater tasks such as seabed inspection scenario, mosaicking etc.) are considered during the control design. The proposed scheme incorporates the full dynamics of the vehicle in which the ocean currents are also involved. The controller is designed in order to find optimal thrusts required for minimizing the way point tracking error. Moreover, the control inputs calculated by the proposed approach are formulated in a way that the vehicle will exploit the ocean currents, when these are in favor of the way-point tracking mission, resulting in reduced energy consumption by the thrusters. The closed-loop system has analytically guaranteed stability and convergence properties. Finally, the performance of the proposed control strategy is experimentally verified using a 4 DoF underwater robotic vehicle operating inside a constrained test tank with obstacles. To the best of the authors knowledge, this is the first time where a NMPC scheme which incorporates the full dynamics of the vehicle is experimentally verified in a constrained workspace including sparse obstacles.

The rest of the chapter is organized as follows: In Section-3.2, the modeling of the underwater vehicle along with the adopted mathematical notation are presented. The verbal

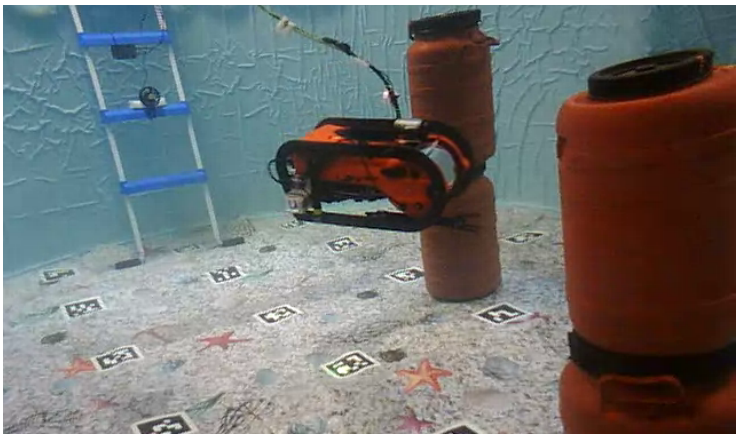


FIGURE 3.1: Experimental setup and problem formulation: the purpose of the controller is to guide the vehicle towards desired way points inside a constrained workspace including sparse obstacles.

description of the problem statement is presented in Section-3.3. An analytical description of the proposed methods including: i) the mathematical formulation of the problem, ii) the proposed control strategy and iii) the robust stability analysis are presented in Section-3.4. The efficiency of the proposed approach is illustrated in Section-3.5 via a set of experimental results. Finally, Section-3.6 concludes the chapter.

3.2 Preliminaries

3.2.1 Notation

Given two sets A and $B \subset \Re^n$, the Minkowski addition set C of two sets A and B is defined as: $C = A \oplus B = \{\mathbf{a} + \mathbf{b} : \mathbf{a} \in A, \mathbf{b} \in B\}$. The Pontryagin difference set P of two sets A and B is defined as $P = A \sim B = \{\zeta \in \Re^n : \zeta + \xi \in A, \forall \xi \in B\}$. We define as $\mathscr{B}(\boldsymbol{c}, r) = \{\boldsymbol{x} \in \Re^3 : \|\boldsymbol{x} - \boldsymbol{c}\| \leq r\}$ the closed sphere with radius r and center $\boldsymbol{c}$. For a given set $A \subset \Re^n$ we define as $\mathrm{cl}(A), \mathrm{int}(A)$ and $\partial S = \mathrm{cl}(A) \backslash \mathrm{int}(A)$ its closure, interior and boundary, respectively. Thus, we have $A = \mathrm{int}(A) \cup \partial A$.

3.2.2 Mathematical Modeling

The prior step before analyze the proposed methodology is the presentation of the preliminary aspects of the modeling of underwater vehicles. Firstly, let us define a common body-fixed frame $\mathscr{V}$ attached to the vehicle center of gravity, as well as the inertial frame $\mathscr{I}$. The pose vector of the vehicle with respect to (w.r.t) the inertial frame $\mathscr{I}$ is denoted by $\boldsymbol{\eta} = \left[\boldsymbol{\eta_1^T}\ \boldsymbol{\eta_2^T}\right]^T \in \Re^6$ including the position (i.e., $\boldsymbol{\eta_1} = [x\ y\ z]^T$) and orientation (i.e.,

$\boldsymbol{\eta_2} = [\phi\ \theta\ \psi]^T$) vectors. The $\boldsymbol{v} = \left[\boldsymbol{v_1^T}\ \boldsymbol{v_2^T}\right]^T \in \Re^6$ is the velocity vector of the vehicle expressed in fixed-body frame $\mathscr{V}$ and includes the linear (i.e., $\boldsymbol{v_1} = [u\ v\ w]^T$) and angular (i.e., $\boldsymbol{v_2} = [p\ q\ r]^T$) velocity vectors. In this chapter, we consider that the vehicle operates under the influence of bounded irrotational ocean currents w.r.t the inertial frame $\mathscr{I}$. An estimation of the ocean currents can be achieved by employing the data obtained from Naval Coastal Ocean Model (NCOM) [181] and Regional Ocean Model Systems (ROMS)[182]. However, an estimation of the ocean current could be achieved locally using an appropriate estimator [183, 184]. Thus, in the following analysis, we consider the effect of ocean currents during the control design, but we assume that these data are inaccurate and the uncertainties on the ocean current's profile should be handled during the robustness analysis. In this chapter, the bounded irrotational ocean current velocities w.r.t the inertial frame $\mathscr{I}$ is denoted by $\boldsymbol{v}_c^{\mathscr{I}} = [(\boldsymbol{v}_{c_1}^{\mathscr{I}})^T, \boldsymbol{0}_{1\times 3}]^T \in \Re^6$ with $\boldsymbol{v}_{c_1}^{\mathscr{I}} = [u_c^{\mathscr{I}}, v_c^{\mathscr{I}}, w_c^{\mathscr{I}}]^T$ to be the vector of linear velocity terms. Therefore, we can define the vehicle velocity vector relative to the water expressed in body frame $\mathscr{V}$ as:

$$\boldsymbol{v}_r = \boldsymbol{v} - \boldsymbol{v}_c \tag{3.1}$$

Notice that the vector $\boldsymbol{v}_c = [u_c, v_c, w_c, \boldsymbol{0}_{1\times 3}]^T$ indicates the expression of the ocean currents with respect to the body frame $\mathscr{V}$. Without loss of generality, according to the standard underwater vehicles' modeling properties [154], assuming that the current velocity is slowly varying with respect to the inertial frame (e.g, $\frac{\partial \boldsymbol{v}_c^{\mathscr{I}}}{\partial t} \cong 0$), and the vehicle is operating at relative low speeds, the dynamic equations of the vehicle can be given as [154, eq:3.112-3.116]:

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\,\boldsymbol{v}_r + \boldsymbol{v}_c^{\mathscr{I}} \tag{3.2 }$$

$$\boldsymbol{M}\dot{\boldsymbol{v}}_r + \boldsymbol{C}(\boldsymbol{v}_r)\,\boldsymbol{v}_r + \boldsymbol{D}(\boldsymbol{v}_r)\boldsymbol{v}_r + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}_{\mathscr{V}} \tag{3.2 }$$

where:

- $\boldsymbol{\tau}_{\mathscr{V}} = [X,\ Y,\ Z,\ K,\ M,\ N]^T \in \Re^6$ is the total propulsion force/torque vector (i.e., the body forces and torques generated by the thrusters) applied on the vehicle and expressed in $\mathscr{V}$;
- $\boldsymbol{M} = \boldsymbol{M}_{RB} + \boldsymbol{M}_A$, where $\boldsymbol{M}_{RB} \in \Re^{6\times 6}$ and $\boldsymbol{M}_A \in \Re^{6\times 6}$ are the inertia matrix for the rigid body and added mass respectively;
- $\boldsymbol{C}(\boldsymbol{v}_r) = \boldsymbol{C}_{RB}(\boldsymbol{v}_r) + \boldsymbol{C}_A(\boldsymbol{v}_r)$, where $\boldsymbol{C}_{RB}(\boldsymbol{v}_r) \in \Re^{6\times 6}$ and $\boldsymbol{C}_A(\boldsymbol{v}_r) \in \Re^{6\times 6}$ are the coriolis and centripetal matrix for the rigid body and added mass respectively;
- $\boldsymbol{D}(\boldsymbol{v}_r) = \boldsymbol{D}_{quad}(\boldsymbol{v}_r) + \boldsymbol{D}_{lin}(\boldsymbol{v}_r)$, where $\boldsymbol{D}_{quad}(\boldsymbol{v}_r) \in \Re^{6\times 6}$ and $\boldsymbol{D}_{lin}(\boldsymbol{v}) \in \Re^{6\times 6}$ are the quadratic and linear drag matrix respectively;

- $\boldsymbol{g}(\boldsymbol{\eta}) \in \Re^6$ is the hydrostatic restoring force vector;
- $\boldsymbol{J}(\boldsymbol{\eta}) = \begin{bmatrix} \boldsymbol{J}_1(\boldsymbol{\eta_2}) & \mathbf{O}_{3\times3} \\ \mathbf{O}_{3\times3} & \boldsymbol{J}_2(\boldsymbol{\eta_2}) \end{bmatrix}$ is the Jacobian matrix transforming the velocities from the body-fixed ($\mathscr{V}$) to the inertial ($\mathscr{I}$) frame, in which $\boldsymbol{J}_1(\boldsymbol{\eta_2}) \in SO(3)$ is the well known rotation matrix and $\boldsymbol{J}_2(\boldsymbol{\eta_2}) \in \Re^{3\times3}$ denotes the lumped transformation matrix;

Notice that the transformation from ocean current velocity defined in the inertial frame $\mathscr{I}$ (i.e., $\boldsymbol{v}_c^{\mathscr{I}}$) into body-fixed one (i.e., $\boldsymbol{v}_c$) is achieved using the transposed rotation matrix i.e., $\boldsymbol{v}_c = \boldsymbol{J}^T(\boldsymbol{\eta})\boldsymbol{v}_c^{\mathscr{I}}$ (See[154]). In Equation 3.2, the total propulsion force/torque vector ($\boldsymbol{\tau}_{\mathscr{V}}$) is computed using the thruster allocator matrix which is formulated by the actuation geometry and properties of the underwater vehicle's thrusters. The vehicle used in this chapter is a 4 DoF Seabotix LBV150. It is equipped with 4 thrusters (i.e., Port (p_o), Starboard (s), Vertical (v_e), Lateral (l)), which are effective in Surge (X), Sway (Y), Heave (Z) and Yaw (N) motion. Thus, we can define a new thrust vector ($\boldsymbol{\tau} = [\tau_{p_o},\ \tau_s,\ \tau_{v_e},\ \tau_l]^T \in \Re^4$) and the appropriate thruster allocator matrix ($\boldsymbol{T_A} \in \Re^{4\times4}$) such as:

$$\boldsymbol{\tau}_{LBV}^{\mathscr{V}} = \boldsymbol{T_A}\boldsymbol{\tau}, \tag{3.3}$$

where $\boldsymbol{\tau}_{LBV}^{\mathscr{V}}\,[X,\ Y,\ Z,\ N]^T \in \Re^4$.

Remark 3.1. In the vehicle used in this chapter, the angles ϕ, θ and angular velocities p and q are negligible and we can consider them to be equal to zero. Thus, from now on, we denote $\boldsymbol{\eta} = [x, y, z, \psi]$ and $\boldsymbol{v} = [u, v, w, r]$. The vehicle is symmetric about x - z plane and close to symmetric about y - z plane. Therefore, we can safely assume that motions in heave, roll and pitch are decoupled [154].

3.3 Problem Formulation

In this section we formally address the problem under consideration:

Problem 3.1. *Given an underwater vehicle with dynamics as described in (3.2), design a robust feedback control law for the autonomous guidance towards a set of way-points* $\boldsymbol{\eta}_i^d,\ i = \{1, \ldots, n\}$*, while guaranteeing the following specifications:*

- *Avoid the workspace boundaries and a limited set of obstacles within.*
- *Respect operational limitations in the form of state (e.g velocity bounds) and input (thrust saturation) constraints.*
- *The energy consumed by thrusters to be retained in a reduced level.*

3.4 Methodology

In this section, we present in detail the methodologies proposed in order to formulate the solution of Problem 3.1 as defined in Section 3.3.

3.4.1 Geometry of Workspace

Consider an underwater vehicle which operates inside the workspace $\mathscr{W} \subset \Re^3$ with boundaries $\partial\mathscr{W} = \{\boldsymbol{p} \in \Re^3 : \boldsymbol{p} \in \text{cl}(\mathscr{W})\backslash\text{int}(\mathscr{W})\}$ and sparse obstacles located within. Without loss of generality, the robot and the obstacles can be modeled as spheres (i.e., we adopt the spherical world representation [185]). In this spirit, let $\mathscr{B}(\boldsymbol{\eta}_1, \bar{r})$ to be a closed sphere that completely surrounds the vehicle volume (main body and additional equipments). Moreover, the $\mathscr{M}$ static obstacles within the workspace are defined as closed spheres described by $\pi_m = \mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m})$, $m \in \{1, \ldots, \mathscr{M}\}$, where $\boldsymbol{p}_{\pi_m} \in \Re^3$ is the center and the $r_{\pi_m} > 0$ the radius of the obstacle π_m. Additionally, based on the property of spherical world [185], for each pair of obstacles $m, m' \in \{1, \ldots, \mathscr{M}\}$ the following inequality holds:

$$||\pi_m - \pi_{m'}|| > 2\bar{r} + r_{\pi_m} + r_{\pi'_m} \tag{3.4}$$

which intuitively means that the obstacles m and m' are disjoint in such a way that the entire volume of the vehicle can pass through the free space between them. Therefore, there exists a feasible trajectory $\boldsymbol{\eta}(t)$ for the vehicle that connects the initial configuration $\boldsymbol{\eta}(t_0)$ with $\boldsymbol{\eta}^d$ such as:

$$\mathscr{B}(\boldsymbol{\eta}_1(t), \bar{r}) \cap \{\mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m}) \cup \partial\mathscr{W}\} = \emptyset, \ m \in \{1, \ldots, \mathscr{M}\}$$

A graphical representation of the feasible trajectory is depicted in Fig. 3.2.

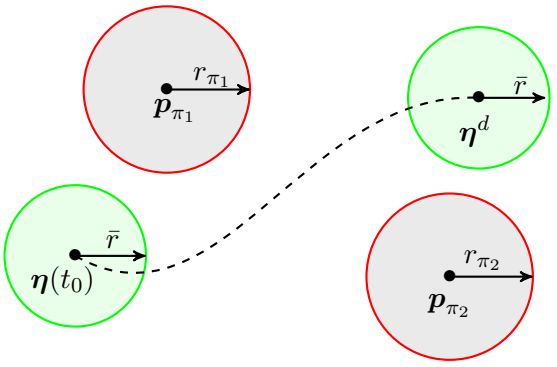


FIGURE 3.2: Graphical representation of a feasible transition of the underwater vehicle from the initial position $\boldsymbol{\eta}(t_0)$ to the desired position $\boldsymbol{\eta}^d$.

3.4.2 Dynamical system

Due to the aforementioned assumptions and following standard simplifications due to symmetries in the mass configuration [154], the dynamic equation (3.2) for the vehicle under consideration, can be written in discrete-time form as:

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{\tau}_k) \Rightarrow \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \mathscr{A}(\boldsymbol{x}_k)dt + \mathscr{C}(\boldsymbol{\tau}_k)dt \tag{3.5}$$

where:

$$\mathscr{A}(\boldsymbol{x}_k) = \begin{bmatrix} u_{r_k} c\psi_k - v_{r_k} s\psi_k + u_c^{\mathscr{I}} \\ u_{r_k} s\psi_k + v_{r_k} c\psi_k + v_c^{\mathscr{I}} \\ w_{r_k} + w_c^{\mathscr{I}} \\ r_{r_k} \\ \frac{1}{m_{11}}(m_{22} v_{r_k} r_{r_k} + X_u u_{r_k} + X_{u|u|} |u_{r_k}| u_{r_k}) \\ \frac{1}{m_{22}}(-m_{11} u_{r_k} r_{r_k} + Y_v v_{r_k} + Y_{v|v|} |v_{r_k}| v_{r_k}) \\ \frac{1}{m_{33}}(Z_w w_{r_k} + Z_{w|w|} |w_{r_k}| w_{r_k}) \\ \frac{1}{m_{44}}((m_{11} - m_{22}) u_{r_k} v_{r_k} + N_r r_{r_k} + N_{r|r|} |r_{r_k}| r_{r_k}) \end{bmatrix}, \quad \mathscr{C}(\boldsymbol{\tau}_k) = \begin{bmatrix} \mathbf{0}_{4\times 1} \\ \boldsymbol{T_A} \boldsymbol{\tau}_k \end{bmatrix}$$

with $c(\cdot) = \cos(\cdot), s(\cdot) = \sin(\cdot)$ and $\mathbf{x_k} = [\boldsymbol{\eta}_k^T, \boldsymbol{v}_{r_k}^T]^T = [x_k,\ y_k,\ z_k,\ \psi_k,\ u_{r_k},\ v_{r_k},\ w_{r_k},\ r_{r_k}]^\top \in \Re^8$ denotes the state vector at the time-step k which includes the position and orientation of the vehicle with respect to the inertial frame $\mathscr{I}$ and the relative linear and angular velocity of the vehicle with respect to the water. In addition, $m_{ii}, i = 1, \ldots, 4$ are the mass terms including added mass, $X_u,\ Y_v,\ Z_w,\ N_r\ < 0$ are the linear drag terms, $X_{u|u|},\ Y_{v|v|},\ Z_{w|w|},\ N_{r|r|}\ < 0$ are the quadratic drag terms, while dt denotes the sampling period. The control input of the system is $\boldsymbol{\tau}_k = [\tau_{p_k},\ \tau_{s_k},\ \tau_{v_k},\ \tau_{l_k}]^T \in \Re^4$ consisting of the thrusters' forces. As mentioned previously, the ocean current profile induced to the dynamic model is inaccurate (i.e., in general, it is an estimation achieved by employing data from NCOM or ROMS) and consequently a difference between the real value of the current and the estimated one must be considered. The effect of these uncertainties can be considered as disturbances on the system (3.5). Thus, by $\delta_{\mathbf{k}} = [\mathbf{0}_{1\times 4},\ \delta_{u_k},\ \delta_{v_k},\ \delta_{w_k},\ \delta_{r_k}]^\top \in D \subset \Re^8$ we present the effect of ocean current profile uncertainties on the system at the time step k, with D to be a compact set. Moreover, D is bounded by $||\delta_{\mathbf{k}}|| \leq \bar{\delta}$. Furthermore, it is assumed that vehicle's dynamic parameters have been identified via a proper identification scheme. However some degree of uncertainty on dynamic parameters denoted by $\Delta f(\mathbf{x}_k, \tau_k)$ should be considered. Taking into consideration the aforementioned disturbances that affect the vehicle, we are now ready

to model the perturbed system as follows:

$$\mathbf{x}_{k+1} = \tilde{f}(\mathbf{x}_k, \tau_k) + \boldsymbol{\delta}_k = f(\mathbf{x}_k, \tau_k) + \Delta f(\mathbf{x}_k, \tau_k) + \boldsymbol{\delta}_k = f(\mathbf{x}_k, \tau_k) + \boldsymbol{\gamma}_k + \boldsymbol{\delta}_k \tag{3.6}$$

with:

$$\boldsymbol{\gamma}_k = \Delta f(\mathbf{x}_k, \tau_k) \in \Gamma,\ ||\boldsymbol{\gamma}_k|| \leq \bar{\gamma}\ \forall \mathbf{x}_k \in X, \tau_k \in T$$

where Γ is the compact set of uncertainties and $\bar{\gamma} \geq 0$ is a positive upper bound for this set. The equation (3.6) can be rewritten as:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \boldsymbol{\tau}_k) + \boldsymbol{w}_k \tag{3.7}$$

with $\boldsymbol{w}_k = \boldsymbol{\gamma}_k + \boldsymbol{\delta}_k \in W \subset \Re^8$ as the result of adding uncertainties and external disturbances of the system. W is a compact set such that $W = D \oplus \Gamma$. Since the sets D and Γ are compact, W is also a compact set, bounded by $||\boldsymbol{w}_k|| \leq \bar{w}$ with $\bar{w} \triangleq \bar{\gamma} + \bar{\delta}$.

Remark 3.2. Notice that Eq. (3.7) is the actual dynamical equation of the system, since it contains the vector of disturbance effecting on the system. In this way, we consider Eq.(3.5) as the nominal model of the system, in which no disturbances are considered.

Property 3.1. *The nominal model $f(\mathbf{x}, \tau)$ is locally Lipschitz in X i.e., there is a positive Lipschitz constant $L_f < \infty$, such that for every $\tau \in T$, $\|f(\mathbf{x}_1, \tau) - f(\mathbf{x}_2, \tau)\| \leq L_f \|\mathbf{x}_1 - \mathbf{x}_2\|$.*

3.4.3 Constraints

State Constraints:

In this chapter, we consider that the robot must avoid the obstacles and the workspace boundaries (test tank). Moreover, for the needs of several common underwater tasks (e.g., seabed inspection, mosaicking), the vehicle is required to move with relatively low speeds with upper bound denoted by the velocity vector $\boldsymbol{v}_p = [u_p\ v_p\ w_p\ r_p]^\top$. These requirements are captured by the state constraint set X of the system, given by:

$$\mathbf{x_k} \in X \subset \Re^8 \tag{3.8}$$

which is formed by the following constraints:

$$u_p + v_p - |u_r + v_r| \geq 0 \tag{3.9}$$

$$w_p - |w_r| \geq 0 \tag{3.10}$$

$$r_p - |r_r| \geq 0 \tag{3.11}$$

$$\mathscr{B}(\boldsymbol{\eta}_1(t), \bar{r}) \cap \{\mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m}) \cup \partial\mathscr{W}\} = \emptyset,\ m \in \{1, \ldots, \mathscr{M}\} \tag{3.12}$$

Input Constraints:
The actuation body forces and torques are generated by the thrusters. Thus, we define the control constraint set T as follows:

$$\boldsymbol{\tau}_k = [\tau_{p_{o_k}},\ \tau_{s_k},\ \tau_{v_{e_k}},\ \tau_{l_k}]^T \in T \subseteq \Re^4 \tag{3.13}$$

These constraints are of the form $|\tau_{p_{o_k}}| \leq \bar{\tau}_{p_o}$, $|\tau_{s_k}| \leq \bar{\tau}_s$, $|\tau_{v_{e_k}}| \leq \bar{\tau}_{v_e}$ and $|\tau_{l_k}| \leq \bar{\tau}_l$, therefore we get $|||\boldsymbol{\tau}_k|| \leq \bar{T}$ where $\bar{T} = (\tau_{p_o}^2 + \tau_s^2 + \tau_{v_e}^2 + \tau_l^2)^{\frac{1}{2}}$ and $\bar{\tau}_{p_o}, \bar{\tau}_s, \bar{\tau}_{v_e}, \bar{\tau}_l \in \Re_{\geq 0}$.

3.4.4 Control Design

The control objective is to guide the actual system (3.7) to a desired compact set around the way points $i = \{1, \ldots, n\}$ that includes the desired state ${}^i\mathbf{x}^d \triangleq [({}^i\boldsymbol{\eta}^d)^T, ({}^i\boldsymbol{v}_r^d)^T]^T = [{}^i x_d, {}^i y_d, {}^i z_d, {}^i \psi_d, {}^i u_d, {}^i v_d, {}^i w_d, {}^i r_d]^T \in X$, while respecting the state constraints (3.9)-(3.12) as well as the input constraints (3.13). A predictive controller is employed in order to achieve this task. In particular, at a given time instant k, the NMPC is assigned to solve an Optimal Control Problem (OCP) with respect to a control sequence $\boldsymbol{\tau}_f(k) \triangleq [\tau(k|k), \tau(k+1|k), \ldots, \tau(k+N-1|k)]$, for a prediction horizon N. The OCP of the NMPC is given as follows:

$$\min_{\boldsymbol{\tau}_f(k)} J_N(\boldsymbol{x}_k, \boldsymbol{\tau}_f(k)) = \tag{3.14}$$
$$\min_{\boldsymbol{\tau}_f(k)} \sum_{j=0}^{N-1} F(\hat{\boldsymbol{x}}(k+j|k), \boldsymbol{\tau}(k+j|k)) + E(\hat{\boldsymbol{x}}(k+N|k))$$

subject to:

$$\hat{\boldsymbol{x}}(k+j|k) \in X_j, \qquad \forall j = 1, \ldots, N-1, \tag{3.15}$$

$$\boldsymbol{\tau}(k+j|k) \in T, \qquad \forall j = 0, \ldots, N-1, \tag{3.16}$$

$$\hat{\boldsymbol{x}}(k+N|k) \in \mathscr{E}_f \tag{3.17}$$

where $\mathscr{E}_f$ is the terminal set and F and E are the running and terminal cost functions, respectively. At time instant k, the solution of the OCP (3.14)-(3.17) is providing an optimal control sequence, denoted as:

$$\boldsymbol{\tau}_f^*(k) = [\tau(k|k), \tau(k+1|k), \ldots, \tau(k+N-1|k)] \tag{3.18}$$

where the first control vector (i.e., $\boldsymbol{\tau}(k|k)$) is applied to the system. The disturbance term $\boldsymbol{w}_k$ can cause discrepancies between the predicted state given from the nominal model (3.5) subject to a specific sequence of inputs and the actual state, given from (3.7) for the same sequence of inputs. In order to account for this mismatch we use the double

subscript notation for the predicted state of system (3.5) inside the OCP of the NMPC:

$$\hat{\boldsymbol{x}}(k+j|k) = f(\hat{\boldsymbol{x}}(k+j-1|k), \tau(k+j-1|k)) \tag{3.19}$$

where the vector $\hat{\boldsymbol{x}}(k+j|k)$ denotes the predicted state of the nominal system (3.5) at sampling time $k+j$ with $j \in \mathbb{Z}_{\geq 0}$. The predicted state is based on the measurement of the state $\boldsymbol{x}_k$ of the actual system at sampling time k (i.e., provided by the on–board navigation system), while applying a sequence of control inputs $[\tau(k|k), \tau(k+1|k), \ldots, \tau(k+j-1|k)]$. It holds that $\hat{\boldsymbol{x}}(k|k) \equiv \boldsymbol{x}_k$. The cost function $F(\cdot)$, as well as the terminal cost $E(\cdot)$, are both of quadratic form, i.e., $F(\hat{\boldsymbol{x}}, \tau) = \hat{\boldsymbol{x}}^\top Q\hat{\boldsymbol{x}} + \tau^\top R\tau$ and $E(\hat{\boldsymbol{x}}) = \hat{\boldsymbol{x}}^\top P\hat{\boldsymbol{x}}$, respectively, with P, Q and R being positive definite matrices. Particularly we define $Q = \text{diag}\{q_1, \ldots, q_8\}$, $R = \text{diag}\{r_1, \ldots, r_4\}$ and $P = \text{diag}\{p_1, \ldots, p_8\}$. Obviously for the running cost function F, we have $F(0,0) = 0$.

Notice, here that the OCP is solved for the nominal system, i.e., the model of the system that is not affected by disturbances. This is the case, due to the fact that the OCP is solved for a prediction horizon, thus, it is not possible to address the disturbances beforehand. However, we distinguish the nominal system that will be denoted as $\hat{\boldsymbol{x}}(\cdot)$ with the actual system, i.e., the system that is affected by disturbances that will be denoted as $\boldsymbol{x}(\cdot)$. Thus, it is shown that the difference between the real evolution of the state, given by (3.7) and the predicted evolution of the state, given by (3.5), is in fact, bounded:

Lemma 3.1. *The difference between the actual state $\boldsymbol{x}_{k+j}$ at the time-step $k+j$ and the predicted state $\hat{\boldsymbol{x}}(k+j|k)$ at the same time-step, under the same control sequence, is upper bounded by (See Appendix-11.1 for the Proof):*

$$||\boldsymbol{x}_{k+j} - \hat{\boldsymbol{x}}(k+j|k)|| \leq \sum_{i=0}^{j-1} L_f \bar{w} \tag{3.20}$$

Notice that in (3.15), the state constraint set X from (3.8), is being replaced by a restricted constraint set $X_j \subseteq X$. This state constraints' tightening for the nominal system guarantees that the evolution of the real system will be admissible for all time. More specifically, using this technique, it can be guaranteed that the evolution of the state of the perturbed system (3.7), when the control sequence developed from the NMPC Problem of (3.14)-(3.17) is applied to it, will necessarily satisfy the state constraint set X. More specifically, given Lemma-3.1, where a bound on the state prediction error is evaluated, we set $X_j = X \sim \mathscr{B}_j$ where $\mathscr{B}_j = \{\boldsymbol{x} \in \Re^8 : ||\boldsymbol{x}|| \leq \sum_{i=0}^{j-1} L_f \bar{w}\}$. More details on this constraint tightening technique can be found in [186].

Before proceeding to the necessary robust analysis of the proposed NMPC strategy, we employ some standard stability conditions that are used in MPC frameworks, as the following:

Assumption 3.1. For the nominal system (3.5), there is an admissible positively invariant set $\mathscr{E} \subset X$ such that the terminal region $\mathscr{E}_f \subset \mathscr{E}$, where $\mathscr{E} = \{\boldsymbol{x} \in X : ||\boldsymbol{x}|| \leq \varepsilon_0\}$ and ε_0 being a positive parameter.

Assumption 3.2. We assume that in the terminal set $\mathscr{E}_f$, there exists a local stabilizing controller $\boldsymbol{\tau}_k = h(\boldsymbol{x}_k) \in T$ for all $\boldsymbol{x} \in \mathscr{E}$, and that E satisfies $E(f(\boldsymbol{x}_k, h(\boldsymbol{x}_k))) - E(\boldsymbol{x}_k) + F(\boldsymbol{x}_k, h(\boldsymbol{x}_k)) \leq 0$ for all $\boldsymbol{x} \in \mathscr{E}$.

Assumption 3.3. The terminal cost function E is Lipschitz in $\mathscr{E}$, with Lipschitz constant $L_E = 2\varepsilon_0\sigma_{max}(P)$ for all $\boldsymbol{x} \in \mathscr{E}$, where $\sigma_{\max}(P)$ denotes the largest singular value of P.

Assumption 3.4. Inside the set $\mathscr{E}$ we have $E(\boldsymbol{x}) = \boldsymbol{x}^T P \boldsymbol{x} \leq \alpha_\varepsilon$, where $\alpha_\varepsilon = \max\{p_1, \ldots, p_8\}\varepsilon_0^2 > 0$. Assuming that $\mathscr{E} = \{\boldsymbol{x} \in X_{(N-1)} : h(\boldsymbol{x}) \in T\}$ and taking a positive parameter α_{ε_f} such that $\alpha_{\varepsilon_f} \in (0, \alpha_\varepsilon)$, we assume that the terminal set designed as $\mathscr{E}_f = \{\boldsymbol{x} \in \Re^8 : E(\boldsymbol{x}) \leq \alpha_{\varepsilon_f}\}$ is such that $\forall \boldsymbol{x} \in \mathscr{E},\ f(\boldsymbol{x}, h(\boldsymbol{x})) \in \mathscr{E}_f$.

Lemma 3.2. *The cost function $F(\boldsymbol{x}, \boldsymbol{\tau})$ is such that $F(\mathbf{0}, \mathbf{0}) = 0$ and $\underline{F}(||\boldsymbol{x}||) \leq F(\boldsymbol{x}, \boldsymbol{\tau})$, $\forall \boldsymbol{x} \in X$ and $\forall \boldsymbol{\tau} \in T$ where $\underline{F}$ is a $\mathscr{K}_\infty$ function. Furthermore, $F(\boldsymbol{x}, \boldsymbol{\tau})$ is Lipschitz continuous with respect to $\boldsymbol{x}$ in X, with a Lipschitz constant $L_F \in \Re_{>0}$.*

Notice that a $\mathscr{K}_\infty$ function, as well as the Lipschitz constant L_F of the Lemma 3.2, can be found analytically (after some mathematic manipulations), due to the quadratic form of the cost function $F(\boldsymbol{x}, \boldsymbol{\tau})$.

Remark 3.3 (Thrust's energy consumption)**.** The OCP of the proposed NMPC scheme is designed in order to find the optimal thrust $\boldsymbol{\tau}_f^*(\cdot)$ required for minimizing the state error. Thus, owing to the existence of ocean currents profile in the dynamic model (3.2), the control inputs $\boldsymbol{\tau}_f^*(\cdot)$ calculated by the OCP(3.14)-(3.17) may exploit the ocean currents when these are in favor of the way point tracking mission. Hence, the proposed scheme calculates the optimal commands, in order to retain the energy consumed by the thrusters in a reduced level, as dictated by the requirements of the considered Problem-3.1.

Remark 3.4. The obstacles within the workspace may be detected on–line by the vehicle's on–board sensors (e.g., multi beam imaging or side scan sonar). In such a case, it should be assured that the predicted state of the NMPC is always within the sensing region of the robot. This intuitively means that the prediction state is always feasible even in the worst case (i.e., maximum velocity of the robot under maximum sea current). Thus, assuming that $\bar{R}$ denotes the sensing range of the system, the prediction horizon N should be set as follows:

$$N \leq \frac{\bar{R}}{(|\bar{\boldsymbol{v}}_1| + |\bar{\boldsymbol{v}}_c|)dt}$$

where the $|\bar{\boldsymbol{v}}_1|$ is the norm of maximum linear velocity of the vehicle and $|\bar{\boldsymbol{v}}_c|$ is the norm of the upper bound of the sea current velocity.

3.4.5 Stability Analysis of the Proposed NMPC

The approach for establishing stability consists of two parts: in the first part it is shown that the initial feasibility implies feasibility afterwards and based on this, it is then shown that the state converges to a bounded set, due to the presence of the persistent disturbances [187–189]. We begin by denoting $\boldsymbol{\tau}_f^*(k-1)$ as the optimal solution that results from (3.14)-(3.17) at a time-step $k-1$. We, also, denote a feasible control sequence $\tilde{\boldsymbol{\tau}}_f(k+j|k)$ of the optimization problem at time-step k, such as:

$$\tilde{\boldsymbol{\tau}}(k+j|k) = \begin{cases} \boldsymbol{\tau}^*(k+j|k-1) & \text{for } j = 0, \ldots, N-2 \\ h(\hat{\boldsymbol{x}}(k+N-1|k)) & \text{for } j = N-1 \end{cases} \tag{3.21}$$

where $h(\boldsymbol{x})$ is the local stabilizing controller defined in *Assumption* 3.2. Moreover, from (3.16) and *Assumption* 3.2 is clear that $\tilde{\boldsymbol{\tau}}(k+j|k) \in T$.

Feasibility analysis:

First we are going to provide a necessary definition: Let $\mathscr{X}^{MPC}$ be the set containing all the state vectors for which a feasible control sequence exists that satisfies the constraints of the optimal control problem. In particular, while having a slight violation of the notation we can define the feasible set $\mathscr{X}^{MPC}$ as follows:

Definition 3.1. $\mathscr{X}^{MPC} = \{\boldsymbol{x}_0 \in \Re^n | \exists$ a control sequence $\boldsymbol{\tau}_f \in T,\ \hat{\boldsymbol{x}}_f(j) \in \mathscr{X}_j\ \ \forall j \in \{1, \ldots, N\}$ and $\hat{\boldsymbol{x}}(N) \in \mathscr{E}_f\}$.

At this point we want to find a bound $\bar{w}$ for the uncertainties, then the closed-loop system in $\mathscr{X}^{MPC}$ is stable. That means that if $\boldsymbol{x}_k \in \mathscr{X}^{MPC}$ then $\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{\tau}_k^*) + \boldsymbol{w}_{k+1} \in \mathscr{X}^{MPC}$, for all $\boldsymbol{w}_{k+1} \in W$. In order to derive feasibility it must be guaranteed that $\hat{\boldsymbol{x}}(k+N|k) \in \mathscr{E}_f$. This results to a permitted upper bound of disturbances $\bar{w}$ under which the system is proven to be feasible. See Appendix-11.2 for the Proof.

Convergence analysis:

In order to treat the convergence property, it should be guaranteed that the state of the perturbed system reaches to a desired terminal set. A proper value function must be shown to be decreasing in order to prove stability of the closed loop system and consequently the state convergence of the system. Consider the optimal cost $J_N^*(k-1) =$

$J^*(\boldsymbol{x}_{k-1}, \boldsymbol{\tau}_f^*(k-1))$ from (3.14), at the time-step $(k-1)$ as a Lyapunov function candidate. Consider, also, the cost of the feasible sequence at time-step k as $\tilde{J}_N(k) = \tilde{J}(\boldsymbol{x}_k, \tilde{\boldsymbol{\tau}}_f(k))$ evaluated from the control sequence $\tilde{\boldsymbol{\tau}}_f(k+i|k)$. After some mathematical manipulations, it can be hold for the difference $\tilde{J}_N(k) - J_N^*(k-1)$ that (See Appendix-11.3 for the Proof):

$$\begin{aligned}\Delta J = \tilde{J}_N(k) - J_N^*(k-1) &\leq \left(L_F \sum_{i=0}^{i=N-2} L_f^i + L_E L_f^{N-1}\right)\bar{w} \\ &- F(\hat{\boldsymbol{x}}(k-1|k-1), \boldsymbol{\tau}^*(k-1|k-1)) \\ &\leq \left(L_F \sum_{i=0}^{i=N-2} L_f^i + L_E L_f^{N-1}\right)\bar{w} - \underline{\mathrm{F}}(||\boldsymbol{x}_{k-1}||)\end{aligned}$$

From the optimality of the solution, we derive the following:

$$\Delta J^* = J_N^*(k) - J_N^*(k-1) \leq \Delta J \tag{3.22}$$

Therefore, the optimal cost $J_N^*(k)$ is an ISS-Lyapunov function of the closed loop system, and hence, the closed-loop system is input-to-state stable.

3.5 Experimental results

This section demonstrates the efficacy of the proposed motion control scheme via a set of real-time experiments employing a small underwater robotic vehicle. In particular, Subsection 3.5.1 introduces the experimental setup and Subsection 3.5.2 presents the detailed results of two cases of experimental studies with the proposed controller.

3.5.1 Setup

The experiments were carried out inside the *NTUA, Control Systems Lab* test tank, with dimensions $5m \times 3m \times 1.5m$ (Fig. 3.3). The bottom of the tank is covered by a custom-made poster with various visual features and markers. Two cylindrical objects with known position and dimensions are placed inside the tank and considered as static obstacles. The vehicle used in this chapter is a 4 DoFs Seabotix LBV, actuated in Surge, Sway, Heave and Yaw via a 4 thruster set configuration.The vehicle is equipped with a down-looking Sony PlayStation Eye camera, with 640×480 pixels at 30 frames per second (fps) enclosed in a waterproof housing. An underwater laser pointer projecting a green dot at the bottom of the test tank is rigidly attached on the vehicle with its

FIGURE 3.3: Experimental setup: The 4 DoFs Seabotix LBV inside of the NTUA, Control Systems Lab test tank including obstacles.

axes aligned to the down-looking camera axis. The vehicle is also equipped with an $SBG\ IG-500A$ AHRS, delivering temperature-compensated 3D acceleration, angular velocity and orientation measurements at $100Hz$. The marker localization system is based on the $ArUco$ library [190]. The complete state vector of the vehicle ($3D$ position, orientation, velocity) in the following experimental studies is available via the sensor fusion and state estimation module based on the Complementary Filter notion presented [191]. The vehicle's dynamic parameters have been previously identified via a proper identification scheme. The analysis of the sensor fusion, state estimation and parameter identification algorithms are out of the scope of this chapter and thus omitted. The software implementation of the proposed motion control scheme was conducted in C++ and Python under the Robot Operating System (ROS) [192]. The Constrained Nonlinear Model Predictive Controller employed in this chapter is designed using the NLopt Optimization library [193] and run on a desktop with 8 cores, 3.60 GHz CPU and 16 GB of RAM at $10Hz$.

The disturbances in the form of water currents, were induced using a $BTD150$ thruster properly mounted inside the water tank. The generated flow field (i.e., assumed ocean current profile), was computed using a GPU-enabled Computational fluid Dynamics (CFD) software [1] developed in the Parallel CFD and Optimization Unit of the school of Mechanical Engineering of NTUA. The flow field distribution inside the water tank is depicted in Fig-3.4.

3.5.2 Results

In order to prove the efficacy of the proposed controller, three experimental sessions are presented, namely Session A, B and C. In all experiments the objective is to follow a set of predefined way points, while simultaneously avoid two static obstacles and respect

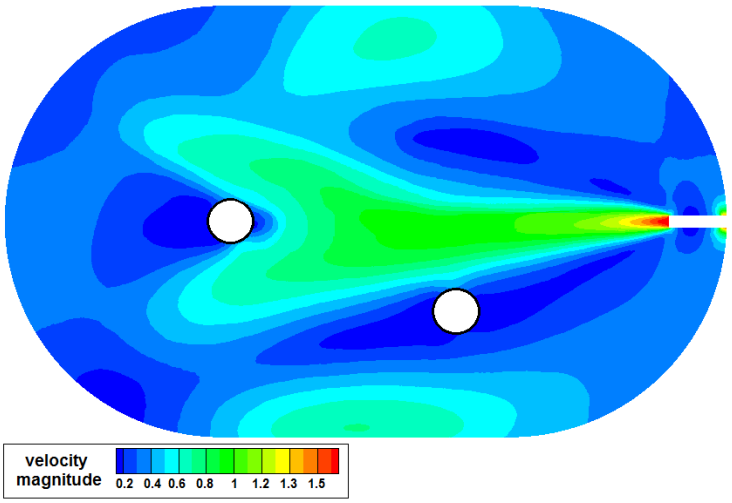
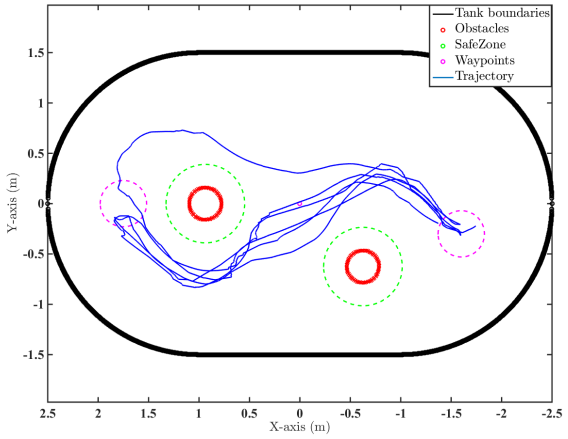


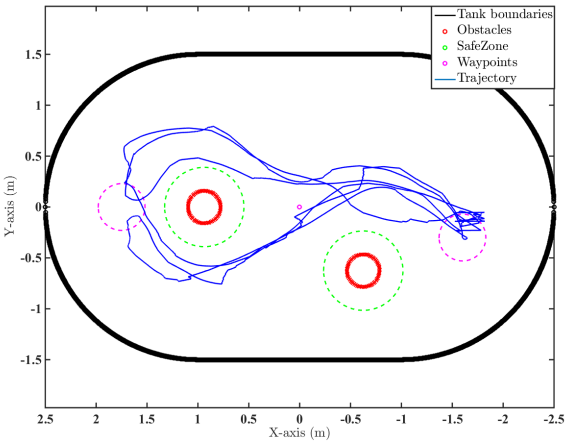
FIGURE 3.4: Distribution of the flow field inside the experimental water tank as computed by the CFD software presented in [1].

the workspace (test tank) boundaries. In Sessions A and B, we consider the dynamic model of vehicle in which the ocean currents are incorporated, hence the controller is aware about the presence of currents. In Sessions A and B, two different experiments were conducted for each case, namely A.1, A.2, B.1, B.2, demonstrating in this way the repeatability of the proposed control strategy. In Session C, the employed dynamic model of vehicle inside the OPC of NMPC, is not aware of the water currents induced by the thruster mounted inside the water tank. A comparative study is then presented, describing the performance of the underwater robot along with the consumed thrust, in both cases where the currents are known and unknown.

The location and geometry of the obstacles are considered known. More specific, the position of the obstacles w.r.t the Inertial Frame $\mathscr{I}$ in $x-y$ plane is given by: $\mathbf{x_{obs_1}} = \begin{bmatrix} -0.625 & -0.625 \end{bmatrix}$, $\mathbf{x_{obs_2}} = \begin{bmatrix} 0.9375 & 0 \end{bmatrix}$. The state constraints of the (3.9)-(3.12) which must be satisfied during all the experiments are analytically formulated as follows: i) The obstacles are cylinders (See Fig.3.3) with radius $r_{\pi_i} = 0.16m,\ i = \{1,2\}$ and are modeled together with the workspace boundaries according to the spherical world representations as consecutive spheres. ii) The radius of the sphere $\mathscr{B}(\boldsymbol{\eta}_1, \bar{r})$ which covers all the vehicle volume (i.e., main body and additional equipment) is defined as $\bar{r} = 0.3m$. However, for the clarity of presentation, we depict it as a safe zone around the obstacles where the vehicle center $\boldsymbol{\eta}_1$ (denoted by blue line, see Fig.3.5, Fig.3.10, Fig.3.15) should not violated it. iii) The vertical position must be between $0 < z < 1.2\ m$. iv) The vehicle's body velocity norm of (3.9) $|u_r + v_r|$ (planar motion) must not exceed $0.5m/s$. v) Heave velocity must be retained between $-0.25 < w_r < 0.25\ m/s$ (i.e., $0.25 - |w_r| \geq 0$). vi) Yaw velocity must be retained between $-1 < r_r < 1\ rad/s$ (i.e., $1 - |r_r| \geq 0$). Moreover, each of the four thrusters must obey the following input constraint: $-12 < \tau_i < 12N,\ i =$



( ) A.1 experiment



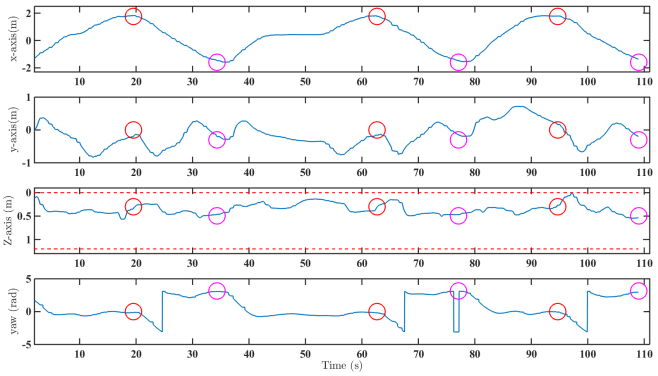
( ) A.2 experiment

FIGURE 3.5: Session A – 2 WP tracking scenario: Vehicle trajectory in horizontal plane

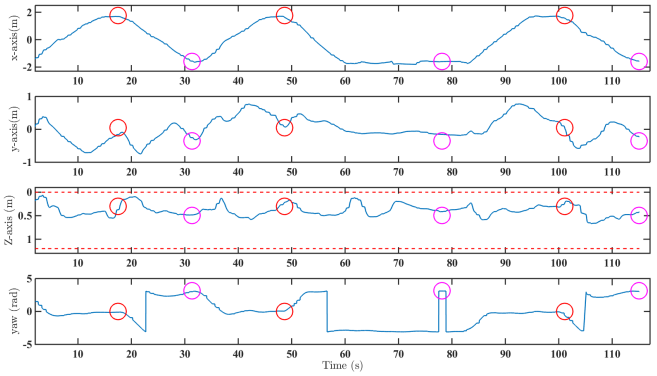
$\{p,\ s,\ v,\ l\}$. The state and input constraints in the following figures are depicted in red dashed lines were applicable. At this point we should mention that each mission is considered as successful only if the vehicle performs the way point tracking three consecutive times, hence repeatability is proved. In all times the vehicle is under the influence of the water currents depicted in Fig 3.4.

3.5.2.1 Session A – Two Way Points Tracking

In this scenario the vehicle must travel via two way points which are placed at $\boldsymbol{\eta}_1^d = \begin{bmatrix} -1.60\,m & -0.35\,m & 0.45\,m & 0\,rad \end{bmatrix}$, $\boldsymbol{\eta}_2^d = \begin{bmatrix} 1.75\,m & 0\,m & 0.30\,m & \pi\,rad \end{bmatrix}$ respectively. The three consecutive trajectories of the vehicle along the horizontal plane are depicted in Fig. 3.5 for each of the two experiments. It can be seen that the vehicle performs successfully the way point tracking while safely avoids the obstacles and the test tank boundaries. We observe that in some cases the vehicle may travel from one way point to the other following a different trajectory. This can be explained by the



( ) A.1 experiment



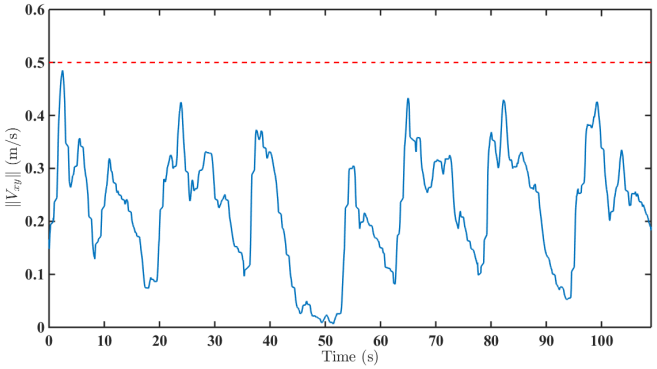
( ) A.2 experiment

FIGURE 3.6: Session A – 2 WP tracking scenario: Evolution of vehicle states

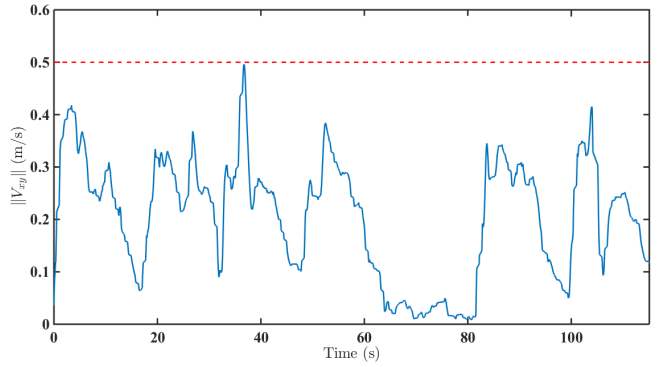
fact that the MPC finds a different optimal solution at the specific time frame, due to the unmodeled dynamics of the tether which significantly affect the vehicle motion. The vertical and angular motion of the vehicle are depicted in Fig. 3.6, where it can be seen that the state constraints are always satisfied. The vehicle is consider to reach each way point if it has entered a terminal region (i.e., spherical region of $0.3m$ and a offset of $\pm 0.15rad$) around the way point. These regions are depicted in circles in Fig. 3.5 and 3.6. In Fig. 3.7 the body velocity norm in planar motion is depicted and the respective constraint is satisfied. The same stands for the heave and yaw velocities, as shown in Fig. 3.8. In Fig. 3.9 the vehicle's thruster inputs are shown. As it can be seen the input constraints are also satisfied.

3.5.2.2 Session B – Three Way Points Tracking

This scenario is similar to the previous one except that the vehicle must travel along 3 way points which makes the mission more challenging considering the narrow workspace. The locations of the three way points are given by: $\boldsymbol{\eta}_1^d = \left[-1.50\,m, 0.30\,m, 0.40\,m, -\frac{\pi}{2}\,rad\right]$, $\boldsymbol{\eta}_2^d = [0.45\,m, -1\,m, 0.25\,m, 0\,rad]$, $\boldsymbol{\eta}_3^d = [1.20\,m,\ 1\,m, 0.30\,m, -\pi\,rad]$. Again for this scenario, two experiments were carried out. The three consecutive trajectories of the vehicle

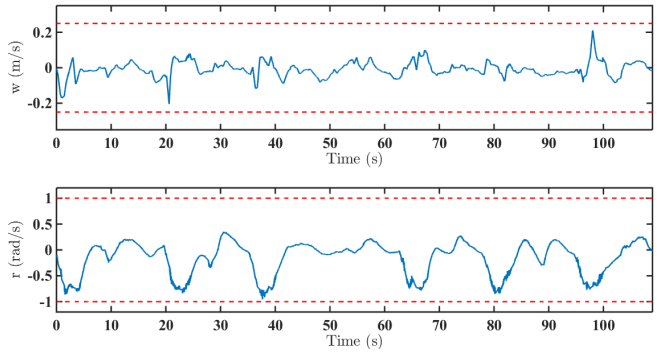


( ) A.1 experiment

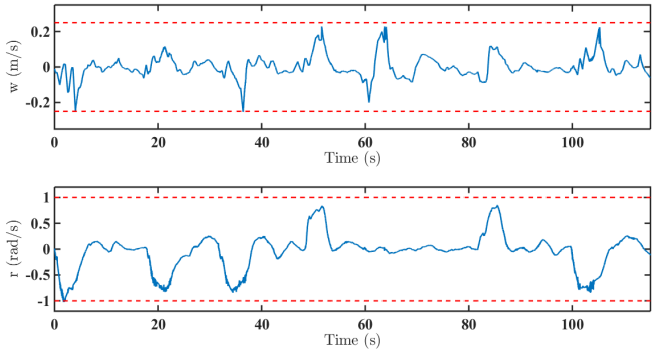


( ) A.2 experiment

FIGURE 3.7: Session A – 2 WP tracking scenario: Vehicle body velocity norm $|u_r+v_r|$

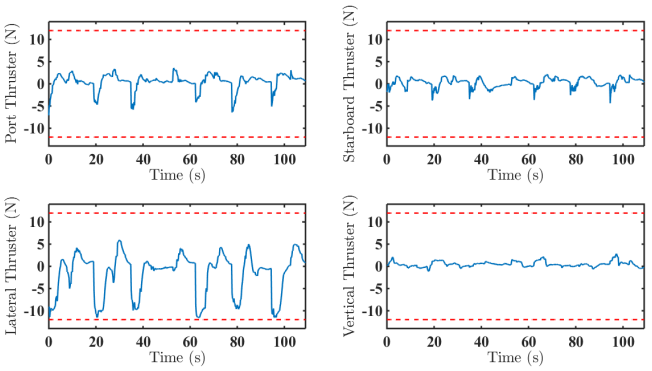


( ) A.1 experiment

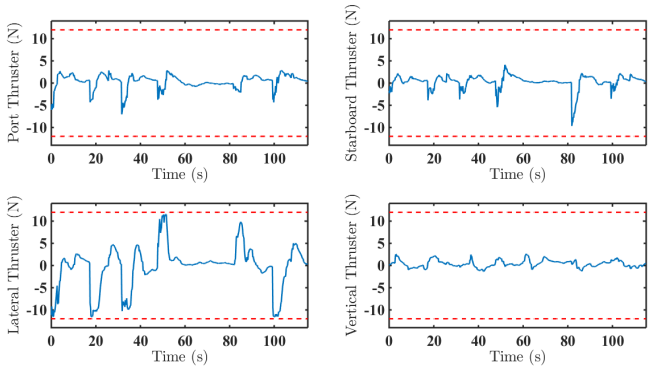


( ) A.2 experiment

FIGURE 3.8: Session A–2 WP tracking scenario: Vehicle heave and yaw velocities



( ) A.1 experiment



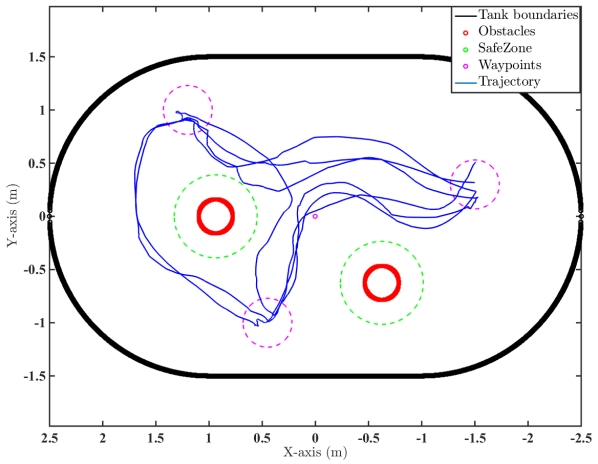
( ) A.2 experiment

FIGURE 3.9: Session A – 2 WP tracking scenario: Thruster Commands

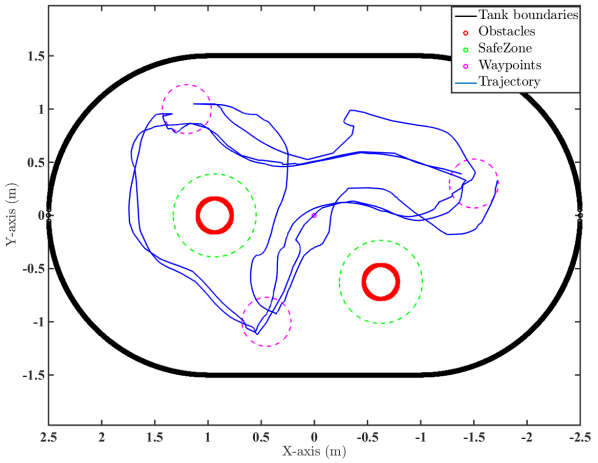
along the horizontal plane are depicted in Fig. 3.10. Although this scenario is more complicated, the vehicle again carries it out successfully. In this mission, the vehicle also follows different trajectories, for the same reasons explained in Session A. The vertical and angular motion are depicted in Fig. 3.11, while in Fig. 3.12 the body velocity norm in planar motion is shown. The heave and yaw velocities are presented in Fig. 3.13, while in Fig. 3.14 the vehicle's thruster inputs are shown. As it can be observed, the vehicle reached all desired way points while simultaneously satisfied all respective state and input constraints.

3.5.2.3 Session C – Comparative Experimental Results

In the following experiment, the vehicle must travel again along 3 way points, which is considered as the more challenging case. The location of the three way points is exactly the same as in Session B. However, the predictive controller considers the dynamic model of the vehicle in which the ocean currents are not incorporated. Hence, the employed dynamic model inside the OCP of the NMPC is not aware of the water currents induced by the thruster mounted inside the water tank. More precisely, instead of Eq. (3.2), we



( ) B.1 experiment



( ) B.2 experiment

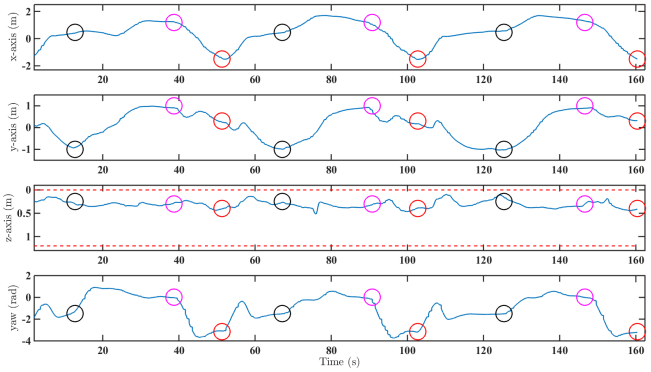
FIGURE 3.10: Session B – 3 WP tracking scenario: Vehicle trajectory in horizontal plane

employ the simple dynamic model, as given in [154, eq:2.172-2.173]:

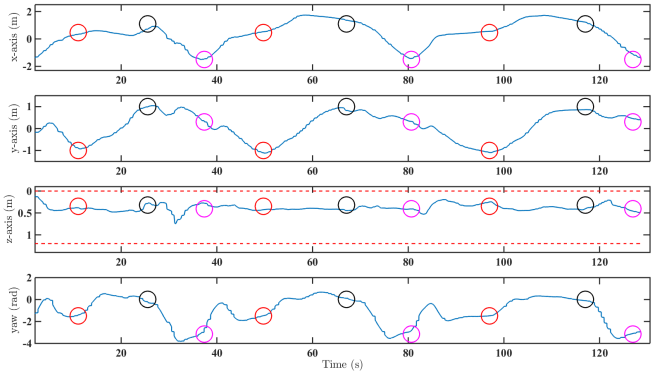
$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\,\boldsymbol{v} \tag{3.23}$$

$$\boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{v})\,\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}_{\mathscr{Y}} \tag{3.24}$$

The three consecutive trajectories of the vehicle along the horizontal plane are depicted in Fig.3.15. It can be seen that the vehicle follows different trajectories each time. In addition to the reasons explained in Session A, the water currents act as unmodeled and dynamic external disturbances. Comparing Fig.3.15 with Fig. 3.10, and taking into account the distribution of the flow field inside the water tank, as indicated in Fig.3.4, it can be observed that in the first case (Fig. 3.10) the robot has exploited the known water current dynamics during its way point tracking mission. On the contrary, when the water currents were not known to the system (Fig. 3.15), in one of the three consecutive trajectories, the vehicle traveled outside the area between the obstacles and close to the

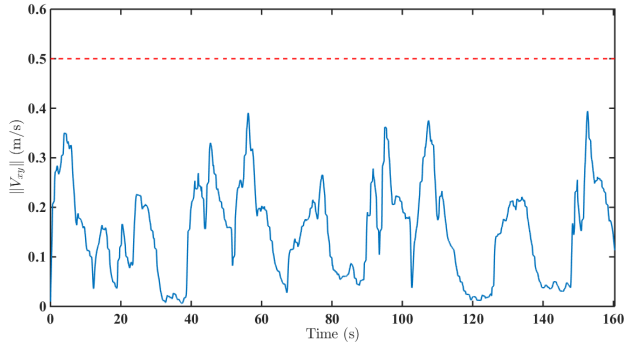


( ) B.1 experiment

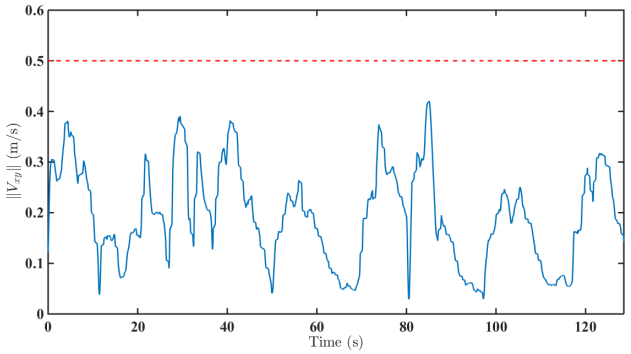


( ) B.2 experiment

FIGURE 3.11: Session B – 3 WP tracking scenario: Evolution of vehicle states

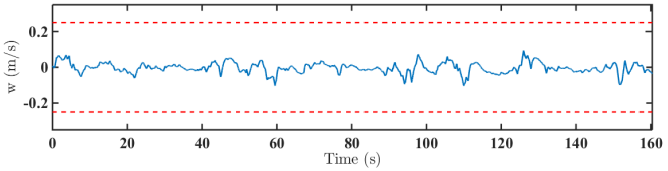


( ) B.1 experiment

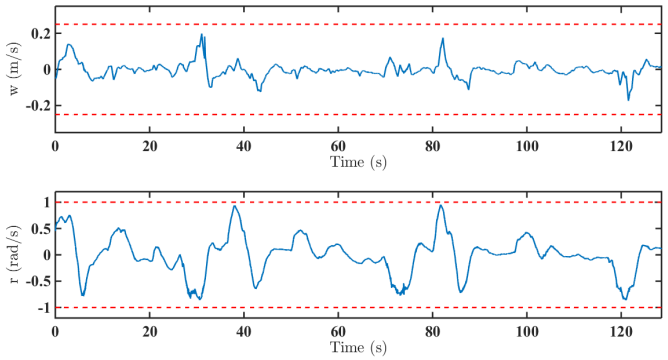


( ) B.2 experiment

FIGURE 3.12: Session B – 3 WP tracking scenario: Vehicle body velocity norm $|u_r + v_r|$

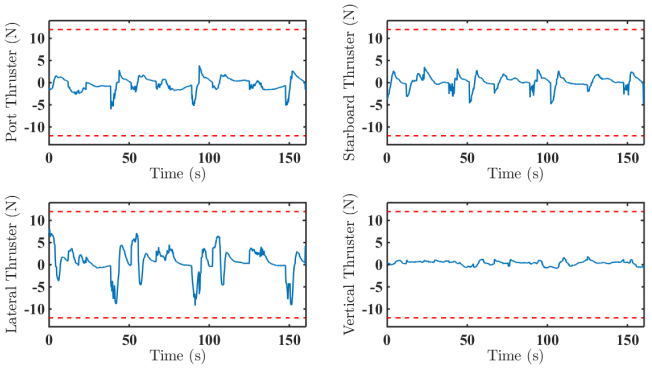


( ) B.1 experiment

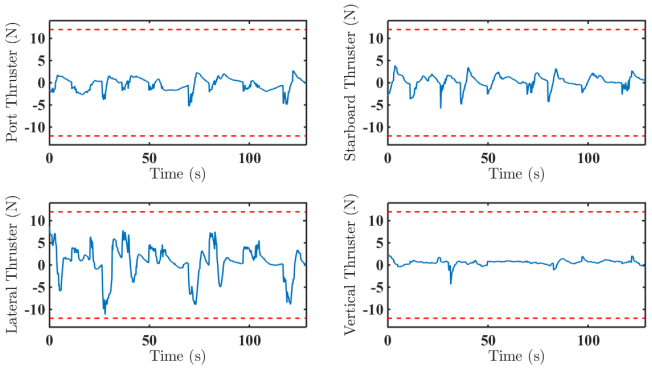


( ) B.2 experiment

FIGURE 3.13: Session B – 3 WP tracking scenario: Vehicle heave and yaw velocities



( ) B.1 experiment



( ) B.2 experiment

FIGURE 3.14: Session B – 3 WP tracking scenario: Thruster Commands

water tank boundary, while moving between the first way point to the second one. In Fig.3.16 the body velocity norm in planar motion is shown, where it can be observed that the predefined velocity constraint has been violated at least once.

The vehicle's thruster inputs are shown in Fig. 3.17. A comparison between the thrust consumption in Sessions B (known current profile) and C (unknown current profile) is given in Tables 3.1-3.2. It is shown that the consumed thrust in B.1 and B.2 experiments is significant less relative to Session C experiment. It is also worth noticing that B.2 experiment was completed in only 128.5 *sec* while the Session C experiment lasted 161.8 *sec*. We can also observe, that while B.1 experiment had almost the same time duration with C (i.e., 160.1 *sec*), a significant 15% reduction on the thrust consumption occurred. Hence, the proposed control scheme appears to be more efficient and optimal in terms of thrust consumption. According to Table 3.2, no significant difference in thrust consumption appears in the vertical direction. This is explained by the fact, that (for the sake of simplicity) we considered a $2D$ (i.e., in horizontal plane) distribution of the flow field inside the water tank and accordingly was computed the flow by the CFD software [1]. However, it is expected that more convincing results can be achieved if a more realistic model of the water current (i.e., $3D$ space) is employed within the dynamic model of the proposed predictive controller.

TABLE 3.1: Thrust consumption Comparison

Thruster	Thrust absolute value (N)		
	B.1 Exp	B.2 Exp	C Exp
Port	1161.5	946.7	1329.5
Starboard	954.0	846.4	1328.7
Lateral	1957.0	1942.5	2249.2
Vertical	940.3	925.5	1001.8
Total	5012.8	4661.1	5909.2
Total Reduction w.r.t C Exp	-15%	-21%	

3.5.3 Video

The aforementioned experimental study is demonstrated in a HD video at the following url: https://youtu.be/z04ELMfCTYk.
Alternatively, it can be found in the attached dvd as file: video_ch3.mpg

TABLE 3.2: Exploitation per thruster

Thruster	Thrust Reduction w.r.t to Session C Experiment	
	B.1 Exp	B.2 Exp
Port	-12.6%	-28.8%
Starboard	-28.2%	-36.3%
Lateral	-12.9%	-13.6%
Vertical	-6.1%	-7.6%

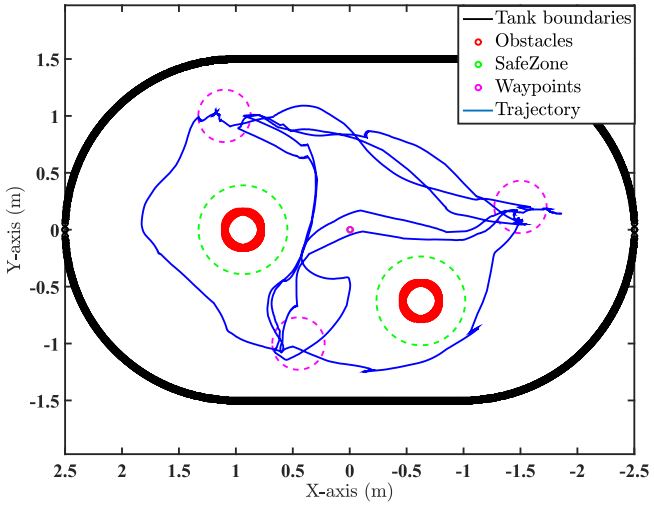


FIGURE 3.15: Session C – Comparative scenario: Vehicle trajectory in horizontal plane

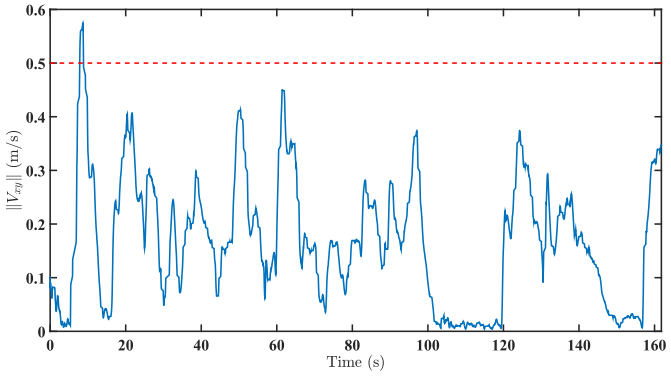


FIGURE 3.16: Session C – Comparative scenario: Vehicle body velocity norm $|u_r + v_r|$

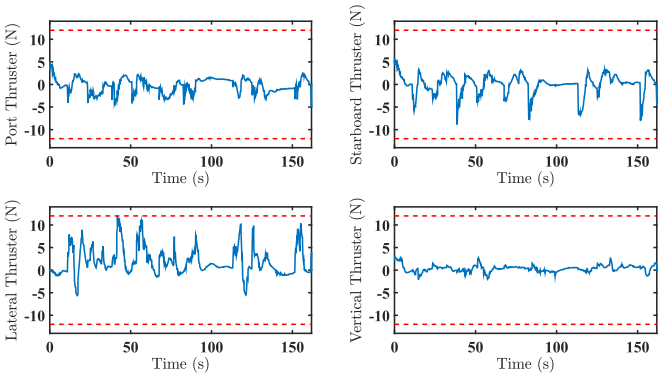


FIGURE 3.17: Session C – Comparative scenario: Thruster Commands

3.6 Conclusion

In this chapter, we presented a novel Model predictive Control strategy for underwater robotic vehicles operating in a constrained workspace including obstacles. The purpose of this control scheme is to guide the vehicle towards specific way points. Various constraints such as: obstacles, workspace boundaries, thruster saturation and predefined upper bound of the vehicle velocity (requirements for various underwater tasks such as seabed inspection, mosaicking etc.) are considered during the control design. Moreover, the proposed control scheme incorporates the dynamics of the vehicle and is designed in order to find optimal thrusts required for minimizing the way point tracking error. The control inputs calculated by the proposed scheme, may exploit the ocean currents when these are in favor of the way point tracking mission, which results in retaining the energy consumed by the thrusters in a reduced level. The efficacy of the proposed controller is experimentally verified using a 4 Degrees of Freedom (DoF) underwater robotic vehicle inside a constrained test tank with sparse static obstacles.

Part IV - Visual Servoing

Chapter 4

A Self-triggered Position Based Visual Servoing for Underwater Robotic Vehicles

In this chapter, we propose a Position Based Visual Servoing (PBVS) scheme for case when the relative position between the robot and object can be obtained and estimated online (e.g., by recognition of a known marker located on the object of interest). The purpose of the proposed controller in this chapter is to navigate and stabilize the camera (located at the vehicle body or at the end effector frame of the UVMS) towards a visual target and assuring that the target will always remain inside the camera's field of view. The proposed controller in this chapter is design at the camera frame. Therefore, the supposed vision system could be located either at the end effector or at some point on the underwater robotic vehicle's body. However, in the following experiments, we assume that the camera system is located at some point on the vehicle's body. This is motivated by the fact that in case when the Cartesian coordinates of the object with respect to the camera frame (e.g., located at some point on the vehicle's body) is estimated online (i.e., by employing computer vision algorithms to recognize a known marker), then it is easy to calculate the relative position of the final operator relative to the point of interest (e.g., using forward kinematic of the robotic chain, see Chapter 2 for more details). This allows the end effector to reach very close relative the object (e.g., it can touch or grasp the object), as long as the marker remains inside of the image plane (e.g., it is possible to estimated the relative position). Therefore, in this chapter we assume that the camera system is located at the vehicle's body. More specifically, the vehicle used in this work is 3 DoF underwater vehicle which is underactuated in Sway direction (See Section 2.1). Notice that owing to: i) underactuation in Sway direction, ii) visual constraints (i.e., avoiding the object to leave the camera's field of view), iii) input constraints, and iv)

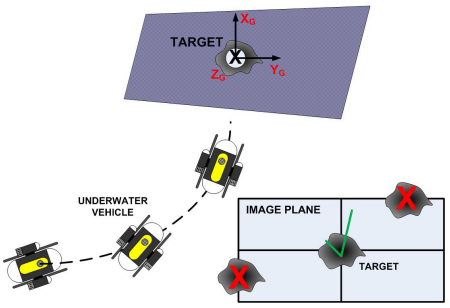


FIGURE 4.1: Problem Statement: Navigation and stabilization of the vehicle with respect to a visual target while maintaining target inside camera's optical field.

energy consumption and cpu efforts issues, the control problem becomes very challenging, with a few previously reported results in the related literature.

4.1 Introduction

Generally, the control of an underwater vehicle is generally a highly nonlinear problem. Moreover, real applications (e.g., inspection and surveillance tasks) impose constraints on the inputs and the states of the vehicle (i.e. visual and/or kinematic constraints), thus the underlying problem is constrained. Nonlinear Model Predictive Control (NMPC) [58], is an ideal approach to be used in Visual Servoing and Robotics due to its strong and efficient ability to handle input and state constraints. In [107] a Visual Predictive Control scheme has been proposed where the visibility constraints are formulated as state constraints. Some applications of IBVS-Model Predictive Control for navigation of unmanned aerial vehicles (UAVs), mobile robots, as well as some medical applications are presented in [110], [111] and in [113], respectively. A vision based terrain mapping-model predictive control approach for autonomous landing of an UAV in unknown terrain is given in [194]. In [195] a vision based approach for path following of an omni-directional mobile robot using MPC is presented.

A standard visual servoing scheme consists the periodic (at every sampling period) use of the vision feedback which is extracted from the image to generate a task error and using a control algorithm to minimize this error. The process of image feature extraction, matching with a desired image and using them to generate the task error, is usually referred in the literature as *Visual Tracking* [68]. The main concerns of a Visual Tracking Algorithm (VTA) in a complex environment are accuracy and robustness. However, it is known that the accurate and robust VTA in real-time robotic applications is considered to be very heavy process with high computational cost. This would result in large energy usage and may cause delays in the closed-loop system. This problem becomes more

apparent when small autonomous robotic systems are considered such as AUVs which suffer from limited energy resources (batteries) and which are usually equipped with small and not so powerful embedded computing units. Long lasting inspection tasks in complex environments require accurate VTA as well as high autonomy. The problem then becomes more evident, because the continuous recharging procedure is undesirable, difficult and time consuming. On the other hand, NMPC schemes constitute in solving a constrained Optimal Control Problem (OCP) at each time instant and they are considered to be computationally demanding. In addition, these systems are usually equipped with weak computing units that need to solve the VTA and the OCP of the NMPC at each time instant. This leads to bigger sampling periods on the closed loop system that consequently reduce the accuracy of the system.

Is it possible to relax the control update and visual tracking while the performance of the visual servoing scheme remains the same? In other words, is it possible to design a visual servoing scheme that decides when the robot needs to track the visual information, update the control input and when not, while the whole system does not lose the required performance? This question, motivates the self-triggered design framework for Visual Servoing in order to track the vision information and compute the control law only when it is needed.

4.1.1 The Self-triggered Control Framework

The standard control framework that is used today is the periodic control which is presented in the textbooks as the only choice for implementing feedback control laws on digital platforms. Quite recently though, a novel formulation of control schemes in a self-triggered manner is becoming popular. In self-triggered control the key attribute is that the decision for the execution of the control task is not made ad-hoc as in the sampled-data case, but it takes into account state or output feedback in order to sample as infrequently as possible while guaranteeing to preserve the stability of the system, see Fig.4.2. This approach results to a more flexible aperiodic sampling, while preserving necessary properties of the system such as stability and convergence. Moreover, self-triggered control can be proven to be less conservative with respect to the constant sampling where the worst case scenario is considered. In particular, the self-triggered methodology may lead to an overall reduction on the number of the control updates, a feature that might be desirable for many systems that have resource constraints and many applications that have limitations on their sensor, control computation and communication capabilities.

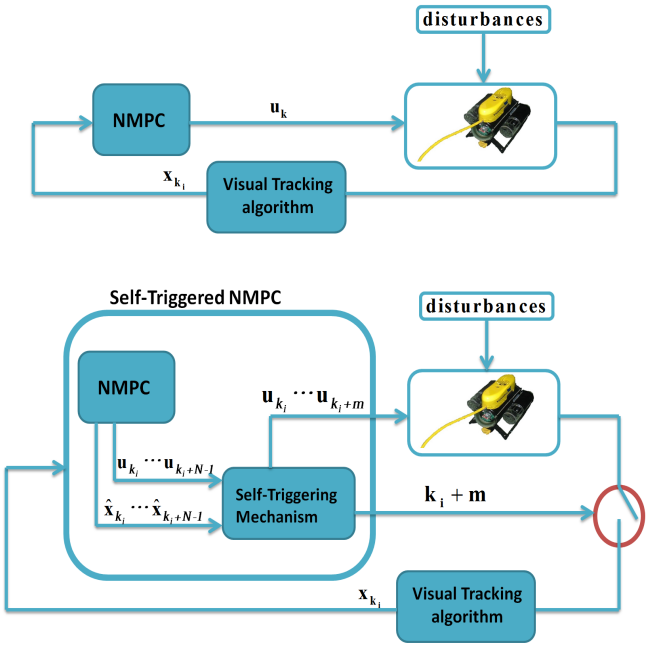


FIGURE 4.2: The classic periodic time-triggered framework is depicted in the top block diagram. The bottom diagram represents the self-triggered control.

The self-triggered control framework along with a closely related framework named event-triggered control comprise the recently introduced event-based control framework. Both approaches, self and event-based control, are comprised, inter alia, by a feedback controller that computes the control input and a triggering mechanism that determines when the new control update should occur. However, these frameworks are different; the event-triggered control can be considered to be reactive as it computes the control input when the plant state deviates more than a certain threshold from a desired value, while the self-triggered control, can be considered to be proactive as it computes the next control input ahead of time. Notice, that the event-triggered techniques require a constant measurement of the state of the plant in order to decide when the control update must be triggered. However, in the case of self-triggered control only the latest state measurement needs to be known for determining the next triggering instant. An introductory paper on event-based control is [196]. More on the event-triggered control can be found in [197], [198], [199], [200] and the papers quoted therein. The most recent developments on self-triggered control can be found in [201], [202], [203], [204].

4.1.2 Contributions

In this chapter, a Self-triggered Position Based Visual Servo NMPC scheme for the motion control of an underwater vehicle is presented when the relative position between

the Camera (i.e., located at some point on the vehicle's body) and the target of interest is available online. The purpose of this work is to navigate and stabilize the vehicle towards a visual target by using the proposed control framework and assuring that the target will always remain inside the camera's field of view (Fig. 4.1). The localization of the vehicle with respect to the visual target is calculated using Computer Vision algorithms and it is described in more detail in Section 4.4.1. The choice of PBVS instead of IBVS or 2-1/2 visual servoing, is mainly motivated by the inherited advantage of PBVS to control the onboard camera and as a result the vehicle itself directly in the Cartesian space. This gives us the ability to design more easily the NMPC scheme. The fact that PBVS cannot guarantee the preservation of the visual target inside the image frame, is handled by the explicit formulation of the visual constraints inside the NMPC framework. Another contribution of the control scheme presented this chapter relies in the design of a framework that will provide aperiodic control sequences that lead to stable closed-loop responses and of a mechanism that will decide when these control updates and next VTA should occur. This results to the reduction of the computational effort, processing of vision data and energy consumption. Thus the accuracy and autonomy of the robotic system can be increased. These are of utmost importance in the case of small autonomous robotic systems and especially AUVs which perform persistent inspection tasks. In this case the lean computational algorithms, low energy consumption, efficient data processing, high accuracy and rate of persistent autonomy are required. Some results is the continuous-time framework was given in [205] and a preliminary version of the results presented in this chapter was reported in [206].

Experimental results on event-based formulations are scarce in the literature [207], [208], [209], [210], [211]. In this chapter, the performance of the proposed control scheme is experimentally verified using a small under-actuated underwater vehicle in a test tank. Moreover, the experimental results are quite satisfying as the vehicle's state vector converges to a bounded set around the desired position and the triggering instants are significantly reduced with respect to the traditional time-triggered case.

The remainder of the chapter is organized as follows: In Section 4.2, the modeling of the underwater vehicle with the imposed constraints and uncertainties as well as the whole problem statement are presented. Section 4.3 accommodates the robust stability analysis for the NMPC scheme which leads to the self-triggered framework. Section 4.4, illustrates the efficiency of the proposed approach through a set of experimental results. Finally, Section 4.5 concludes the chapter.

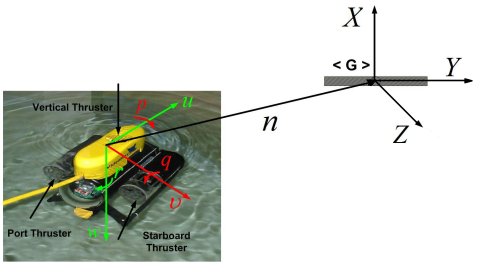


FIGURE 4.3: Differential Drive Robot. Red color indicates no actuation availability, while green color indicates actuation availability along body frame axes.

4.2 Problem Formulation

In this section, the overall problem is formulated. Initially, the nominal model of the under-actuated underwater vehicle and the constraints that must be fulfilled are presented. Next, we consider additive perturbations in order to describe the actual system. Finally, the proposed motion control scheme is designed.

4.2.1 Mathematical Modeling

As it is already explained in Section 2.1, generally an underwater vehicle is considered as a 6 DOF free body with position and Euler angle vector $\boldsymbol{x} = [\chi\ y\ z\ \phi\ \theta\ \psi]^\top$. The body velocities vector is defined as $\boldsymbol{v} = [u\ v\ w\ p\ q\ r]^\top$ where the components have been named according to SNAME [212] as surge, sway, heave, roll, pitch and yaw respectively (Fig.4.3). The forces and moments vector acting on the body-fixed frame are defined as $\tau = [X\ Y\ Z\ K\ M\ N]^\top$. The general form of the dynamics of an underwater vehicle expressed in the body-fixed frame are given in matrix form by the equations below (see Section 2.1):

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{J}(\boldsymbol{x})\boldsymbol{v} \\ \boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{x}) &= \boldsymbol{\tau} \end{aligned} \tag{4.1}$$

where: $\mathbf{M} = \mathbf{M_{RB}} + \mathbf{M_A}$ is the inertia matrix for rigid body and added mass respectively, $\mathbf{C}(\mathbf{v}) = \mathbf{C_{RB}}(\mathbf{v}) + \mathbf{C_A}(\mathbf{v})$ is the Coriolis and centripetal matrix for rigid body and added mass respectively, $\mathbf{D}(\mathbf{v}) = \mathbf{D_{quad}}(\mathbf{v}) + \mathbf{D_{lin}}(\mathbf{v})$ is the quadratic and linear drag matrix respectively, $\mathbf{g}(\mathbf{x})$ is the hydrostatic restoring force vector, $\boldsymbol{\tau}$, is the thruster input vector and $\mathbf{J}(\mathbf{x})$ is the Jacobian matrix transforming the velocities from the body-fixed to earth-fixed frame. The vehicle used in this chapter is a 3 DOF VideoRay Pro ROV. It is equipped with three thrusters, which are effective only in surge, heave and yaw motion (Fig.4.3), meaning that the vehicle is under-actuated along the sway axis. The angles ϕ, θ and angular velocities p and q are negligible and we can consider them to be equal to zero. The ROV is symmetric about x - z plane and close to symmetric about y - z plane. Therefore, we can safely assume that motions in heave, roll and pitch are

decoupled [154]. However we will be considering the coupling between surge, sway and yaw that will be affecting the surge and sway motions since this is important for our task. Because the vehicle is operating at relative low speeds, the coupling effects are considered to be negligible. Due to the above assumptions and the relative low speed of the vehicle, we consider the kinematic model of the vehicle[1], which in its discrete-time form is formulated as follows:

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{v}_k) + \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{v}_k) \Rightarrow$$

$$\begin{bmatrix} \chi_{k+1} \\ y_{k+1} \\ z_{k+1} \\ \psi_{k+1} \end{bmatrix} = \begin{bmatrix} \chi_k \\ y_k \\ z_k \\ \psi_k \end{bmatrix} + \begin{bmatrix} \cos\psi_k & 0 & 0 \\ \sin\psi_k & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_k \\ w_k \\ r_k \end{bmatrix} dt + \begin{bmatrix} -\sin\psi_k \\ \cos\psi_k \\ 0 \\ 0 \end{bmatrix} \boldsymbol{v}_k dt \tag{4.2}$$

where $\boldsymbol{x}_k = [\chi_k, y_k, z_k, \psi_k]^\top$ denotes the state vector at the time-step k which includes the position and orientation of the vehicle with respect to the target frame $\mathcal{G}$. The control input of the system is $\boldsymbol{v}_k = [u_k, w_k, r_k]^\top$ and dt denotes the sampling period.

In [213], using Input-to-State Stability (ISS) framework, it was shown that for any vehicle described by (4.2) and for any bounded control input $[u_k, r_k]$ the velocity about the sway direction $\boldsymbol{v}_k$ can be seen as a bounded perturbation with upper bound $||\boldsymbol{v}_k|| \leq \bar{v}$ that vanishes at the point $x = 0$. Consequently, this point is an equilibrium of the kinematic system of (4.2). Note that the notation $(\bar{\cdot})$ will denote the upper bound for each of the variables. Therefore we consider the system:

$$\begin{aligned} &\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{v}_k) \Rightarrow \\ &\begin{bmatrix} \chi_{k+1} \\ y_{k+1} \\ z_{k+1} \\ \psi_{k+1} \end{bmatrix} = \begin{bmatrix} \chi_k \\ y_k \\ z_k \\ \psi_k \end{bmatrix} + \begin{bmatrix} \cos\psi_k & 0 & 0 \\ \sin\psi_k & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_k \\ w_k \\ r_k \end{bmatrix} dt \end{aligned} \tag{4.3}$$

as the nominal kinematic system of the underwater vehicle. The function $g(\cdot)$ is considered as a bounded inner disturbance of the system that vanishes at the origin. Also, $g(\boldsymbol{x}_k, \boldsymbol{v}_k) \in \Gamma \subset \Re^4$ with Γ being a compact set, such that:

$$||g(\boldsymbol{x}_k, \boldsymbol{v}_k)|| \leq \bar{\gamma} \quad \text{with} \quad \bar{\gamma} \triangleq \bar{v} dt \tag{4.4}$$

[1]Notice that the camera is rigidly located on the vehicle's body. Thus, estimating the pose of the camera leads easily into estimating the vehicle position relative to the marker. For simplicity, we have assumed that the camera is located close to vehicle's center of mass.

The robot moves under the influence of an irrotational current which behaves as an external disturbance. The current has components with respect to the χ, y and z axes, denoted by δ_χ, δ_y and δ_z, respectively. Also it has a slowly-varying velocity δ_c which is upper bounded by $||\delta_{c_k}|| \leq \bar{\delta}_c$ and it has direction β in $\chi - y$ plane and α with respect to the z axis of the global frame, see Fig.4.4. In particular we set $\boldsymbol{\delta}_k = [\delta_{(\chi/k)}, \delta_{(y/k)}, \delta_{(z/k)}, 0]^\top \in \Delta \subset \Re^4$ with Δ being a compact set, where:

$$\begin{aligned} \delta_{(\chi/k)} &\triangleq \delta_{c_k} \cos\beta_k \sin\alpha_k dt \\ \delta_{(y/k)} &\triangleq \delta_{c_k} \sin\beta_k \sin\alpha_k dt \\ \delta_{(z/k)} &\triangleq \delta_{c_k} \cos\alpha_k dt \end{aligned} \tag{4.5}$$

It is easy to show that $||\delta_k|| \leq \bar{\delta}$, with $\bar{\delta} = \bar{\delta}_c\ dt$. Taking into consideration the aforementioned disturbances that affect the vehicle, we are now ready to model the perturbed system as follows:

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{v}_k) + \boldsymbol{\omega}_k \tag{4.6}$$

with $\boldsymbol{\omega}_k = g(\boldsymbol{x}_k, \boldsymbol{v}_k) + \boldsymbol{\delta}_k \in \Omega \subset \Re^4$ as the result of adding the inner and external disturbances of the system. Ω is a compact set such that $\Omega = \Delta \oplus \Gamma$, where "$\oplus$" denotes the Minkowski addition of two sets. The definition for Minkowski addition set C of two sets $A, B \subset \Re^n$ is $C = A \oplus B = \{a + b | a \in A, b \in B\}$. Since the sets Δ and Γ are compact, we have that Ω is also a compact set, bounded by $||\boldsymbol{\omega}_k|| \leq \bar{\omega}$ with $\bar{\omega} \triangleq \bar{\delta} + \bar{\gamma}$.

The robot is equipped by a vision system that includes a pinhole camera with limited angles of view **a** and **b** for $\chi - y$ and $\chi - z$ plane respectively. Also a vision algorithm provides the state vector $\boldsymbol{x}$ of the system, with respect to the known-target frame $\mathcal{G}$, see Fig.4.4. The requirements for the vision system, namely the visibility constraints,

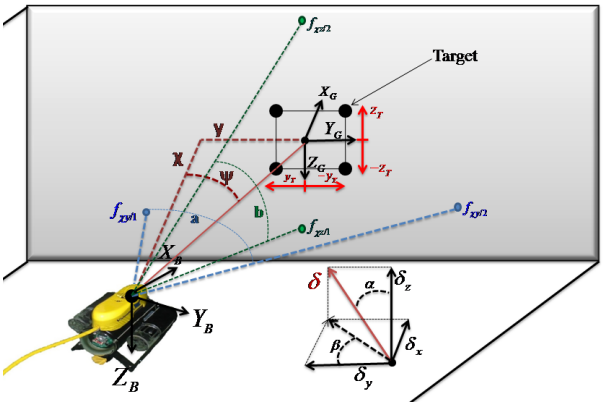


FIGURE 4.4: Modeling of the state constraints (4.9)-(4.12) imposed by the sensor system and modeling of the external disturbance (4.5).

are imposed in order to ensure that the target will not leave the image-plane during the control operation. That is: $[-y_T, y_T] \subseteq [f_{\chi y/1}, f_{\chi y/2}]$ and $[-z_T, z_T] \subseteq [f_{\chi z/1}, f_{\chi z/2}]$, where $2y_T$ and $2z_T$ are the width and height of the target respectively. Also $[f_{\chi y/1}, f_{\chi y/2}]$ and $[f_{\chi z/1}, f_{\chi z/2}]$ are the camera's field-of-view on $\chi - y$ and $\chi - z$ plane respectively (Fig. 4.4). Moreover, the vehicle should not exceed the maximum distance R_{max} at which the target is visible and recognizable to the vision. These requirements are captured by the state constraint set X of the system, given by:

$$\boldsymbol{x}_k \in X \subset \Re^4 \tag{4.7}$$

which is formed by the following constraints:

$$-y + \chi \tan(\psi - \frac{\mathbf{a}}{2}) - y_T \geq 0 \tag{4.8}$$

$$y - \chi \tan(\psi + \frac{\mathbf{a}}{2}) - y_T \geq 0 \tag{4.9}$$

$$-z - \chi \tan(\frac{\mathbf{b}}{2}) - z_T \geq 0 \tag{4.10}$$

$$z - \chi \tan(\frac{\mathbf{b}}{2}) - z_T \geq 0 \tag{4.11}$$

$$R_{\max}^2 - \chi^2 - y^2 \geq 0 \tag{4.12}$$

Furthermore, the control constraint set V is assumed to be compact and is given by:

$$\boldsymbol{v}_k \triangleq [u_k, w_k, r_k] \in V \subseteq \Re^3 \tag{4.13}$$

The constraints of the input are of the form $|u| \leq \bar{u}$, $|w| \leq \bar{w}$ and $|r| \leq \bar{r}$, therefore we get $\|\boldsymbol{v}_k\| \leq \bar{V}$ where $\bar{V} = (\bar{u}^2 + \bar{w}^2 + \bar{r}^2)^{\frac{1}{2}}$ and $\bar{V}, \bar{u}, \bar{w}, \bar{r} \in \Re_{\geq 0}$. It can be shown that system (4.3) is Lipschitz continuous, specifically:

Lemma 4.1. *The nominal model* (4.3)*, subject to constraints* (4.9)-(4.12) *and* (4.13)*, is locally Lipschitz in* $\boldsymbol{x}$ *for all* $\boldsymbol{x} \in X$*, with a Lipschitz constant* $L_f \triangleq (\max\{8, 8(\bar{u}dt)^2\} + 1)^{\frac{1}{2}}$*. See* Appendix *11.4 for the proof.*

4.2.2 Control Design and Objective

The control objective is to drive the actual system (4.6) to a desired compact set that includes the desired state $\boldsymbol{x}_d \triangleq [\chi_d, y_d, z_d, \psi_d]^\top \in X$, while respecting the state constraints (4.9)-(4.12) as well as the control constraints (4.13). A predictive controller is employed in order to achieve this task. In particular, at a given time instant k, the NMPC consists in solving an Optimal Control Problem (OCP) with respect to a control sequence $\boldsymbol{v}_f(k) \triangleq [\boldsymbol{v}(k|k), \boldsymbol{v}(k+1|k), \ldots, \boldsymbol{v}(k+N-1|k)]$, for a prediction horizon N. The OCP

of the NMPC is given as follows:

$$\min_{\boldsymbol{v}_f(k)} J_N(\boldsymbol{x}_k, \boldsymbol{v}_f(k)) = \tag{4.14}$$
$$\min_{\boldsymbol{v}_f(k)} \sum_{j=0}^{N-1} F(\hat{\boldsymbol{x}}(k+j|k), \boldsymbol{v}(k+j|k)) + E(\hat{\boldsymbol{x}}(k+N|k))$$

subject to:

$$\hat{\boldsymbol{x}}(k+j|k) \in X_j, \qquad \forall j = 1, \ldots, N-1, \tag{4.15}$$

$$\boldsymbol{v}(k+j|k) \in V, \qquad \forall j = 0, \ldots, N-1, \tag{4.16}$$

$$\hat{\boldsymbol{x}}(k+N|k) \in \mathscr{E}_f \tag{4.17}$$

where $\mathscr{E}_f$ is the terminal set and F and E are the running and terminal cost function, respectively. At time instant k, the solution of the OCP (4.14)-(4.17) is providing an optimal control sequence, denoted as $\boldsymbol{v}_f^*(\cdot)$. It should be pointed out that the specifics for the design parameters, such as the running and terminal costs, as well as the state sets will be provided in more detail in the sequel. The vector $\hat{\boldsymbol{x}}(k+j|k)$ denotes the predicted state of the nominal system (4.3) at sampling time $k+j$ with $j \in \mathbb{Z}_{\geq 0}$. The predicted state is based on the measurement of the state $\boldsymbol{x}_k$ of the actual system at sampling time k, while applying a sequence of control inputs $[\boldsymbol{v}(k|k), \boldsymbol{v}(k+1|k), \ldots, \boldsymbol{v}(k+j-1|k)]$. Thus:

$$\hat{\boldsymbol{x}}(k+j|k) = f(\hat{\boldsymbol{x}}(k+j-1|k), \boldsymbol{v}(k+j-1|k))$$

It holds that $\hat{\boldsymbol{x}}(k|k) = \boldsymbol{x}_k$. Notice, here that the OCP is solved for the nominal system, i.e., the model of the system that is not affected by disturbances. This is the case, due to the fact that the OCP is solved for a prediction horizon, thus, it is not possible to address the disturbances beforehand. However, we distinguish the nominal system that will be denoted as $\hat{\boldsymbol{x}}(\cdot)$ with the actual system, i.e., the system that is affected by disturbances that will be denoted as $x(\cdot)$ and we obtain the following preliminary result:

Lemma 4.2. *The difference between the actual state $\boldsymbol{x}_{k+j}$ at the time-step $k+j$ and the predicted state $\hat{\boldsymbol{x}}(k+j|k)$ at the same time-step, under the same control sequence, is upper bounded by:*

$$||\boldsymbol{x}_{k+j} - \hat{\boldsymbol{x}}(k+j|k)|| \leq \sum_{i=0}^{j-1} (L_f)^i \bar{w} \tag{4.18}$$

The proof is similar to the Appendix-11.1.

In Lemma 4.2, the difference between the actual state trajectory of system (4.6) and the predicted state trajectory of the nominal system (4.3), is obtained. To address this difference, in (4.15) we use a restricted constraint set $X_j \subseteq X$ instead of the state constraint set X. More details for this constraint tightening technique are presented in

[189] and [186]. Using this technique, it can be guaranteed that the evolution of the state of the perturbed system (4.6), when the control sequence developed from the NMPC Problem of (4.14)-(4.17) is applied to it, will necessarily satisfy the state constraint set X. More specifically, we denote the restricted constraint set as $X_j = X \sim \mathscr{B}_j$ where $\mathscr{B}_j = \{\boldsymbol{x} \in \Re^4 : ||\boldsymbol{x}|| \leq \sum_{i=0}^{j-1}(L_f)^i \bar{w}\}$. The set operator " $\sim$ " denotes the Pontryagin difference $A, B \subset \Re^n$ that is defined as the set $C = A \sim B = \{\zeta \in \Re^n : \zeta + \xi \in A, \forall \xi \in B\}$. Let the cost function $F(\cdot)$, as well as the terminal cost $E(\cdot)$, to both be of quadratic form, i.e., $F(\hat{\boldsymbol{x}}, \boldsymbol{v}) = \hat{\boldsymbol{x}}^\top \boldsymbol{Q}\hat{\boldsymbol{x}} + \boldsymbol{v}^\top \boldsymbol{R}\boldsymbol{v}$ and $E(\hat{\boldsymbol{x}}) = \hat{\boldsymbol{x}}^\top \boldsymbol{P}\hat{\boldsymbol{x}}$, respectively, with $\boldsymbol{P}$, $\boldsymbol{Q}$ and $\boldsymbol{R}$ being positive definite matrices. Particulary we define $\boldsymbol{Q} = diag\{q_1, q_2, q_3, q_4\}$, $\boldsymbol{R} = diag\{r_1, r_2, r_3\}$ and $\boldsymbol{P} = diag\{p_1, p_2, p_3, p_4\}$. For the running cost function F, we have $F(0, 0) = 0$, and we can also obtain the following:

Lemma 4.3. *For the cost function $F(\boldsymbol{x}, \boldsymbol{v})$ it holds that:*

$$F(\boldsymbol{x}, \boldsymbol{v}) \geq min(q_1, q_2, q_3, q_4, r_1, r_2, r_3)||[\boldsymbol{x}, \boldsymbol{v}]^\top||^2 \geq min(q_1, q_2, q_3, q_4, r_1, r_2, r_3)||\boldsymbol{x}||^2 \tag{4.19}$$

The state and input constraint sets are bounded, therefore it can be derived:

Lemma 4.4. *The cost function $F(\boldsymbol{x}, \boldsymbol{v})$ is Lipschitz continuous in $X \times V$, with a Lipschitz constant:*

$$L_F = 2(R_{max}^2 + z_{max}^2 + (\frac{\pi}{2})^2)^{\frac{1}{2}}\sigma_{max}(\boldsymbol{Q}) \tag{4.20}$$

where $\sigma_{max}(\boldsymbol{Q})$ is the largest singular value of the matrix $\boldsymbol{Q}$ and $z_{max} = R_{max}\tan(\frac{\mathrm{b}}{2}) - z_T$ is the maximum feasible value along the z axis. See Appendix *11.5 for the proof.*

In order to assert that the NMPC strategy results in a robust stabilizing controller we are going to employ some standard stability conditions that are used in MPC frameworks, as the following:

Assumption 4.1. For the nominal system (4.3), there is an admissible positively invariant set $\mathscr{E} \subset X$ such that the terminal region $\mathscr{E}_f \subset \mathscr{E}$, where $\mathscr{E} = \{\boldsymbol{x} \in X : ||\boldsymbol{x}|| \leq \varepsilon_0\}$ and ε_0 being a positive parameter.

Assumption 4.2. We assume that in the terminal set $\mathscr{E}_f$, there exists a local stabilizing controller $\boldsymbol{v}_k = h(\boldsymbol{x}_k) \in V$ for all $x \in \mathscr{E}$, and that E satisfies $E(f(\boldsymbol{x}_k, h(\boldsymbol{x}_k))) - E(\boldsymbol{x}_k) + F(\boldsymbol{x}_k, h(\boldsymbol{x}_k)) \leq 0$ for all $\boldsymbol{x} \in \mathscr{E}$.

Assumption 4.3. The terminal cost function E is Lipschitz in $\mathscr{E}$, with Lipschitz constant $L_E = 2\varepsilon_0\sigma_{max}(P)$ for all $\boldsymbol{x} \in \mathscr{E}$.

Assumption 4.4. Inside the set $\mathscr{E}$ we have $E(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{P}\boldsymbol{x} \leq \alpha_\varepsilon$, where $\alpha_\varepsilon = max\{p_1, p_2, p_3, p_4\}\varepsilon_0^2 > 0$. Assuming that $\mathscr{E} = \{\boldsymbol{x} \in \boldsymbol{x}_{(N-1)} : h(\boldsymbol{x}) \in V\}$ and taking a positive parameter α_{ε_f} such that $\alpha_{\varepsilon_f} \in (0, \alpha_\varepsilon)$, we assume that the terminal set designed as $\mathscr{E}_f = \{\boldsymbol{x} \in \Re^4 : E(\boldsymbol{x}) \leq \alpha_{\varepsilon_f}\}$ is such that $\forall \boldsymbol{x} \in \mathscr{E}$, $f(\boldsymbol{x}, h(\boldsymbol{x})) \in \mathscr{E}_f$.

4.2.3 Problem Statement

The solution of the OCP (4.14)-(4.17) at a time-step k provides an optimal control sequence $\boldsymbol{v}_f^*(\cdot)$ which equals to $\boldsymbol{v}_f^*(k) \triangleq [\boldsymbol{v}^*(k|k), \ldots, \boldsymbol{v}^*(k+N-1|k)]$. The classic framework of the MPC consists in applying to the system only the first control vector, i.e., $\boldsymbol{v}^*(k|k)$ and to discard all the remaining elements of the sequence. At the next time-step $k+1$, a new state measurement is received and the whole procedure is repeated again. This is iteratively repeated until the system has reached to the desired terminal set. However, the self-triggered framework suggests that a portion of the computed control sequence may be applied to the system and not only the first vector. Suppose k_i to be a triggering instant when the OCP of the MPC is solved. In the self-triggered case, the control sequence that is applied to the plant is of the form:

$$[\boldsymbol{v}^*(k_i|k_i), \boldsymbol{v}^*(k_i+1|k_i), \ldots \boldsymbol{v}^*(k_i+d_i|k_i)] \tag{4.21}$$

for all $d_i \in [0, k_{i+1} - k_i] \in \mathbb{Z}_{\geqslant 0}$, where k_{i+1} is the next triggering instant. During the time interval $[k_i, k_{i+1})$ the control law is applied to the plant in an open-loop fashion, i.e., no state measurement from the vision system is received which implies that no image processing is taking place during the time period between two consecutive triggering events. A question that naturally arises is how large this time interval can be? Notice, that the smallest time interval is obviously 1, i.e., $k_{i+1} = k_i + 1$ and that the largest time interval is $N-1$. The self-triggered strategy that will be presented later in this chapter, addresses this question and provides sufficient conditions for finding the recalculation periods, or in other words sufficient conditions for triggering the computation of the NMPC law. This leads us to the statement of the problem treated in this chapter:

Problem 4.1. *Consider the system* (4.6) *that is subject to the constraints* (4.7) *and* (4.13). *The objective is* (*i*) *to design a robust feedback control law provided by* (4.14)-(4.17) *such that the system* (4.6) *converges to the terminal constraint set and* (*ii*) *to find a mechanism to decide when the control updates, state measurement and next VTA should occur.*

4.3 Stability Analysis of Self-triggering NMPC framework

In this section the stability analysis for the closed-loop system (4.6)-(4.21) is presented. It has been shown that in the case of the classic MPC set-up, the closed-loop scheme is Input-to-State Stable (ISS) with respect to the disturbances [189]. More details on the notion of ISS in the discrete-time case can be found in [214]. In the subsequent analysis we are going to use the ISS notion in order to derive the self-triggering mechanism.

The traditional approach in establishing stability in predictive control consists of two parts: in the first part it is shown that the initial feasibility implies feasibility afterwards and based on this, it is then shown that the state converges to a bounded set, due to the presence of the persistent disturbances.

We begin by treating the feasibility property. First we are going to provide a necessary definition: Let $\mathcal{X}^{MPC}$ be the set containing all the state vectors for which a feasible control sequence exists that satisfies the constraints of the optimal control problem. In particular, while having a slight violation of the notation we can define $\mathcal{X}^{MPC}$ as follows:

Definition 1. $\mathcal{X}^{MPC} = \{\boldsymbol{x}_0 \in \Re^n | \exists$ a control sequence $\boldsymbol{v}_f \in V,\ \hat{\boldsymbol{x}}_f(j) \in \mathcal{X}_j\ \forall j \in \{1,\dots,N\}$ and $\hat{\boldsymbol{x}}(N) \in \mathscr{O}_f\}$.

Assume, now, that at $k_i \triangleq k$ an event is triggered, thus an OCP is solved and new control sequence is provided. More specifically, solving the OCP of the NMPC (4.14)-(4.17) at a time step k results in an optimal control trajectory $\boldsymbol{v}_f^*(k) \triangleq [\boldsymbol{v}^*(k|k),\dots,\boldsymbol{v}^*(k+N-1|k)]$. Now, consider control inputs at time instants $k+m$ with $m = 1,\dots,N-1$, that are based on the optimal solution at time instant k, $\boldsymbol{v}_f^*(k)$. These can be defined as follows:

$$\tilde{\boldsymbol{v}}(k+j|k+m) = \qquad (4.22)$$
$$\begin{cases} \boldsymbol{v}^*(k+j|k) & \text{for} \quad j = m,\dots,N-1 \\ h(\hat{\boldsymbol{x}}(k+j|k+m)) & \text{for} \quad j = N,\dots,N+m-1 \end{cases}$$

Let $N-1$ control sequences $\tilde{\boldsymbol{v}}_f^m(k)$ be comprised by the control inputs of (4.22), i.e.,

$$\begin{aligned} &\tilde{\boldsymbol{v}}_f^1(k) = \\ &\quad [\boldsymbol{v}^*(k+1|k), \boldsymbol{v}^*(k+2|k),\dots,h(\hat{\boldsymbol{x}}(k+N|k+1))] \\ &\tilde{\boldsymbol{v}}_f^2(k) = \\ &\quad [\boldsymbol{v}^*(k+2|k),\dots,h(\hat{\boldsymbol{x}}(k+N|k+2)), h(\hat{\boldsymbol{x}}(k+N+1|k+2))] \\ &\quad\ \vdots \\ &\tilde{\boldsymbol{v}}_f^{N-1}(k) = \\ &\quad [\boldsymbol{v}^*(k+N-1|k)),\dots,h(\hat{\boldsymbol{x}}(k+2N-2|k+N-1))] \end{aligned}$$

Notice that the time-steps $k+m$ are the discrete-time instants after the time-step of the triggering instant k_i, i.e., $[k, k+1, k+2,\dots,k+N-1] \equiv [k_i, k_i+1, k_i+2,\dots,k_i+N-1]$. From the feasibility of the initial sequence at time-step k and with the help of Assumption 2, it follows that for $m = 1,\dots N-1$ we have $\tilde{\boldsymbol{v}}(k+j|k+m) \in V$. It can be proven that $\hat{\boldsymbol{x}}(k+N+1|k+m) \in \mathscr{O}_f$ for all $m = 1,\dots,N-1$. See *Appendix* 11.6 for the proof.

The state convergence of the perturbed system to a desired terminal set is going to be shown in the sequel. A proper value function must be shown to be decreasing in

order to prove stability of the closed-loop system. At the time-step k, the optimal cost is denoted as $J_N^*(k) = J_N(\boldsymbol{x}_k, \boldsymbol{v}_f^*(k))$, which is evaluated under the optimal control sequence. Analogously, the optimal cost at a time-step $k+m$ with $m \in [1, N-1]$ is denoted as $J_N^*(k+m) = J^*(\boldsymbol{x}_{k+m}, \boldsymbol{v}_f^*(k+m))$. Now let $\tilde{J}_N(k+m)$ to denote the "feasible" cost, evaluated from the control sequence $\tilde{\boldsymbol{v}}_f^m(k)$, that is $\tilde{J}_N(k+m) = \tilde{J}_N(\boldsymbol{x}_{k+m}, \tilde{\boldsymbol{v}}_f^m(k))$. This "feasible" cost will help us to obtain the difference $J_N^*(k+m) - J_N^*(k)$. In particular the difference between the feasible sequence at time-step $k+j$ and the optimal cost at time k using (4.22) is given by:

$$\Delta J_m = \tilde{J}_N(k+m) - J_N^*(k) \leq \left(L_E (L_f)^{(N-m)} + L_F \sum_{i=0}^{N-(m+1)} (L_f)^i \right) \bar{w} - \sum_{i=0}^{m-1} min(q_1, q_2, q_3, q_4, r_1, r_2, r_3) ||\hat{\boldsymbol{x}}(k+i|k)|| \tag{4.23}$$

See *Appendix* 11.7 for the proof. The optimality of the solution yields:

$$J_N^*(k+m) - J_N^*(k) \leq \tilde{J}_N(k+m) - J_N^*(k) \tag{4.24}$$

This result along with the triggering condition that is going to be derived in the next subsection will enable us to provide conclusions for the stability and convergence of the closed-loop system.

4.3.1 The Self-triggered Framework

In this section the self-triggering mechanism is going to be presented. Consider that at time-step k_i an event is triggered. The next control update time k_{i+1} is considered to be unknown and should be found. In particular, the next control update time $k_{i+1} \triangleq k_i + d_i$ should be such that the closed-loop system does not loose any of its desired properties. Thus, we need the value function $J_N^*(\cdot)$ to be decreasing. In particular, given (4.23) and (4.24), for some triggering instant k_i and some time-step d_i after k_i, with $d_i = 1, 2, \ldots, N-1$ it can be obtained the following:

$$J_N^*(k_{i+1}) - J_N^*(k_i) \leq \left(L_E (L_f)^{(N-d_i)} + L_F \sum_{i=0}^{N-(d_i+1)} (L_f)^i \right) \bar{w} - L_Q(d_i) \tag{4.25}$$

where:

$$L_Q(d_i) = \sum_{i=0}^{d_i-1} min(q_1, q_2, q_3, q_4, r_1, r_2, r_3) ||\hat{\boldsymbol{x}}(k+i|k)||$$

The time instant k_{i+1} should be such that:

$$\left(L_E(L_f)^{(N-d_i)} + L_F \sum_{i=0}^{i=N-(d_i+1)} (L_f)^i\right)\bar{w} \leq \sigma L_Q(d_i) \tag{4.26}$$

with $0 < \sigma < 1$. Plugging (4.26) to (4.25), we get

$$J_N^*(k_{i+1}) - J_N^*(k_i) \leq (\sigma - 1)L_Q(d_i) \tag{4.27}$$

This suggests that provided $0 < \sigma < 1$, the decreasing property of the value function is guaranteed. Thus, the next control update time should be triggered when (4.26) is violated. The condition (4.26) should be checked for each consecutive time-step, i.e., for $d_i = 1, 2, \dots$. The time-step that this condition does no longer hold should be the next triggering instant k_{i+1}. Notice that the time-step k_{i+1} can be found beforehand at time k_i, i.e, this is a self-triggering mechanism. Moreover, it should be pointed out that the term $L_Q(d_i)$ includes only predictions of the nominal system that is forming a predicted sequence and that can also be computed by forward integration of (4.3) for time-steps $d_i \in [1, N-1]$. Next we describe the self-triggering mechanism. At time k_i a control update is triggered and a control trajectory for $[k_i, k_i + N - 1]$ is provided. The solution of (4.26) will provide the next update time k_{i+1}. During the time interval $k \in [k_i, k_{i+1})$ the control trajectory $\boldsymbol{v}^*(k_i + d_i|k_i)$ is applied to the plant in an open-loop fashion. Next, at time-step k_{i+1} the optimal control problem of the NMPC is solved again using the current state measurement $\boldsymbol{x}(k_{i+1})$ as the initial state. The controller follows this procedure until the system converges to the terminal constraint set.

We are now ready to state the stability result for this self-triggered MPC framework:

Theorem 4.1. *Consider an underwater robotic vehicle equipped with a camera system with dynamics given in* (4.6) *that is subject to visibility and input constraints* (4.7)-(4.13)*. The control update times that are provided by* (4.26) *and the NMPC law provided by* (4.14)-(4.17) *which is applied to the underwater robotic system in an open-loop fashion during the inter-sampling periods, drive the closed-loop system towards the compact and ultimately bounded set* $\mathscr{E}_f$*.*

4.4 Experiments

The aim of this section is to show the efficacy of the proposed self-triggered NMPC controller to perform a real experiment. A stabilization of an underwater vehicle was performed in real-time. The experiment was held inside a water tank in the Control Systems lab of the Mechanical Engineers Dept. of the National Technical University of

Athens. Notice that the water is a significant source of disturbance that is affecting the trajectory of the vehicle.

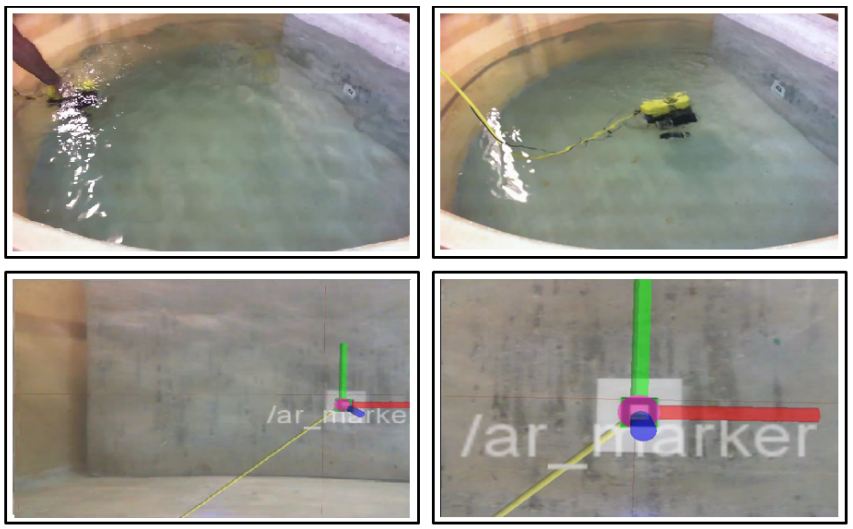


FIGURE 4.5: Experimental setup. (a) The initial configuration: the vehicle is in the initial configuration. Vehicle's view: the target is observed and the target is detected (b) The desired pose of the vehicle wrt target frame. Vehicle's view at the desired position.

4.4.1 System Components

The ROV used is a 3-DOF (VideoRay PRO, VideoRay LLC, Fig.4.3), equipped with three thrusters, a control unit, and a CCD camera. The control unit is connected with a Personal Computer (PC) through a serial communication interface (RS-232). The image dimensions are 640x480 pixels. A target is located on an aluminium plate which is fixed inside the tank. The system software is implemented using the Robotics Operating System (ROS) [215] in Ubuntu Linux and the source code is written in C++ and Python. The state vector of the vehicle with respect to the visual target is calculated in real time using the ROS package *ar_pose* (http://www.ros.org/wiki/ar_pose) which is an Augmented Reality Marker Pose Estimation algorithm based on the *ARToolkit* software library (http://www.hitl.washington.edu/artoolkit/). The target detection and vehicle localization are shown in Fig.4.5. The Constrained Nonlinear Model Predictive Controller employed in this chapter is designed using the NLopt Optimization library [193] and run on a desktop with 8 cores, 3.60 GHz CPU and 16 GB of RAM at $40Hz$.

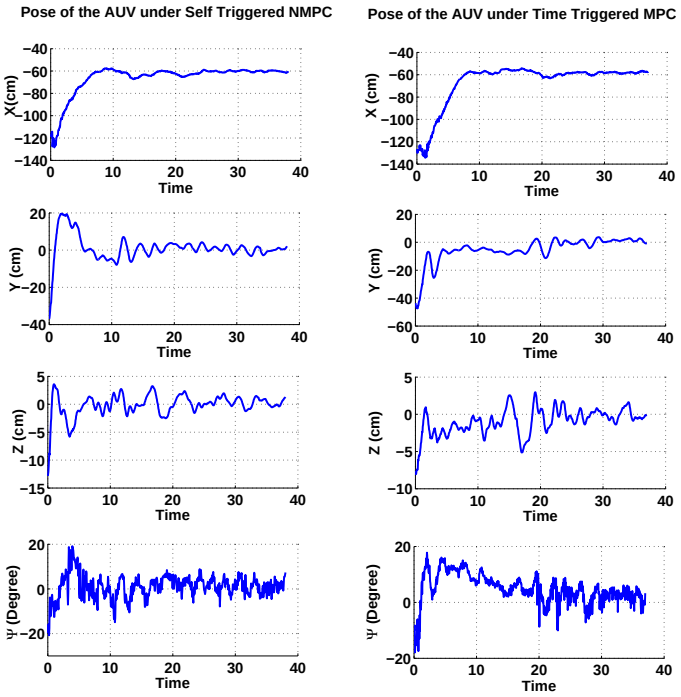


FIGURE 4.6: The state of the robot with respect to the target frame. (a) Self-Triggered MPC. (b) Time-Triggered MPC.

4.4.2 Experimental Results

The goal of the experiment is the stabilization of the vehicle at the desired position. Two experiments were held for comparison. The results of the first experiment using the classical (time triggering) NMPC, will be compared to the second experiment where the Self-Triggering NMPC proposed in this chapter is used. The initial and the desired position of the robot with respect to the target frame is $[\chi_{in}, y_{in}, z_{in}, \psi_{in}] = [-1.2, 0.45, 0.1, -0.401]$ and $\boldsymbol{x}_d = [\chi_d, y_d, z_d, \psi_d]^\top = [-0.6, 0.0, 0.0, 0.0]$ respectively. In the initial pose, the target appears in the right side of the camera view because of the negative yaw angles of the vehicle with respect to the target frame (See Fig.4.5). Note that this is a difficult initial pose and if one does not take the visual constraints into account, the experiment will fail.

The sampling time and the prediction horizon of this experiment are equal to $dt = 0.15$ sec and $N = 6$, respectively. The maximum translational and angular velocity of the vehicle in surge, heave and yaw direction are $\bar{u} = 0.2$ m/sec, $\bar{w} = 0.3$ m/sec and $\bar{r} = 0.3$ rad/sec, respectively. The matrices $\boldsymbol{Q}$, $\boldsymbol{R}$ and $\boldsymbol{P}$ are defined as $\boldsymbol{Q} = diag(0.5, 4.5, 4.5, 0.1)$, $\boldsymbol{R} = diag(0.17, 0.1, 1)$ and $\boldsymbol{P} = diag(1, 1, 1, 1)$, respectively. The maximum permissible distance on the referred water tank is $R_{max} = 1.5$ m. The results of the experiment are presented in Fig.4.6-Fig.4.10.

In Fig.4.6 the states evolution of the system about x, y and ψ in both experiments are depicted. Comparing two experiments, it is apparent that the states of the system are converging to a bounded set around the desired position in both cases. It can be witnessed that the state evolution performances in the case of self triggering are better (or at least the same) compared to the classical time triggering. In the both cases, all of the constraints (FoV, input bound) are satisfied and the robot reaches to the desired position with respect to the target.

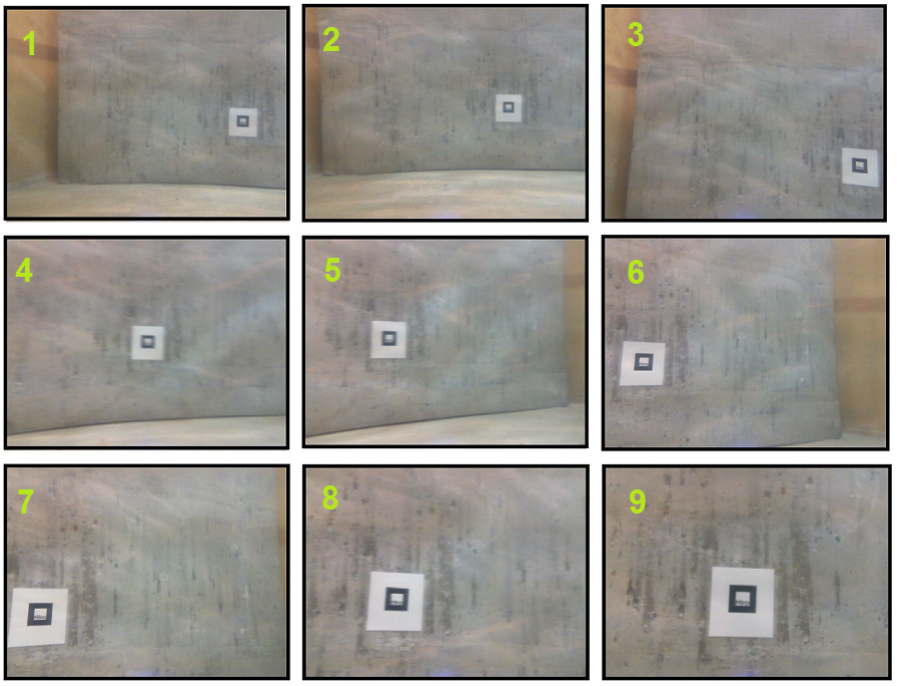


FIGURE 4.7: Camera view during the experiment. From initial view (top and left) to the final view(bottom and right) The target remains inside the field of view of the camera

In Fig.4.7 and Fig.4.8 the camera view and the coordinate of the target center during the experiments are presented, respectively. It is obvious that the target remains inside of the FoV of the camera. In Fig.4.9 the triggering instants in the case of self triggered are captured. When the vertical axis has the value 1, the NMPC is triggered, thus the image is processed and the state vector is calculated, consequently the optimization algorithm of the NMPC is running and a new control input sequence is computed, where the first one of this control input sequence is sent to the robot. For value 0 the rest of the last computed control input sequence is used, thus the control law is implemented on the system in an open-loop fashion, and therefore no optimization and no image processing is running. Notice that in the case of classical NMPC, the scheme is triggering at all sampling times, which means that the image is always being processed and the optimization algorithm is always running. Using the self triggered condition proposed in this chapter, the triggering of the vision data and the NMPC controller

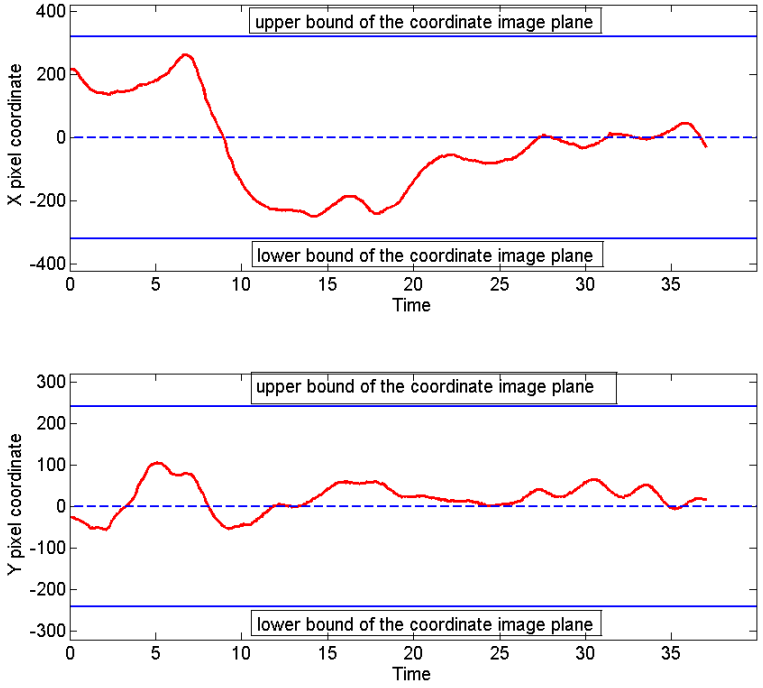


FIGURE 4.8: Image coordinate of the target center. The target remains inside the image coordinate bound.

have been reduced by 50% (124 triggering instead of 253) with respect to the classical time triggered NMPC.

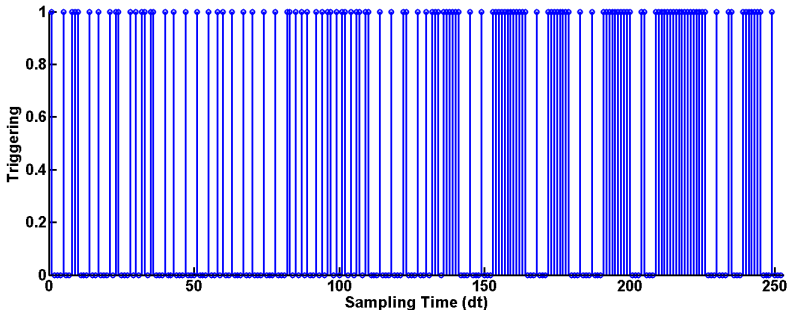


FIGURE 4.9: The triggering instants in Self triggered NMPC . When the vertical axis has the value 1, the NMPC is triggered. For value 0 the control law is implemented on the robot in an open-loop fashion.

Comparing triggering instants of the Fig.4.9 to the image target center coordinate in Fig.4.8, one may notice that when the target is going to leave the image plane (at the region of 6 and 14-17 seconds of the experiment) the triggering instants are more frequent. This fact appears at region of the 40 and the 80-110 sampling times respectively in the Fig.4.9.

Comparing triggering instants of the Fig.4.9 to the state evolution of the system Fig.4.6, one may notice that when the robot is getting near to the desired position the triggering instants are more frequent. This is happening because, near the desired position the

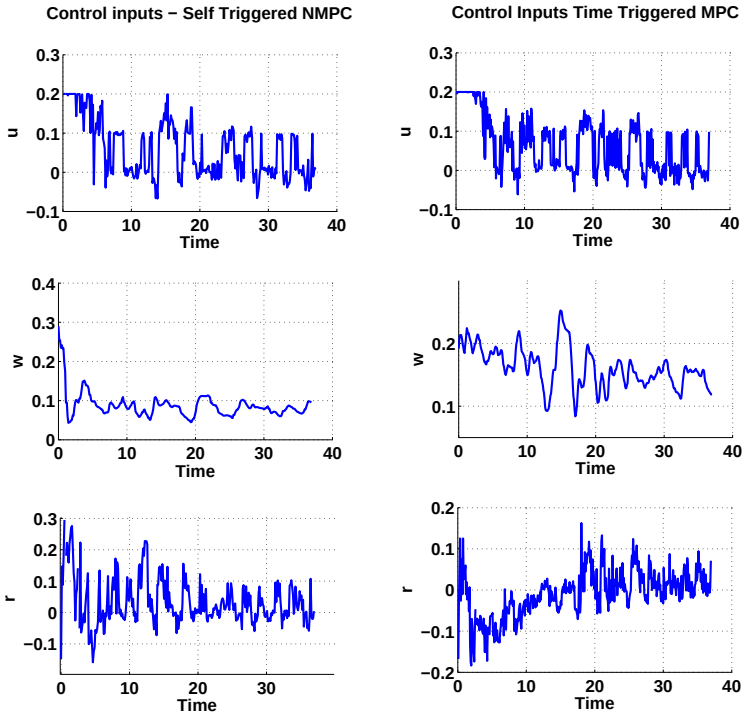


FIGURE 4.10: Control inputs. (a) Self-Triggered MPC. (b) Time-Triggered MPC.

system becomes more demanding due to the FoV constraints (the target becomes larger in the image plane) and because the external disturbances move the robot away from desired position.

The computational time in the case when a new state information from vision system and a new control sequence are calculated (triggering instant) is about 0.1 sec while in the case of the open loop control (using the self triggered NMPC) is being reduced to 0.0002 sec. This is happening because neither the image processing nor the optimization algorithm are running between two triggering instants in the case of self triggered control. This is very important in the case of Autonomous Underwater Vehicle where the energy sources (batteries) are very limited and the recharging procedure is very difficult and time consuming. Finally in Fig.4.10 the control inputs are depicted. It can be seen that no violation of the constraints took place.

4.4.3 Video

The aforementioned experimental study is demonstrated in a HD video at the following url: `http://youtu.be/_NdalTHclGY`.
Alternatively, it can be found in the attached dvd as file: video_ch4.mpg

4.5 Conclusion

In this chapter, a Self-triggered Position Based Visual Servoing scheme based on Model Predictive Control is presented for Underwater Robotic Vehicles. The purpose of the controller is to navigate and stabilize the vehicle towards a visual target and assuring that the target will always remain inside the camera's field of view. The main idea is to trigger and run the optimal control problem of the NMPC and the vision tracking algorithm only when it is needed and not periodically as in the case of the classic (time triggered) NMPC schemes. With the self-triggered approach both the control input and the next control update time are evaluated in order to avoid continuous supervision of the actual state of the system, i.e., to avoid continuous measurements from the vision system. During the inter-sampling instants the control sequence from the NMPC is applied to the system in an open-loop fashion, i.e., neither the image processing nor the optimization algorithm are running between two triggering instants. This results to the reduction of the CPU computational effort, processing of vision data and the energy consumption which are of utmost importance especially in the case of AUVs where lean computational algorithms and low energy consumption are required. Moreover, the inputs and visibility constraints (i.e preserving the target inside the camera's field of view), the external disturbances induced by currents and waves, as well as the vehicle's kinematic constraints due to under-actuation, are being considered during the control design.

Chapter 5

Self Triggered Image Based Visual Servoing for Underwater Robotic Vehicles

Previously in Chapter 4 based on Nonlinear Model Predictive Control (NMPC), we proposed a Position Based Visual Servoing (PBVS) scheme for case when the relative position between the robot and object can be obtained real-time. This relative position can usually be estimated either by observing a known marker in the subject or by using knowledge about the object's dimensions. However, there are many cases where neither the object's dimensions nor a known marker can be available ahead of time. In such a case, we assume that a desired image view of the object is available and thus the control objective is to drive and stabilize the current image view to the desired image view. Therefore, the error function that is going to be minimized is based on the position of the image features in the image plane between the current and desired images. Motivated by the aforementioned fact, in this chapter we extend our previous results on Position Based Visual Servoing into a general image based version by presenting the Self Triggered Image Based Visual Servoing scheme for cases where the relative position between the robot/camera system and the point of interest can not be available online.

5.1 Introduction

In this chapter, the IBVS scheme is considered, as it is more efficient than the other two, owing to its inherent robustness against camera calibration imperfections. A significant issue in visual servoing is handling the visibility constraints, imposed by the fact that the image features are required not to leave the Field Of View (FOV) of the camera

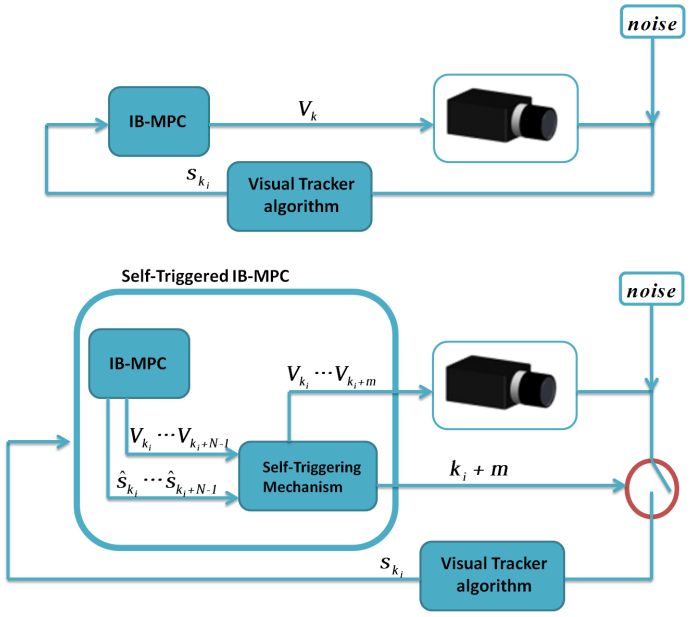


FIGURE 5.1: The classic IBVS-MPC is depicted in the top block diagram. The bottom diagram represents the ST-IBVS-MPC scheme.

during its motion [82]. In order to handle the visibility constraints various methods have been developed. More specifically, in [216] and [94] path planning of the image features based on the motion of the camera in 3D space are presented. Moreover a path planning strategy via LMI optimization has been studied in [99] while the prescribed transient and steady state response on the image feature error is presented in [217].

Nonlinear Model Predictive Control (NMPC) [58], due to its strong and efficient ability to handle input and state constraints is an ideal approach to be used in IBVS. The IBVS-NMPC framework has been studied in [218], [107], [109], [110], [111], [112] and [113]. In our previous work [218] a robustness analysis of the IBVS-MPC w.r.t the disturbances and noises on the image features has been proposed. The Visual Predictive Control (VPC) scheme has been proposed in [107] where the visibility constraints are formulated as state constraints. Some applications of IBVS-MPC for navigation of an Unmanned Aerial Vehicles (UAV), mobile robots, robot manipulators as well as some medical applications have been presented in [110], [111], [112] and in [113], respectively.

A standard visual servoing scheme consists the periodic (at every sampling period) use of the vision feedback which is extracted from the image to generate a task error and using a control algorithm to minimize this error. The process of image feature extraction, matching with a desired image and using them to generate the task error, is usually referred in the literature as the *Visual Tracking* [68]. The main concerns of a Visual Tracking Algorithm (VTA) in a complex environment are accuracy and robustness. It is well known that, a real-time robotic application in a complex environment, the accurate

and robust VTA is very heavy process and has high computational cost which usually results in large energy usage and delays on the closed loop system.

This problem becomes more apparent when autonomous robotic systems are considered such as Autonomous Underwater Vehicles (AUVs) and Underwater Vehicle Manipulator Systems (UVMSs) that suffer from limited energy resources (batteries) and usually are equipped with small and not so powerful embedded computing unit. Long lasting inspection tasks in complex environments require accurate VTA and concurrently high autonomy rate of the system. The problem then, becomes more evident, because the continuous recharging procedure is undesirable, difficult and time consuming. In addition, the continuous visual tracking at every sampling time owing to the existence of a weak computing unit in these systems, leads to bigger sampling periods on the closed loop system, that consequently reduce the accuracy of the system.

Is it possible to relax the visual tracking while the performance of the visual servoing scheme remains the same?", in other words, is it possible to design a visual servoing scheme that decides when the robot needs to track the visual information and when not, while the whole system does not lose the required performance? This question, motivates the self-triggered design framework for Visual Servoing in order to track the vision information and compute the control law only when it is needed. In Self Triggered control the key attribute is that the decision for sampling the state measurement as well as the execution of the control task is not made ad-hoc as in the sampled-data case, but it takes into account state or output feedback in order to sample as infrequently as possible while guaranteeing to preserve the stability of the system, see Fig5.1. Some introductory papers on self-triggered control can be found in [196] and [201]. In previous Chapter 4 a Self Triggered Position Based Visual Servoing scheme for an under-actuated underwater robotic vehicle was given. In this chapter we extend our previous results into a general image based version by presenting the Self Triggered-Image Based Visual Servoing-Model Predictive Control (ST-IBVS-MPC) scheme. However, apart from this, the analytical stability analysis with respect to the optimality rate of the controller as well as a rigorous robustness analysis of the IBVS in the case of self triggering case, are presented in this chapter. This framework provides aperiodic control sequences that lead to stable closed-loop responses of the IBVS scheme. Moreover, its mechanism decides when the next VTA should occur. This results to the reduction of the computational effort, processing of vision data, energy consumption and therefore it increases the autonomy rate of the system. The visibility constraints, inputs constraints, optimality rate of the predictive controller, as well as the external disturbances induced by noises on image features, are also being considered during the control design.

The chapter is organized as follows: Modeling of the IBVS, visibility and input constraints are given in Section 5.2. Section 5.3 includes the problem statement and the control design of the IBVS-MPC. Section 5.4 accommodates the stability analysis of the proposed self-triggered IBVS-MPC scheme. In Section 5.5 a simulation study is given and finally, Section 5.6 concludes the chapter.

5.2 Mathematical Modelling

In this section, the mathematical formulation of the image based visual servoing problem is presented. Let $[X_c, Y_c, Z_c]^\top$ be the axes of the camera frame $\mathcal{C}$ attached at the center of the camera O_c. The coordinates of the image frame $\mathcal{I}$ are given by $[u, v]^\top$ with O_I denoting the center of the image, as depicted in Fig. 5.2. Given a set of n fixed 3D

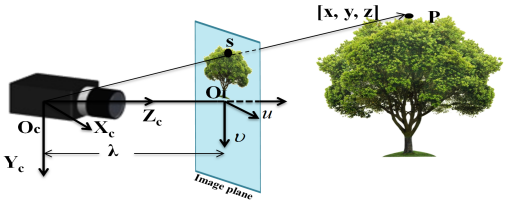


FIGURE 5.2: The coordinate frame of the camera system.

points $\boldsymbol{P}^i = [x^i, y^i, z^i]^\top$, $i = 1, \ldots, n$ expressed in the camera frame, the corresponding 2D image feature $\boldsymbol{s}^i = [u^i, v^i]^\top$, $i = 1, \ldots, n$ are given as follows [68]:

$$\boldsymbol{s}^i = \begin{bmatrix} u^i \\ v^i \end{bmatrix} = \frac{\lambda}{z^i} \begin{bmatrix} x^i \\ y^i \end{bmatrix} \tag{5.1}$$

where λ is the focal length of the camera (see Fig.5.2). The time derivative of (5.1) is given by:

$$\dot{\boldsymbol{s}}^i = \boldsymbol{L}^i(z^i, \boldsymbol{s}^i)\boldsymbol{v},\ i = 1, \ldots, n \tag{5.2}$$

where:

$$\boldsymbol{L}^i\left(z^i, \boldsymbol{s}^i\right) = \begin{bmatrix} -\frac{\lambda}{z^i} & 0 & \frac{u^i}{z^i} & \frac{u^i v^i}{\lambda} & -\frac{(\lambda^2+(u^i)^2)}{\lambda} & v^i \\ 0 & -\frac{\lambda}{z^i} & \frac{v^i}{z^i} & \frac{(\lambda^2+(v^i)^2)}{\lambda} & -\frac{u^i v^i}{\lambda} & -u^i \end{bmatrix}$$

is the interaction matrix [68], and $\boldsymbol{v} = [\boldsymbol{T}, \boldsymbol{\Omega}]^\top = [T_x, T_y, T_z, \omega_x, \omega_y, \omega_z]^\top$, denotes the translational and angular velocities of the camera system. Let us also define the overall image feature vector $\boldsymbol{s} = \left[\boldsymbol{s}^{i\top}, \cdots, \boldsymbol{s}^{i\top}\right]^\top \in \Re^{2n}$, the time derivative of which is given by:

$$\dot{\boldsymbol{s}} = \boldsymbol{L}(\boldsymbol{z}, \boldsymbol{s})\boldsymbol{v} \tag{5.3}$$

where $\boldsymbol{L}(\boldsymbol{z},\boldsymbol{s}) = \left[\boldsymbol{L}^{i\top}\left(z^i,\boldsymbol{s}^i\right),\cdots,\boldsymbol{L}^{n\top}\left(z^n,\boldsymbol{s}^n\right)\right]^\top$ is the overall interaction matrix and $\boldsymbol{z} = \left[z^i,\ldots,z^n\right]^\top$. Using the Newton-Euler method for approximating (5.3), the model of the system in the discrete-time frame becomes:

$$\boldsymbol{s}_{k+1} = \boldsymbol{s}_k + dt(\boldsymbol{L}_k \cdot \boldsymbol{v}_k) \tag{5.4}$$

where dt and k are the sampling period and time-step respectively. For the rest of paper we denote $\boldsymbol{s}_k = [\boldsymbol{s}_k^{1\top},\cdots,\boldsymbol{s}_k^{n\top}]^\top$ to be the vector of the state at a time-step k. The vector of the velocity of the camera $\boldsymbol{v}_k = [T_{(x,k)} \quad T_{(y,k)} \quad T_{(z,k)} \quad \omega_{(x,k)} \quad \omega_{(y,k)} \quad \omega_{(z,k)}]^\top$ will denote the input of the system at a time-step k. The aforementioned discrete-time system (5.4) is the nominal system and can be written in stack vector form as:

$$\boldsymbol{s}_{k+1} = f(\boldsymbol{s}_k,\boldsymbol{v}_k) \tag{5.5}$$

The control constraint set V is compact and is given by:

$$\boldsymbol{v}_k \in V \subseteq \Re^6 \tag{5.6}$$

The constraints of the input are of the form $|T_x| \leq \bar{T}_x$, $|T_y| \leq \bar{T}_y$, $|T_z| \leq \bar{T}_z$, $|\omega_x| \leq \bar{\omega}_x$, $|\omega_y| \leq \bar{\omega}_y$ and $|\omega_z| \leq \bar{\omega}_z$. where $(\ \bar{\cdot}\ )$ denotes the upper bound for each of the variables. We set $\|\boldsymbol{v}_k\| \leq \bar{V}$, with $\bar{V}$ to be the predefined upper bound for the camera velocity. Finally, owing to the limited field of view of the camera, the image coordinates are subject to the following visibility constraints:

$$u_{\min} \leq u^i \leq u_{\max},\ i = 1,\ldots,n \tag{5.7}$$

$$v_{\min} \leq v^i \leq v_{\max},\ i = 1,\ldots,n \tag{5.8}$$

where $u_{\min}$, $v_{\min}$ and $u_{\max}$, $v_{\max}$ are the lower and upper bounds (in pixels) of the image plane coordinates u, v respectively. The visibility constraints (5.7)-(5.8) ensure that the image features do not leave the image plane during the control operation. These visibility constraints form the state constraint set $S_{set} \subseteq \Re^{2n}$ i.e., $\boldsymbol{s}_k \in S_{set}$. Now, assume that the system (5.5) is affected by noise on image features, in the form of output noise from the VTA. This noise is introduced into the system as an external disturbance vector that is formed by:

$$\boldsymbol{\xi}_k = [\boldsymbol{\xi}_k^{1\top}\ldots\boldsymbol{\xi}_k^{n\top}]^\top$$

where $\boldsymbol{\xi}_k^i = [\boldsymbol{\xi}_u^i,\boldsymbol{\xi}_v^i]^\top \quad i = 1,\ldots,n$. These disturbances form the compact disturbance set $\boldsymbol{\xi} \subseteq \Re^{2n}$ with $\boldsymbol{\xi}_k \in \boldsymbol{\xi}$, upper bounded by $\bar{\boldsymbol{\xi}}$, where $||\boldsymbol{\xi}_k|| \leq \bar{\xi}$, $\forall \boldsymbol{\xi}_k \in \boldsymbol{\xi}$. Therefore, an

actual system can be considered as:

$$\boldsymbol{s}_{k+1} = f(\boldsymbol{s}_k, \boldsymbol{v}_k) + \boldsymbol{\xi}_k \tag{5.9}$$

5.3 Control Design and Problem Statement

In this section the problem of Self Triggered IBVS-MPC will be formulated. The goal is to control the actual system (5.9) subject to the visibility and control constraints of (5.7)-(5.8) and (5.6) to reach to a compact image feature set that includes the desired state $\boldsymbol{s}_d = [\boldsymbol{s}_d^1, \dots, \boldsymbol{s}_d^n]^\top \in S_{set}$. In order to achieve this task we use a nonlinear model predictive controller that consists in solving iteratively an open-loop Optimal Control Problem (OCP) w.r.t a control sequence $\boldsymbol{v}_f(k)$. The OCP of the IBVS-MPC is given as follows:

$$\min_{\boldsymbol{v}_f(k)} J_N(\boldsymbol{s}_k, \boldsymbol{v}_f(k)) = \tag{5.10}$$

$$\min_{\boldsymbol{v}_f(k)} \sum_{i=0}^{N-1} F(\hat{\boldsymbol{s}}(k+i|k), \boldsymbol{v}(k+i|k)) + E(\hat{\boldsymbol{s}}(k+N|k))$$

subject to

$$\hat{\boldsymbol{s}}(k+j|k) \in S_j, \qquad \forall j = 1, \dots, N-1, \tag{5.11}$$

$$\boldsymbol{v}(k+j|k) \in V, \qquad \forall j = 0, \dots, N-1, \tag{5.12}$$

$$\hat{\boldsymbol{s}}(k+N|k) \in \mathscr{E}_f \tag{5.13}$$

where N denotes the prediction horizon and the set $\mathscr{E}_f$ is the terminal set. F and E are the running and terminal cost functions, respectively and are of quadratic form, i.e., $F(\hat{\boldsymbol{s}}, \boldsymbol{v}) = \hat{\boldsymbol{s}}^\top \boldsymbol{Q} \hat{\boldsymbol{s}} + \boldsymbol{v}^\top \boldsymbol{R} \boldsymbol{V}$ and $E(\hat{\boldsymbol{s}}) = \hat{\boldsymbol{s}}^\top \boldsymbol{P} \hat{\boldsymbol{s}}$, with $\boldsymbol{P}$, $\boldsymbol{Q}$ and $\boldsymbol{R}$ to be positive definite matrices. Particularly we define $Q = diag\{q_1, q_2, \cdots, q_{(2n-1)}, q_{(2n)}\}$, $R = diag\{r_1, r_2, r_3, r_4, r_5, r_6\}$ and $P = diag\{p_1, p_2, \cdots, p_{(2n-1)}, p_{(2n)}\}$. The vector $\hat{\boldsymbol{s}}(k+j|k)$ denotes the predicted state of the nominal system (5.5) at sampling time $k+j$ with $j \in \mathbb{Z}_{\geq 0}$. The predicted state is based on the measurement of the image features $\boldsymbol{s}_k$ at a sampling time k, while applying a sequence of control inputs $\{\boldsymbol{v}_k, \dots, \boldsymbol{v}_{k+j-1}\}$. Thus: $\hat{\boldsymbol{s}}(k+j|k) = f(\hat{\boldsymbol{s}}(k+j-1|k), \boldsymbol{v}_{k+j-1})$. It holds that $\hat{\boldsymbol{s}}(k|k) = \boldsymbol{s}_k$. We distinguish the state of nominal system that will be denoted as $\hat{\boldsymbol{s}}(\cdot)$ with the state of actual system, i.e. the system that is affected by disturbances which will be denoted as $\boldsymbol{s}(\cdot)$. In order to proceed to the subsequent analysis of the ST-IBVS-MPC, we are going to briefly present some preliminary results:

Property 5.1 ([218]). *The nominal model (5.5), subject to constraints (5.7)-(5.8) and (5.6), is locally Lipschitz in S_{set} with Lipschitz constant $0 < C_f < \infty$, such that for every $\boldsymbol{v} \in V$, $\|f(\boldsymbol{s}_1, \boldsymbol{v}) - f(\boldsymbol{s}_2, \boldsymbol{v})\| \leq C_f \|\boldsymbol{s}_1 - \boldsymbol{s}_2\|$.*

Property 5.2 ([218]). *The difference between the real state $\boldsymbol{s}_{k+j}$ at the time $k+j$ and the predicted state $\hat{\boldsymbol{s}}(k+j|k)$ at the same time under the same control sequence, starting at the same initial state $\boldsymbol{s}_k$ is upper bounded by:*

$$||\boldsymbol{s}_{k+j} - \hat{\boldsymbol{s}}(k+j|k)|| \leq \sum_{i=0}^{j-1} (C_f)^i \bar{\xi} \tag{5.14}$$

Property 5.3 ([218]). *The cost function $F(\boldsymbol{s}, \boldsymbol{v})$ is lower bounded by a $\mathcal{K}_\infty$-function. In particular:*

$$F(\boldsymbol{s}, \boldsymbol{v}) \geq min(q_1, \cdots, q_{2n}, r_1, \cdots, r_6)||\boldsymbol{s}||^2 \tag{5.15}$$

Property 5.4 ([218]). *The cost function $F(\boldsymbol{s}, \boldsymbol{v})$ is Lipschitz continuous in $S_{set} \times V$, with Lipschitz constant C_F, where:*

$$C_F = 2(n(\bar{u}^2 + \bar{v}^2))^{\frac{1}{2}} \cdot \sigma_{max}(\boldsymbol{Q}) \tag{5.16}$$

where $\sigma_{max}(\boldsymbol{Q})$ denotes the largest singular value of the $\boldsymbol{Q}$.

Assumption 5.1. There is an admissible positively invariant set $\mathscr{E} \subset S_{set}$ such that $\mathscr{E}_f \subset \mathscr{E}$, where $\mathscr{E} = \{s \in S_{set} : ||\boldsymbol{s}|| \leq \varepsilon_0\}$ with ε_0 being a positive parameter.

Assumption 5.2. Inside the $\mathscr{E}_f$, there is a local controller $\boldsymbol{v}_k = h(\boldsymbol{s}_k) \in V, \quad \forall s \in \mathscr{E}$ and a Lyapunov function E such that $E(f(\boldsymbol{s}_k, h(\boldsymbol{s}_k))) - E(\boldsymbol{s}_k) + F(\boldsymbol{s}_k, h(\boldsymbol{s}_k)) \leq 0$.

Assumption 5.3. The associated Lyapunov function for the terminal region is Lipschitz in $\mathscr{E}$, with Lipschitz constant
$C_E = 2\varepsilon_0 \sigma_{max}(P)$ for all $s \in \mathscr{E}$. Considering that from *Assumption 1* we have: $||s|| = (|u^1|^2 + |v^1|^2 + \cdots + |u^n|^2 + |v^n|^2)^{\frac{1}{2}} \leq \varepsilon_0$ for all $s \in \mathscr{E}$.

Assumption 5.4. Inside the set $\mathscr{E}$ we have $E(s) = \boldsymbol{s}^T P s \leq \alpha_\varepsilon$, where $\alpha_\varepsilon = max\{p_1, \cdots, p_{(2n)}\}\varepsilon_0^2 > 0$. Assuming that $\mathscr{E} = \{\boldsymbol{s} \in S_{set(N-1)} : h(\boldsymbol{s}) \in \boldsymbol{v}\}$ and taking a positive parameter α_{ε_f} such that $\alpha_{\varepsilon_f} \in (0, \alpha_\varepsilon)$, we assume that the terminal set $\mathscr{E}_f = \{\boldsymbol{s} \in \Re^3 : E(\boldsymbol{s}) \leq \alpha_{\varepsilon_f}\}$ is such that $\forall \boldsymbol{s} \in \mathscr{E}$, $f(\boldsymbol{s}, h(\boldsymbol{s})) \in \mathscr{E}_f$.

5.3.1 Problem Statement

Before proceeding the analysis of ST-IBVS-MPC, the general problem statement that clarifies the aim of the analysis must be given. The solution of the Image based Model

Predictive Controller (5.10)-(5.13) at a time-step k provides an control sequence $\boldsymbol{v}_f^*$ which equals to $\boldsymbol{v}_f^*(k) \triangleq [\boldsymbol{v}^*(k|k), \ldots, \boldsymbol{v}^*(k+N-1|k)]$. In the classic IBVS-MPC approach, such as in our basic previous work [218], only the first control vector, i.e $\boldsymbol{v}^*(k|k)$ is applied to the system and all the remaining elements of the $\boldsymbol{v}_f^*$ are discarded. At the next time-step $k+1$, the VTA is computed and new vision measurements are received and then the whole procedure is repeated again. This is iteratively repeated until the camera reaches to the desired position w.r.t target. Here, we suggest that a portion of the computed control sequence may be applied to the system and not only the first vector. More specifically, suppose a triggering instant k_i, at which the VTA has been computed and the IBVS-MPC has been solved. The control sequence that will now be applied to the robot is of the form:

$$[\boldsymbol{v}^*(k_i|k_i), \boldsymbol{v}^*(k_i+1|k_i), \ldots \boldsymbol{v}^*(k_i+d_i|k_i)] \tag{5.17}$$

for all $d_i \in [1, k_{i+1}-k_i] \in \mathbb{Z}_{\geqslant 1}$, where k_{i+1} is the next triggering instant. During the time interval $[k_i, k_{i+1})$ the control law is applied to the robot in an open-loop fashion, i.e., no VTA is needed to run as well as, no vision measurements are received. But how large this time interval can be? Notice that the smallest time interval is obviously 1, that is if $k_{i+1} = k_i + 1$ and that the largest time interval is $N-1$. The new self-triggered IBVS approach that will be presented in this paper, addresses this question and provides sufficient conditions for finding the triggering periods d_i, or in other words sufficient conditions for running the VTA and the computation of the NMPC law.

5.4 Stability Analysis of Self-triggering IBVS-MPC framework

In this section the stability analysis for the closed-loop system of the proposed Self Triggering-IBVS scheme w.r.t disturbances is going to be presented. We begin by assuming that at $k_i \triangleq k-1$ an event is triggered, the solution of the (5.10)-(5.13) at this time results in an optimal control trajectory $\boldsymbol{v}_f^*(k-1) \triangleq [\boldsymbol{v}^*(k-1|k-1), \ldots, \boldsymbol{v}^*(k+N-2|k-1)]$. Based on this optimal control trajectory, we can define a feasible control input, i.e., for $m = 0, \ldots, N-2$ given by:

$$\tilde{\boldsymbol{v}}(k+j|k+m) = \begin{cases} \boldsymbol{v}^*(k+j|k-1) & \text{for} \quad j = m, \ldots, N-2 \\ h(\hat{\boldsymbol{s}}(k+j|k+m)) & \text{for} \quad j = N-1, \ldots, N+m-1 \end{cases} \tag{5.18}$$

Notice that the time-steps $k+m$ are the discrete-time instants after the time-step of the triggering instant k_i, i.e., $[k-1, k, k+1, \ldots, k+N-2] \equiv [k_i, k_i+1, k_i+2, \ldots, k_i+N-1]$. From the (5.12) and with the help of Assumption5.2, it follows that for $m = 0, \ldots N-$

2 we have $\tilde{\boldsymbol{v}}(k+j|k+m) \in V$. The traditional approach in establishing stability in predictive control consists of two parts: in the first part it is shown that the initial feasibility implies feasibility afterwards and based on this, it is then shown that the state converges to a bounded set, due to the presence of the persistent disturbances. We begin by treating the feasibility property.

5.4.1 Feasibility of ST-IBVS-MPC

Let $\mathcal{S}^{IB}$ be the set containing all the state vectors for which a feasible control sequence exists that satisfies the constraints of the optimal control problem. Now we are going to find an upper bound $\bar{\xi}$ for disturbances such that $\hat{\boldsymbol{s}}(k+N|k+m) \in \mathscr{E}_f$ for all $m = 0, \ldots, N-2$. Assume that at time instant $k_i \triangleq k-1$ an event is triggered, thus the VTA is activated, the OCP of (5.10)-(5.13) is solved and a new control sequence is provided. A partition of this control sequence, for $m \in [0, \ldots, N-2]$ drives and stabilizes the image features vector $\boldsymbol{s}$ to a set $\mathscr{E}$ around the desired image features vector $\boldsymbol{s}^*$, satisfying all constraints, if and only if the disturbances are bounded by:

$$\bar{\xi} \leq \frac{\alpha_\varepsilon - \alpha_{\varepsilon_f}}{C_E C_f^{(N-1)-m} \sum_{i=0}^{m} (C_f^i)} \tag{5.19}$$

Proof. In view of Property-5.2 we can derive the following:

$$s||\hat{\boldsymbol{s}}(k+N-1|k+m) - \hat{\boldsymbol{s}}(k+N-1|k-1)|| \leq \quad \leq C_f^{(N-1)-m} \sum_{i=0}^{m} (C_f^i) \bar{\xi}$$

From the Lipschitz property of $E(\cdot)$ (*Assumption-5.3*) we get:

$$\begin{aligned} &E(\hat{\boldsymbol{s}}(k+N-1|k+m)) - E(\hat{\boldsymbol{s}}(k+N-1|k-1)) \\ &\leq C_E||\hat{\boldsymbol{s}}(k+N-1|k+m) - \hat{\boldsymbol{s}}(k+N-1|k-1)|| \leq C_E C_f^{(N-1)-m} \sum_{i=0}^{m} (C_f^i) \bar{\boldsymbol{\xi}} \end{aligned}$$

Noticing that $\hat{\boldsymbol{s}}(k+N-1|k-1) \in \mathscr{E}_f$, from *Assumption-5.4* we get $E(\hat{\boldsymbol{s}}(k+N-1|k-1)) \leq \alpha_{\varepsilon_f}$. We want $\hat{\boldsymbol{s}}(k+N-1|k+m) \in \mathscr{E}$, thus from *Assumption-5.4*, it should satisfy $E(\hat{\boldsymbol{s}}(k+N-1|k+m) \leq \alpha_\varepsilon$, so we get:

$$E(\hat{\boldsymbol{s}}(k+N-1|k+m) \leq \alpha_{\varepsilon_f} + C_E q(m) \bar{\xi} \leq \alpha_\varepsilon$$

with $q(m) \triangleq C_f^{(N-1)-m} \sum_{i=0}^{m} (C_f^i)$. Furthermore, we obtain the following:

$$\alpha_{\varepsilon_f} + C_E q(m) \bar{\xi} \leq \alpha_\varepsilon \Rightarrow \bar{\xi} \leq \frac{\alpha_\varepsilon - \alpha_{\varepsilon_f}}{C_E C_f^{(N-1)-m} \sum_{i=0}^{m} (C_f^i)} \tag{5.20}$$

which states that the set $\mathcal{S}^{IB}$ is robustly positively invariant for disturbances bounded by (5.20) for all $m = 0, \ldots, N-2$, and from from *Assumption-5.4* we get $\hat{\boldsymbol{s}}(k+N|k+m) \in \mathscr{E}_f$, which concludes the proof. □

5.4.2 Convergence of ST-IBVS-MPC

Now we will show the convergence property of the actual system (5.9) using the optimal cost function that will be used as a Lyapunov function. At the time-step $k-1$, the optimal cost is denoted as $J_N^*(k-1) = J_N(\boldsymbol{s}_{k-1}, \boldsymbol{v}_f^*(k-1))$, which is evaluated under the optimal control sequence. Analogously, the optimal cost at a time-step $k+m$ with $m \in [0, N-2]$ is denoted as $J_N^*(k+m) = J^*(\boldsymbol{s}_{k+m}, \boldsymbol{v}_f^*(k+m))$. Now let $\tilde{J}_N(k+m)$ to denote the "feasible" cost, evaluated from the control sequence $\tilde{\boldsymbol{v}}_f^m(k-1)$, that is $\tilde{J}_N(k+m) = \tilde{J}_N(\boldsymbol{s}_{k+m}, \tilde{\boldsymbol{v}}_f^m(k-1))$. It is well known that in a real experiment with a real robotic system, owing to the finite iteration's number of the optimization procedure, it is not possible to find the exact optimal solution of (5.10)-(5.13) and always some sub-optimality rate must be considered. Thus for the feasible cost function defined above we consider a "real" version of it denoted by $\tilde{J}'(\cdot)$ and a sub-optimality rate denoted by B such that $B = \frac{\tilde{J}}{\tilde{J}'}, B \in (0, 1]$. Now, the difference between the real-feasible sequence at time-step $k+j$ and the optimal cost at time $k-1$ using (5.18) is given by:

$$
\begin{aligned}
&\Delta J_m = \tilde{J}'_N(k+m) - J^*{}_N(k-1) = \\
&= \tilde{J}_N(k+m) - J_N^*(k-1) + \tilde{J}_N(k+m)(\frac{1}{B} - 1) \leq \\
&\leq C_E (C_f)^{(N-(m+1))} \bar{\xi} + L_S(m) - L_Q(m) + (\frac{1}{B} - 1)\big(L_S(m) + L_P(m) + \max(p_1, \ldots, p_{2n})\varepsilon_0^2\big)
\end{aligned}
\tag{5.21}
$$

Where:

$$
\begin{aligned}
L_S(m) &= C_F \cdot \sum_{i=0}^{N-(m+2)} (C_f)^i \bar{\xi}, \\
L_P(m) &= \max(p_1, \ldots, p_{2n}, r_1, \ldots, r_6) \sum_{i=0}^{N-(m+2)} |||[\hat{\boldsymbol{s}}(\cdot), \boldsymbol{v}^*(\cdot)]||| \\
L_Q(m) &= \sum_{i=-1}^{m-1} min(q_1, \ldots, q_{(2n)} r_1, \ldots, r_6) ||\hat{\boldsymbol{s}}(k+i|k-1)||
\end{aligned}
$$

where we denote by$(\cdot) = (k+i+m|k-1)$. See *Appendix 11.8* for the proof. The optimality of the solution yields:

$$
J^*{}_N(k+m) - J^*{}_N(k-1) \leq \tilde{J}'_N(k+m) - J^*{}_N(k-1) \tag{5.22}
$$

This result along with the triggering condition that is going to be derived in the next subsection will enable us to provide conclusions for the stability and convergence of the closed-loop system.

5.4.3 The Self-triggered Framework

Consider that at time-step k_i a measurement from Visual Tracking algorithm is received and a new control input is calculated. The next triggering time $k_{i+1} \triangleq k_i + d_i$ that is going to be found should be such that the closed-loop system of IBVS-MPC does not loose any of its desired performances. Thus, we need the value function $J_N^*(\cdot)$ to be decreasing. Given (5.21) and (5.22), for some triggering instant k_i and some time-step $d_i = m + 1$ with $d_i \in [1, N-1]$ we get:

$$J_N^*(k_{i+1}) - J_N^*(k_i) \leq C_E(C_f)^{(N-(d_i))}\bar{\xi} + L_S - L_Q + (\frac{1}{B} - 1)\big(L_S + L_P + \max(p_1, \ldots, p_{2n})\varepsilon_0^2\big) \tag{5.23}$$

For a decreased Lyapunov function, it is obvious that the time instance d_i should be such that:

$$\sigma L_Q \geq C_E(C_f)^{(N-(d_i))}\bar{\xi} + L_S(m) + (\frac{1}{B} - 1)\big(L_S(m) + L_P(m) + \max(p_1, \ldots, p_{2n})\varepsilon_0^2\big) \tag{5.24}$$

with $\sigma \in (0, 1)$. Plugging (5.24) and (5.23), it yields:

$$J_N^*(k_{i+1}) - J_N^*(k_i) \leq (\sigma - 1)L_Q(m) \tag{5.25}$$

In view of (5.25), it can be concluded that the Lyapunov function $J_N^*(\cdot)$ has been proven to be decreasing, thus the closed-loop system converges to the compact set $\mathscr{E}_f$. Thus, the next visual measurement needs to be triggered when (5.24) is violated. The condition (5.24) should be checked for each consecutive time-step, i.e., for $d_i = 1, 2, \ldots$. The time-step that indicates this condition does no longer hold should be the next triggering instant k_{i+1}. This time-step k_{i+1} also can be found beforehand at time k_i, because the term $L_Q(m)$ and $L_P(m)$ include only predictions of the nominal system. The pseudo-code description of the proposed real-time self-triggering IBVS scheme is given in *Algorithm-1*. At time k_i we assume that the VTA is triggered, the OCP of (5.10)-(5.13) has run and the control trajectory $\boldsymbol{v}_f^*(k_i)$ for $[k_i, k_i + N - 1]$ is provided. The solution of (5.24) will provide the next update time k_{i+1}. During the time interval $i \in [k_i, k_{i+1})$ the control trajectory $\boldsymbol{v}^*(k_i + i|k_i)$ is applied to the system in an open-loop fashion. Next, at time k_{i+1}, the VTA is triggered again, the MPC provides a new control trajectory based on the current vision measurement $\boldsymbol{s}(k_{i+1})$ and the whole procedure is repeated again until

Algorithm 1 Self Triggered IBVS-MPC algorithm:

Triggering: ▷ At triggering time k_i
  $s(k_i) \leftarrow$ VTA ▷ Trigger the VTA, get $s(k_i)$
  $\boldsymbol{v}_f^*(k_i) \leftarrow \text{OCP}(s(k_i))$ ▷ Run OCP of (5.10)-(5.13)
  Solve eq.(5.24) **for** d_i ▷ Notice: $m = d_i - 1$
  $k_{i+1} = k_i + d_i$ ▷ The next triggering time
for $i = 1 \rightarrow d_i$ **do**
    Apply the $\boldsymbol{v}^*(k_i + i|k_i)$ to the robot.
goto *Triggering.*

the robotic camera system reaches to the desired position w.r.t target. We are now ready to state the result for this self-triggered IBVS-MPC framework:

Theorem 5.1. *Consider the Image Based Visual Servo system* (5.9) *that is subject to constraints* (5.7)(5.8) *and* (5.6)*. The triggering times* d_i *that are provided by* (5.24) *and the IBVS-MPC law provided by* (5.10)-(5.13) *which is applied to the robotic-camera system in an open-loop fashion during the inter-sampling periods, drive the closed-loop system towards the compact set* $\mathscr{E}_f$*, where it is ultimately bounded.*

Remark 5.1. It could be noticed that triggering times d_i that are provided by (5.24) and the IBVS-MPC law provided by (5.10)-(5.13) have been designed at camera frame. However, if the supposed camera system is located at the end effector of a robotic manipulator system (e.g., UVMS), the calculated control law can be expressed equivalently in the configuration space via:

$$\boldsymbol{\zeta}^d(t) = \boldsymbol{J}(\boldsymbol{q})^{\#}\boldsymbol{v}_e(\boldsymbol{s},t) + \big(\boldsymbol{I}_{n\times n} - \boldsymbol{J}(\boldsymbol{q})^{\#}\boldsymbol{J}(\boldsymbol{q})\big)\boldsymbol{v}_e(\boldsymbol{s},t)^0 \in \Re^n \tag{5.26}$$

where $\boldsymbol{q}$ is the vector of joint states of the robotic manipulator system (see Chapter 2), $\boldsymbol{J}(\boldsymbol{q})^{\#}$ denotes the generalized pseudo-inverse [219] of the Jacobian $\boldsymbol{J}(\boldsymbol{q})$, $\boldsymbol{v}_e(\boldsymbol{s},t)$ is the task-space version of control law calculated by (5.10)-(5.13) combined with (5.24) and $\boldsymbol{v}_e(\boldsymbol{s},t)^0$ denotes secondary tasks (e.g., maintaining manipulator's joint limits, increasing manipulability) to be regulated independently since they do not contribute to the end-effector's velocity [145] (i.e., they belong to the null space of the Jacobian $\boldsymbol{J}(\boldsymbol{q})$)[1].

5.5 Simulation Results

To validate the theoretical findings and verify the efficiency of the proposed ST-MPC-IBVS scheme, a comparative simulation study with a conventional MPC-IBVS controller was conducted using the dynamic simulation environment built in MATLAB® presented in Section 2.4, where we assumed that the supposed camera system is located at the

[1]For more details on task priority based control and redundancy resolution for UVMSs the reader is referred to [145] and [151].

end effector of the UVMS (see Fig. 5.3). Therefore, the calculated camera based control inputs is expressed into joint space using the (5.26) (see Remark-5.1). a simple model based PD controller is designed and employed in dynamic level in order to track that control inputs at joint space level.

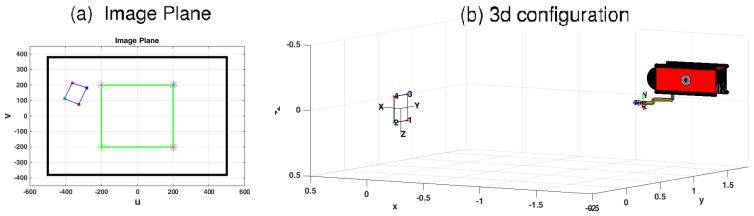


FIGURE 5.3: The simulation setup comprises a camera, attached at the end effector of a UVMS, observing a planar target. The initial and the desired position of the target on the image plane are illustrated by blue and green lines respectively. (b) The initial and the desired pose of the camera attached at the UVMS end effector.

In the following simulations, for both cases of Self-Triggered-IBVS-MPC and Classic IBVS-MPC, the sub-optimality rate is considered to be equal to 0.7 (B=0.7). Moreover, the sampling period was equal to 0.033 sec, representing in this way a real time operation with a pinhole camera of 30 FPS. Furthermore, in both cases, the prediction horizon and the focal length of the camera are equal to $N = 15$ and $\lambda = 0.001$, respectively. The target comprised of four feature points on a vertical plane, forming a square with edge 0.1 m. The desired pose of the target with respect to the camera frame O_c was $\boldsymbol{p}_{d/O_c} = [0\ 0\ 0.5\ 0\ 0\ 0]^\top$. It should be noticed that the initial pose of the target with respect to the camera frame (i.e., $\boldsymbol{p}_{init/O_c} = [1.3, 0.0, 1.37, -0.4, 0.0, -0.4]^\top$ was a rather difficult pose for conventional image based visual servoing control [220] (see Fig 5.3). The desired and initial feature coordinates (i.e., $\boldsymbol{s}_{in}$ and $\boldsymbol{s}^*$) at the initial and desired pose of the camera are:

$$\boldsymbol{s}^* = \begin{bmatrix} -200 & 200 & -200 & 200 \\ -200 & -200 & 200 & 200 \end{bmatrix}, \qquad \boldsymbol{s}_{in} = \begin{bmatrix} 327 & 406 & 282 & 363 \\ 75 & 111 & 182 & 213 \end{bmatrix},$$

The visibility constraints of the system are defined as:

$$\begin{bmatrix} u = -500 \\ v = -380 \end{bmatrix} \leq \begin{bmatrix} u(t) \\ v(t) \end{bmatrix} \leq \begin{bmatrix} \bar{u} = 500 \\ \bar{v} = 380 \end{bmatrix}$$

In addition, for both of the following simulations, the matrices $\boldsymbol{P}$, $\boldsymbol{Q}$ and $\boldsymbol{R}$ of of the OCP (5.10)-(5.13) are taken as identity matrices of appropriate dimension. Also the the predefined upper bound of the camera velocity (i.e., control inputs) is bounded by $0.1\frac{m}{sec}$ for the translational and $0.1\frac{rad}{sec}$ for rotational velocity. Finally, In order to test the robustness of the proposed scheme, in all subsequent simulations studies, the

dynamics of the UVMS were affected by external disturbances in the form of slowly time varying sea currents modeled by the corresponding velocities $v_x^c = 0.1\sin(\frac{\pi}{30}t)\frac{m}{s}$ and $v_y^c = 0.1\cos(\frac{\pi}{30}t)\frac{m}{s}$.

5.5.1 Simulation Results

The comparison was performed via two simulation procedures. In the first experiment a conventional IBVS-MPC scheme was employed, while in the second experiment, the Self Triggered IBVS-MPC scheme proposed in this chapter was employed. Through both simulations procedures, comparisons are made to show the efficacy and performance of the proposed self triggered IBVS-MPC with respect to the classic IBVS-MPC. In Fig.5.4, the evolution of UVMS/Camera system in 3D space employing the proposed ST-IBVS-MPC and the Classic time triggered IBVS-MPC are illustrated. It can be seen that in both cases, the UVMS/Camera system has reached to the desired position with respect to the object. In Fig.5.5 the image error evolution in both experiments are depicted.

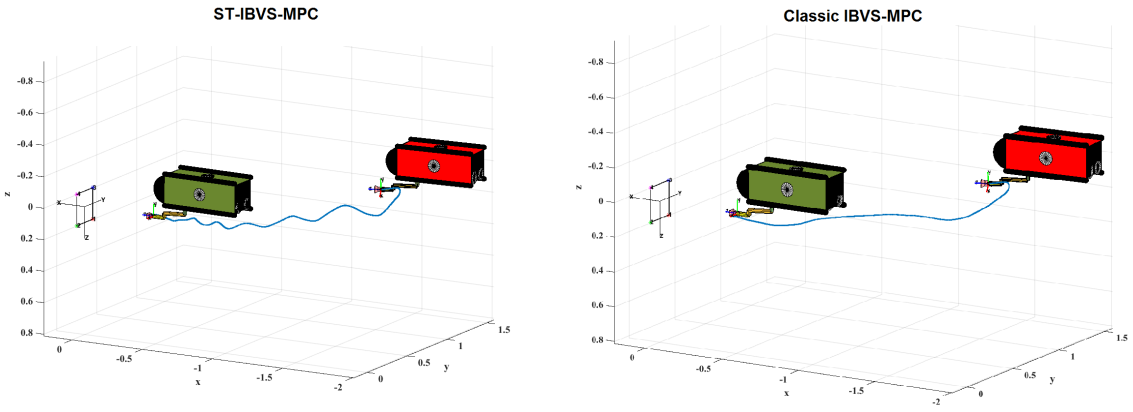


FIGURE 5.4: The evolution of UVMS/Camera system in 3D space, (a) ST-IBVS-MPC, (b) Classic IBVS-MPC. The initial and final position of the UVMS/Camera system is indicated by red and green color respectively.

It can be witnessed that in both cases, the image errors converge to zeros. As it was expected, the features were constrained within the camera FOV as presented in Fig. 5.6. In Fig.5.7, the triggering instants in the case of ST-IBVS-MPC are captured. When the vertical axis has the value 1, the Visual Tracking Algorithm is triggered and a new image vector is calculated, consequently the IBVS-MP Controller is running and a new control input trajectory is computed. For value 0, the control law is implemented on the system in an open-loop fashion using the rest of the last computed control input trajectory, thus no optimization and no Visual Tracking Algorithm is running. Notice that in the case of classic IBVS-MPC, the VTA is triggered at each sampling time. It can be noticed that the triggering instants when the camera is close to the desired position w.r.t the target

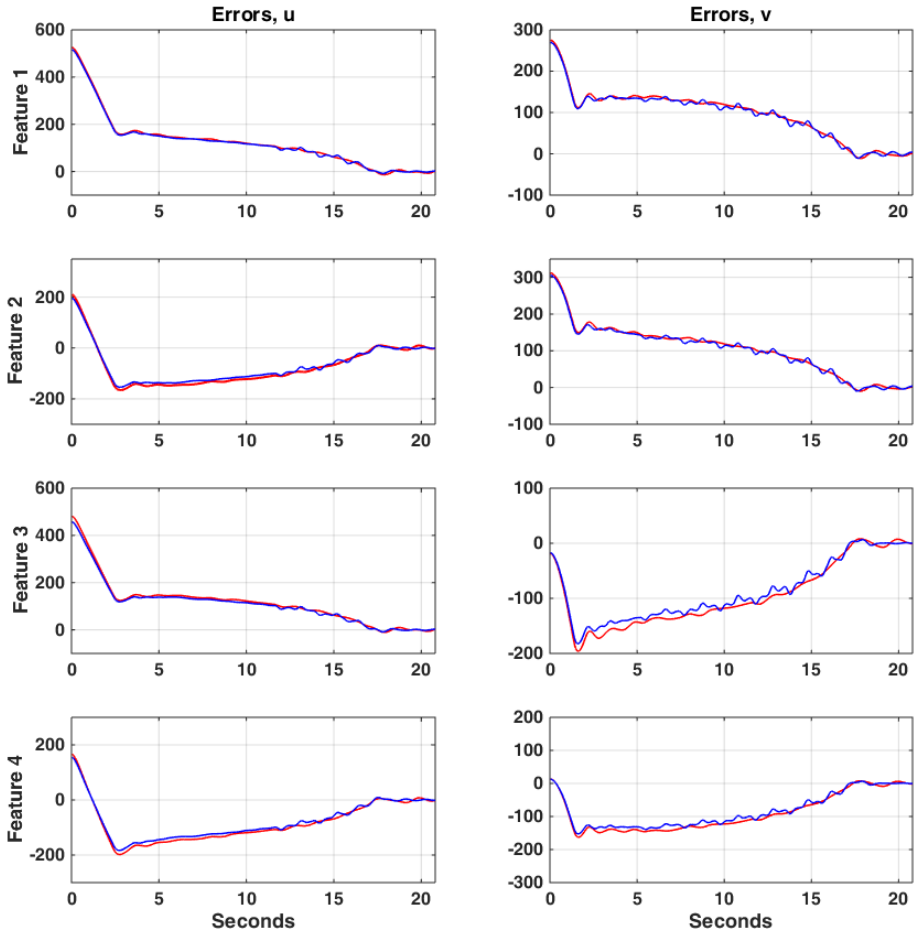


FIGURE 5.5: The feature coordinate errors: the ST-IBVS-MPC and Classic IBVS-MPC are presented with blue and red color respectively.

are more frequent. This fact is easily interpretable, because near the desired position, the system becomes more demanding due to the FoV constraints and the disturbances, thus the need for new measurements and calculation of a new control input is increased. However, using the self triggered condition proposed in this work, the triggering of the VTA and the MPC controller have been reduced by 57% (223 triggering instants instead of 520) w.r.t the traditional IBVS-MPC. Finally, the control input signals are illustrated in Fig. 5.8.

5.5.2 Video

The proposed ST-IBVS-MPC control scheme was tested and compared experimentally with time triggered IBVS-MPC using a YouBot mobile-manipulator system equipped with a USB camera, validating the theoretical findings of this chapter. That experimental

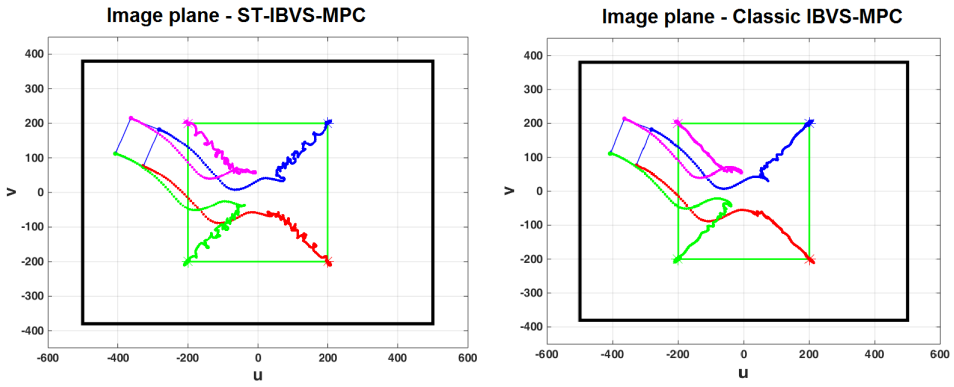


FIGURE 5.6: The evolution of the features on the image plane. (a) ST-IBVS-MPC, (b) Classic IBVS-MPC. The desired position of the features on the image plane is denoted by cubes.

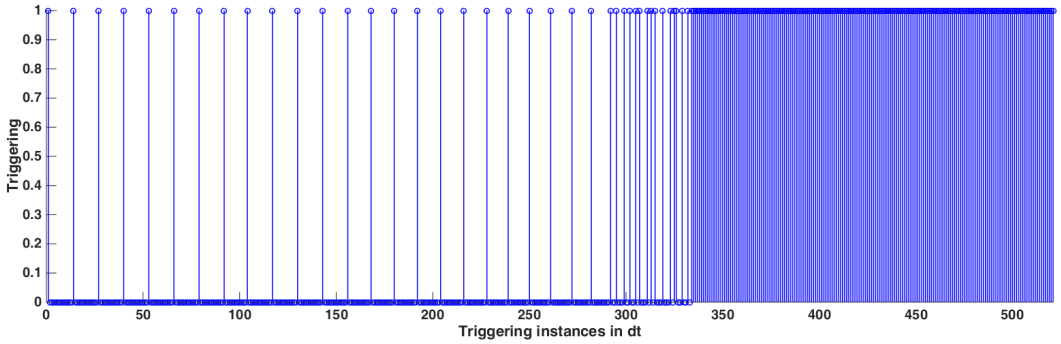


FIGURE 5.7: The triggering instants for the case of ST-IBVS-MPC. For value 1, The VTA is triggered. For value 0 the control law is implemented in an open-loop fashion.

study can be found at the following url: https://www.youtube.com/watch?v=C4_zcFQiJt8. Alternatively, it can be found in the attached dvd as file: video_ch5.mpg

5.6 Conclusions

In this chapter a novel image based visual servoing-model predictive control scheme is presented. A mechanism for the decision of the VTA triggering and the calculation of a new control input is designed and implemented. This results in aperiodic triggering instants only when needed and not periodically as in the case of the classic IBVS controller. During the inter-sampling instants, the control sequence calculated by controller is applied to the camera robotic system in a open-loop fashion, i.e., neither VTA nor the MPC are running between two triggering instants. This results in the reduction of the computational effort, energy consumption and increases in this way the autonomy rate of the system. Thus it can be used effectively in Underwater Vehicle Manipulator

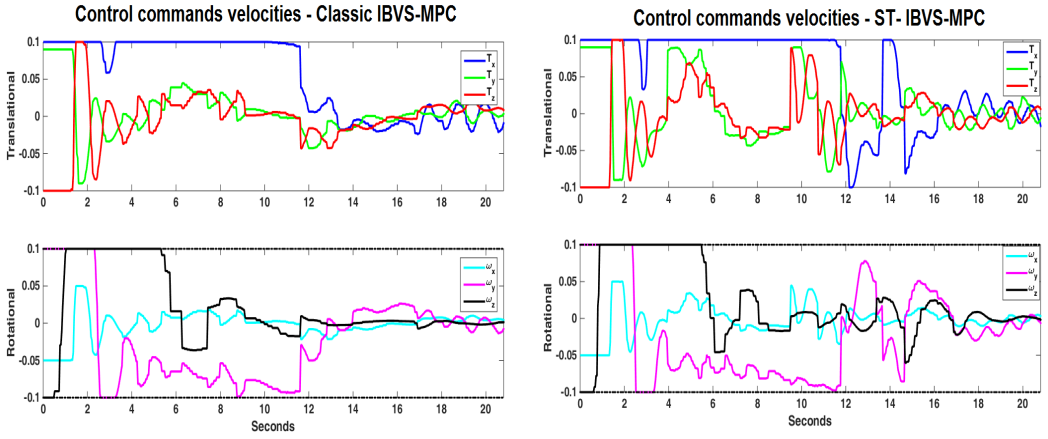


FIGURE 5.8: The control input signals.

systems which perform long lasting inspection tasks, where low energy consumption and high system autonomy are required.

Chapter 6

Robust Image Based Visual Servoing with Prescribed Performance under Field of View Constraints

Previously in Chapter 4 we proposed a Position Based Visual Servoing (PBVS) scheme for case when the relative position between the robot and object can be obtained real-time (e.g., by observing a known marker on the object and estimating the relative Cartesian pose). Then, in Chapter 5 we extended this work to an Image Based Visual Servoing (IBVS) scheme for case that the relative position is not available online. In such a case, a desired image is considered to be available and the controller is designed at image-plane coordinates. However, the proposed controller of the Chapter 5 requires that the intrinsic camera parameters are available during the control operation. We have assumed in Chapter 5 that these camera parameters have been acquired via a calibration process in prior. In this chapter we propose a model free IBVS control strategy which satisfies the visibility constraints for case when the camera parameters of the vision system are not available ahead of time. The proposed controller in this chapter similarly in to Chapter 5 is designed at the camera frame. Thus, the camera could be located either at the end effector or at some point on the underwater robotic vehicle's body. However, in the following simulation studies, we have attached the camera at the UVMS end effector frame.

6.1 Introduction

It is well known that IBVS does not require the geometric model of the target. However, the image jacobian that is employed involves the intrinsic camera parameters and the depth of the image features with respect to the camera frame. Therefore, accurate camera parameters, which may be acquired via a calibration process, are crucial for the closed loop system performance and stability [221, 222]. However, even perfect calibration does not eliminate the image resolution issue that causes extra uncertainty in the control [223]. In addition, camera calibration is in general a tedious procedure [221] and its precision may vary. Unfortunately, most of the aforementioned strategies rely heavily on the knowledge of the 3D reconstruction and camera calibration parameters. Even though in some of the aforementioned control strategies the camera calibration uncertainty has been considered, they are highly affected by uncertainties on environment modeling. In addition, most of these approaches adopt off-line calculations, which cannot be performed easily in real time systems in the presence of 3D model reconstruction errors. Thus, their applicability in real time tasks renders questionable. Consequently, recent research studies have been focused on online trajectory generation for vision based systems [92, 94].

Alternatively, various approaches such as adaptive control [221, 224–228], calibration free path planning [91, 92], online identification[229] and machine learning [230] have been proposed in the related literature. More specifically, a depth-independent interaction matrix has been adopted in [221] that renders the unknown camera parameters appear linearly in the closed loop dynamics and subsequently an adaptive scheme is developed to estimate them. The aforementioned work was extended later in [228] and [227] to cope with features trajectory tracking for eye-in-hand robotic systems. Alternatively, robust path planning algorithms have been proposed in [71, 222, 231, 232] to achieve the desired stability properties despite any calibration or depth measurement errors. Concurrently, other calibration free solutions have been presented in [92, 94, 233–235].

Another important issue associated with IBVS schemes concerns the transient and steady state response of the closed loop system. Unfortunately, apart from the [235][1], where bounds on the task error have been addressed, the related literature lacks of any systematic procedure that imposes accurately predefined transient and steady state performance specifications. Towards this direction, the common practice in conventional IBVS schemes is to tune appropriately the control gains via a tedious trial and error procedure without, however, any a priori guarantees for the achieved performance. It

[1] In this work, a strategy for modulating the control gains is proposed to meet a predefined upper bound on the tracking error.

should be noted that owing to the presence of multiple and probably conflicting operational constraints (i.e., field of view, transient and steady state specifications, model imperfections) that increase significantly the complexity of the IBVS problem, no results have been previously reported in the related literature, up to the best of the authors' knowledge.

In this chapter, motivated by the prescribed performance control technique [236], a novel IBVS scheme is proposed, capable of guaranteeing prescribed transient and steady state performance as well as the satisfaction of the field of view constraints, despite the inevitable camera calibration and depth measurement errors. Visualizing the performance specifications and the field of view constraints as error bounds, the key idea is to provide an error transformation that converts the original constrained model into an equivalent unconstrained one. It is then proven that stabilizing the unconstrained model is sufficient to achieve prescribed performance guarantees and satisfy the field of view constraints. Moreover, the performance of the developed scheme is a priori and explicitly imposed by certain designer-specified performance functions, and is fully decoupled by the control gains selection. In that respect, the selection of the control gains is only confined to adopting those values that lead to reasonable control effort, thus simplifying further the control design. Finally, the computational complexity of the proposed scheme proves considerably low (i.e., it is a static scheme involving very few and simple calculations to output the control signal), which makes implementation on fast embedded control platforms straightforward. As it is mentioned previously, the proposed controller is design at the camera frame. Thus, be located either at the end effector or at some point on the underwater robotic vehicle's body.

The remainder of this chapter is structured as follows. The adopted notation, which is necessary throughout the manuscript, is reviewed in the sequel. Section 6.2 formulates the problem. The main results are presented in Section 6.3. Simulation studies, clarifying and verifying the approach, are given in Section 6.4. Finally, conclusions are drawn in Section 6.5.

Notation

The n-th dimensional Euclidean space is denoted by $\Re^n$. The set of all $n \times m$ real matrices is represented by $\Re^{n \times m}$. The absolute value of a scalar $a \in \Re$ and the Euclidean norm of a vector $\boldsymbol{a} \in \Re^n$ are denoted by $|a|$ and $\|\boldsymbol{a}\|$ respectively. The function $\text{diag}\,(\boldsymbol{a})$, where $\boldsymbol{a} \triangleq [a_1, \ldots, a_n]^T \in \Re^n$ is an n-th dimensional vector, denotes the $n \times n$ diagonal matrix of the elements $a_1, \ldots, a_n$. Finally, $\mathbf{0}_{n \times m} \in \Re^{n \times m}$ represents the zero elements $n \times m$ matrix.

6.2 Problem Statement

In this section, the mathematical formulation of the image based visual servoing problem is presented for a pinhole camera model. Let $[X_c, Y_c, Z_c]^T$ be the axes of the camera frame $\mathscr{C}$ attached at the center of the camera O_c. The coordinates of the image frame $\mathscr{I}$ are given by $[u, v]^T$ with O_I denoting the center of the image, as depicted in Fig. 6.1. Notice that the Z_c axis of the camera frame is perpendicular to the image plane transversing O_I. Thus, given a set of n fixed 3D points $\boldsymbol{P}_i = [x_i, y_i, z_i]^T$, $i = 1, \ldots, n$ expressed in

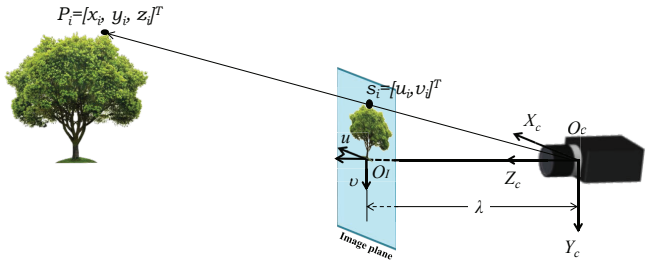


FIGURE 6.1: The geometric model of a pinhole camera.

the camera frame, the corresponding 2D image features $\boldsymbol{s}_i = [u_i, v_i]^T$, $i = 1, \ldots, n$ are given (in pixels) as follows [68]:

$$\boldsymbol{s}_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \frac{\lambda}{z_i} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{6.1}$$

where λ is the focal length of the camera (see Fig.6.1). In this way, the effect of the camera motion on the feature coordinates at the image plane is modeled by:

$$\dot{\boldsymbol{s}}_i = \boldsymbol{L}_i(z_i, s_i)\boldsymbol{v},\ i = 1, \ldots, n \tag{6.2}$$

where:

$$\boldsymbol{L}_i\left(z_i, s_i\right) = \begin{bmatrix} -\frac{\lambda}{z_i} & 0 & \frac{u_i}{z_i} & \frac{u_i v_i}{\lambda} & -\frac{\lambda^2+u_i^2}{\lambda} & v_i \\ 0 & -\frac{\lambda}{z_i} & \frac{v_i}{z_i} & \frac{\lambda^2+v_i^2}{\lambda} & -\frac{u_i v_i}{\lambda} & -u_i \end{bmatrix} \tag{6.3}$$

is the interaction matrix [68], and

$$\boldsymbol{v} = \begin{bmatrix} \boldsymbol{T} \\ \boldsymbol{\Omega} \end{bmatrix} = [T_x, T_y, T_z, \omega_x, \omega_y, \omega_z]^T$$

denotes the linear $\boldsymbol{T}$ and angular $\boldsymbol{\Omega}$ velocities of the camera. Let us also define the overall image feature vector $\boldsymbol{s} = \left[s_1^T, \cdots, s_n^T\right]^T \in \Re^{2n}$. Hence, the dynamics of the feature coordinates is given by:

$$\dot{\boldsymbol{s}} = \boldsymbol{L}(\boldsymbol{z}, \boldsymbol{s})\boldsymbol{v} \tag{6.4}$$

where $\boldsymbol{L}(\boldsymbol{z},\boldsymbol{s}) = \left[\boldsymbol{L}_1^T(z_1,\boldsymbol{s}_1),\cdots,\boldsymbol{L}_n^T(z_n,\boldsymbol{s}_n)\right]^T \in \Re^{2n\times 6}$ is the overall interaction matrix and $\boldsymbol{z} = [z_1,\ldots,z_n]^T$.

Owing to the limited field of view of the camera, the image coordinates are subject to the following visibility constraints:

$$u_{\min} \leq u_i \leq u_{\max},\ i = 1,\ldots,n \tag{6.5}$$

$$v_{\min} \leq v_i \leq v_{\max},\ i = 1,\ldots,n \tag{6.6}$$

where $u_{\min}$, $v_{\min}$ and $u_{\max}$, $v_{\max}$ denote the lower and upper bounds (in pixels) of the image plane coordinates, dictated by the camera resolution. Ensuring that the feature coordinates do not violate the aforementioned visibility constraints and therefore they constantly lie within the camera field of view is an issue of paramount importance in visual servoing, since otherwise unpredictable phenomena (even instability) may occur in the closed loop system owing to the partial or total loss of visual feedback.

In this chapter, the control objective is to design an image based visual servoing scheme such that all feature coordinates $\boldsymbol{s}_i = [u_i, v_i]^T$, $i = 1,\ldots,n$ converge to their corresponding desired values $\boldsymbol{s}_i^d = [u_i^d, v_i^d]^T$, $i = 1,\ldots,n$ with prescribed transient and steady state performance, despite the inevitable camera calibration and depth measurement errors (i.e., the focal length λ and the features depth z_i, $i = 1,\ldots,n$ are not accurately computed). By prescribed performance, we mean that the desired feature coordinates $\boldsymbol{s}_i^d$, $i = 1,\ldots,n$ are attained in a predefined transient period and are maintained with arbitrarily fine accuracy, while satisfying the field of view constraints (6.5) and (6.6) for all time.

6.3 Main Results

The prescribed performance control technique, originally proposed to design robust state feedback controllers for various classes of nonlinear systems [236–238], will be adopted to: i) achieve robust predefined transient as well as steady state response for all image feature errors and ii) avoid the violation of the camera field of view constraints.

6.3.1 Sufficient Conditions

Let us initially define the image feature errors:

$$e_i^u(t) = u_i(t) - u_i^d,\ i = 1,\ldots,n \tag{6.7}$$

$$e_i^v(t) = v_i(t) - v_i^d,\ i = 1,\ldots,n \tag{6.8}$$

where u_i^d, v_i^d denote the corresponding desired feature values, as well as the overall error vector:

$$\boldsymbol{e} = [e_1^u, e_1^v, \ldots, e_n^u, e_n^v]^T .$$

Prescribed performance characterizes the behavior when the image feature errors $e_i^u(t)$, $e_i^v(t)$, $i = 1, \ldots, n$ evolve strictly within predefined regions that are bounded by absolutely decaying functions of time, called performance functions. In this chapter, the mathematical expression of prescribed performance is formulated, for all $t \geq 0$, by the following inequalities:

$$-\underline{M}_i^u \rho_i^u(t) < e_i^u(t) < \bar{M}_i^u \rho_i^u(t), \; i = 1, \ldots, n \tag{6.9}$$

$$-\underline{M}_i^v \rho_i^v(t) < e_i^v(t) < \bar{M}_i^v \rho_i^v(t), \; i = 1, \ldots, n \tag{6.10}$$

where:

$$\rho_i^u(t) = \left(1 - \frac{\rho_\infty}{\max\{\underline{M}_i^u, \bar{M}_i^u\}}\right) \exp(-lt) + \frac{\rho_\infty}{\max\{\underline{M}_i^u, \bar{M}_i^u\}} \tag{6.11}$$

$$\rho_i^v(t) = \left(1 - \frac{\rho_\infty}{\max\{\underline{M}_i^v, \bar{M}_i^v\}}\right) \exp(-lt) + \frac{\rho_\infty}{\max\{\underline{M}_i^v, \bar{M}_i^v\}} \tag{6.12}$$

are designer-specified smooth, bounded and decreasing functions of time with l, $\rho_\infty > 0$

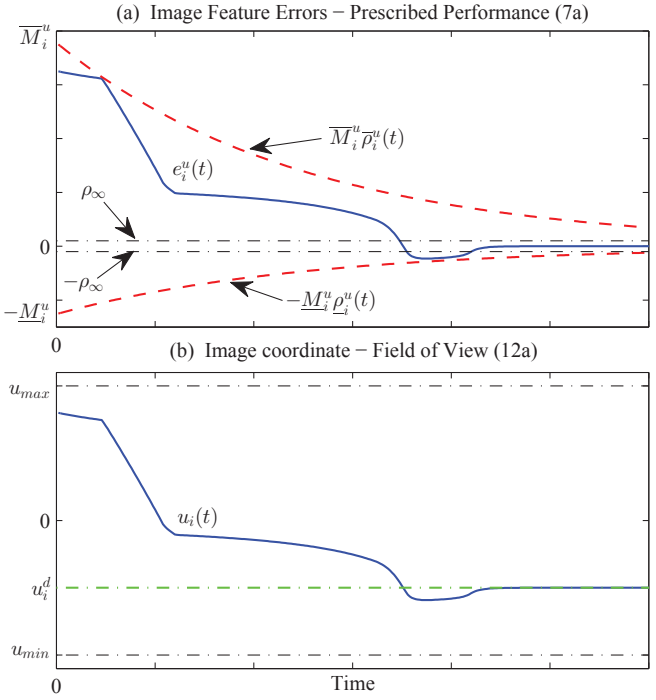


FIGURE 6.2: The graphical illustration of prescribed performance (6.9) and field of view constraints (6.19).

incorporating the desired transient and steady state performance specifications respectively, and $\underline{M}_i^u,\ \bar{M}_i^u,\ \underline{M}_i^v,\ \bar{M}_i^v$ are positive parameters selected appropriately to satisfy the field of view constraints, as presented in the sequel. In particular, the decreasing rate of $\rho_i^u(t),\ \rho_i^v(t)$, which is affected by the parameter l, introduces a lower bound on the speed of convergence of $e_i^u(t),\ e_i^v(t),\ i = 1, \ldots, n$. Furthermore, depending on the resolution of the camera, the constant ρ_∞ can be set arbitrarily small $\rho_\infty << \min_{i=1,\ldots,n} \{\underline{M}_i^u,\ \bar{M}_i^u,\ \underline{M}_i^v,\ \bar{M}_i^v\}$, thus achieving practical convergence of $e_i^u(t),\ e_i^v(t)$, $i = 1, \ldots, n$ to zero. Additionally, we select:

$$\underline{M}_i^u = u_i^d - u_{\min}\ \&\ \bar{M}_i^u = u_{\max} - u_i^d,\ i = 1, \ldots, n \tag{6.13}$$

$$\underline{M}_i^v = v_i^d - v_{\min}\ \&\ \bar{M}_i^v = v_{\max} - v_i^d,\ i = 1, \ldots, n \tag{6.14}$$

Apparently, under the assumption that the features initially lie in the camera field of view (i.e., $u_{\min} < u_i(0) < u_{\max}$ and $v_{\min} < v_i(0) < v_{\max},\ i = 1, \ldots, n$) the aforementioned selection ensures that:

$$-\underline{M}_i^u \rho_i^u(0) < e_i^u(0) < \bar{M}_i^u \rho_i^u(0),\ i = 1, \ldots, n \tag{6.15}$$

$$-\underline{M}_i^v \rho_i^v(0) < e_i^v(0) < \bar{M}_i^v \rho_i^v(0),\ i = 1, \ldots, n \tag{6.16}$$

Hence, guaranteeing prescribed performance via (6.9) and (6.10) for all $t > 0$ and employing the decreasing property of $\rho_i^u(t),\ \rho_i^v(t),\ i = 1, \ldots, n$, we obtain:

$$-\underline{M}_i^u < e_i^u(t) < \bar{M}_i^u,\ i = 1, \ldots, n \tag{6.17}$$

$$-\underline{M}_i^v < e_i^v(t) < \bar{M}_i^v,\ i = 1, \ldots, n \tag{6.18}$$

and consequently owing to (6.7)-(6.8) and (6.13)-(6.14):

$$u_{\min} < u_i(t) < u_{\max}, i = 1, \ldots, n \tag{6.19}$$

$$v_{\min} < v_i(t) < v_{\max}, i = 1, \ldots, n \tag{6.20}$$

for all $t > 0$, which ensures that the field of view constraints are constantly satisfied. Therefore, imposing prescribed performance via (6.9) and (6.10) with appropriately selected performance functions $\rho_i^u(t),\ \rho_i^v(t),\ i = 1, \ldots, n$ and positive constant parameters $\underline{M}_i^u,\ \bar{M}_i^u,\ \underline{M}_i^v,\ \bar{M}_i^v,\ i = 1, \ldots, n$, as dictated in (6.13) and (6.14) respectively, proves sufficient to solve the image based visual servoing problem stated in Section II. The aforementioned statements are clearly illustrated in Fig.6.2 for exponentially decreasing performance functions $-\underline{M}_i^u \rho_i^u(t),\ \bar{M}_i^u \rho_i^u(t)$ with $\underline{M}_i^u,\ \bar{M}_i^u$ satisfying (6.13) and $l,\ \rho_\infty$ appropriately selected positive constants that determine the desired transient and steady state performance specifications.

6.3.2 Control Design

In the sequel, we propose a control protocol that incorporates neither accurate depth measurements nor accurate focal length estimation, and guarantees (6.9) and (6.10) for all $t \geq 0$, thus leading to the solution of the robust image based visual servoing problem with prescribed performance under field of view constraints. Given the image feature tracking errors $e_i^u(t)$, $e_i^v(t)$, $i = 1, \ldots, n$, defined in (6.7) and (6.8), we select the corresponding performance functions $\rho_i^u(t)$, $\rho_i^v(t)$ and positive parameters $\underline{M}_i^u$, $\bar{M}_i^u$, $\underline{M}_i^v$, $\bar{M}_i^v$, $i = 1, \ldots, n$ following (6.13) and (6.14) respectively, in order to incorporate the desired transient and steady state performance specifications as well as the field of view constraints. We define the normalized image feature errors as:

$$\xi_i^u(u_i, t) = \frac{e_i^u}{\rho_i^u(t)}, \; i = 1, \ldots, n$$
$$\xi_i^v(v_i, t) = \frac{e_i^v}{\rho_i^v(t)}, \; i = 1, \ldots, n$$

and the transformed image feature errors as:

$$E_i^u\left(\xi_i^u(u_i, t)\right) = \ln\left(\frac{1 + \frac{\xi_i^u(u_i,t)}{\underline{M}_i^u}}{1 - \frac{\xi_i^u(u_i,t)}{\bar{M}_i^u}}\right), \; i = 1, \ldots, n$$
$$E_i^v\left(\xi_i^v(v_i, t)\right) = \ln\left(\frac{1 + \frac{\xi_i^v(v_i,t)}{\underline{M}_i^v}}{1 - \frac{\xi_i^v(v_i,t)}{\bar{M}_i^v}}\right), \; i = 1, \ldots, n.$$

for which $e_i^u \to 0$ ($e_i^v \to 0$) implies $E_i^u \to 0$ ($E_i^v \to 0$), $i = 1, \ldots, n$. Finally, we design the image based visual servoing protocol as follows:

$$\boldsymbol{v}(\boldsymbol{s}, t) = -k\hat{\boldsymbol{L}}^+ \boldsymbol{E}(\boldsymbol{s}, t) \text{ with } k > 0 \tag{6.21}$$

where $\hat{\boldsymbol{L}}^+$ is the pseudo-inverse of the estimated interaction matrix [231] and:

$$\boldsymbol{E}(\boldsymbol{s}, t) = \begin{bmatrix} E_1^u\left(\xi_1^u(u_1, t)\right) \\ E_1^v\left(\xi_1^v(v_1, t)\right) \\ \vdots \\ E_n^u\left(\xi_n^u(u_n, t)\right) \\ E_n^v\left(\xi_n^v(v_n, t)\right) \end{bmatrix}. \tag{6.22}$$

Remark 6.1. The prescribed performance control technique enforces the normalized image feature errors $\xi_i^u(t)$ and $\xi_i^v(t)$ to remain strictly within the sets $\left(-\underline{M}_i^u, \; \bar{M}_i^u\right)$ and $\left(-\underline{M}_i^v, \; \bar{M}_i^v\right)$, $i = 1, \ldots, n$ respectively for all $t > 0$. Notice that modulating $\xi_i^u(t)$ and

$\xi_i^v(t)$ via the logarithmic functions $\ln\left(\frac{1+\frac{\star}{\underline{M}_i^u}}{1-\frac{\star}{\bar{M}_i^u}}\right)$ and $\ln\left(\frac{1+\frac{\star}{\underline{M}_i^v}}{1-\frac{\star}{\bar{M}_i^v}}\right)$ in the control protocol (6.21) and selecting $\underline{M}_i^u,\ \bar{M}_i^u,\ \underline{M}_i^v,\ \bar{M}_i^v$ according to (6.13) and (6.14), the overall transformed image feature error vector (6.22) is initially well-defined. Moreover, it is not difficult to verify that maintaining simply the boundedness of the modulated errors $E_i^u(u_i,t)$ and $E_i^v(v_i,t)$ for all $t>0$ is equivalent to guaranteeing $\xi_i^u(u_i,t)\in\left(-\underline{M}_i^u,\ \bar{M}_i^u\right)$ and $\xi_i^v(v_i,t)\in\left(-\underline{M}_i^v,\ \bar{M}_i^v\right)$, $i=1,\ldots,n$ for all $t>0$. Therefore, the problem at hand can be simply visualized as stabilizing the modulated error vector $E(s,t)$. A careful inspection of the proposed control scheme (6.21) reveals that it actually operates similarly to barrier functions in constrained optimization, admitting high negative or positive values depending on whether $e_i^u(t)\to-\underline{M}_i^u\rho_i^u(t)$ and $e_i^v(t)\to-\underline{M}_i^v\rho_i^v(t)$ or $e_i^u(t)\to\bar{M}_i^u\rho_i^u(t)$ and $e_i^v(t)\to\bar{M}_i^v\rho_i^v(t)$, $i=1,\ldots,n$ respectively; eventually preventing $e_i^u(t)$ and $e_i^v(t)$, $i=1,\ldots,n$ from reaching the corresponding boundaries.

Remark 6.2. Regarding the construction of the performance functions, we stress that unlike what is common practice in the related literature, the desired performance specifications concerning the transient and steady state response as well as the field of view constraints are introduced directly in the proposed control scheme via $\rho_i^u(t),\ \rho_i^v(t)$ and the positive parameters $\underline{M}_i^u,\ \bar{M}_i^u,\ \underline{M}_i^v,\ \bar{M}_i^v$, $i=1,\ldots,n$ respectively. In particular, the decreasing rate l of the performance functions $\rho_i^u(t),\ \rho_i^v(t)$ introduces a lower bound on the speed of convergence of the image feature errors. Furthermore, ρ_∞ regulates the maximum allowable size of the image feature errors at steady state. Moreover, the parameters $\underline{M}_i^u,\ \bar{M}_i^u,\ \underline{M}_i^v,\ \bar{M}_i^v$ are selected via (6.13) and (6.14) in order to incorporate the field of view constraints. Hence, the performance attributes of the proposed control protocol are selected a priori, in accordance to the desired transient and steady state performance specifications. In this way, the selection of the control gain k, that has been isolated from the actual control performance, is significantly simplified to adopting those values that lead to reasonable control effort. Nevertheless, it should be noted that the gain selection affects both the quality of evolution of the image feature errors inside the corresponding performance envelopes as well as the control input characteristics (i.e., decreasing the gain value leads to increased oscillatory behaviour within the prescribed performance envelope, which is improved when adopting higher values, enlarging, however, the control effort both in magnitude and rate). Additionally, fine tuning might be needed in real-time scenarios, to retain the required control input signals within the feasible range that can be implemented by the actuators. Similarly, the control input constraints impose an upper bound on the required speed of convergence of the performance functions $\rho_i^u(t),\ \rho_i^v(t)$, as obtained by the exponential $\exp(-lt)$. Therefore, the selection of the control gain k can have positive influence on the overall closed loop system response. Finally, the computational complexity of the proposed scheme proves considerably low. It is actually a static scheme involving very few and simple calculations

to output the control signal, which enables easily its implementation on fast embedded control platforms.

Remark 6.3. Notice from (6.3) that $\boldsymbol{L}$ depends on the depth distribution $\boldsymbol{z}$ of the image features as well as on the camera focal length λ. Thus, considering camera calibration and depth measurement errors, the matrix $\hat{\boldsymbol{L}}$, which is employed in the control design, is an estimate of the actual interaction matrix $\boldsymbol{L}(\boldsymbol{z}, \boldsymbol{s})$. Under the assumption that L is full rank, which is rather realistic in the context of visual servoing, the task function parametrization approach, which aims at providing a controllable system over the task workspace, employs the estimated interaction matrix $\hat{\boldsymbol{L}}$ that is also full rank for reasonable focal length and depth estimates. A common approach is to employ the depth distribution at the desired pose with a rough estimate of the focal length via an initial calibration procedure. Alternatively, an estimate of the depth up to a scalar factor could be adopted, owing to the fact that the visual features are fixed in the workspace.

Remark 6.4. It could be noticed that the calculated control input of (6.21) is designed at camera frame. However, if the supposed camera system is located at the end effector of a robotic manipulator system (e.g., UVMS), the (6.21) can be expressed equivalently in the configuration space via:

$$\boldsymbol{\zeta}^d(t) = \boldsymbol{J}(\boldsymbol{q})^{\#}\boldsymbol{v}_e(\boldsymbol{s}, t) + \big(\boldsymbol{I}_{n\times n} - \boldsymbol{J}(\boldsymbol{q})^{\#}\boldsymbol{J}\big(\boldsymbol{q}\big)\big)\boldsymbol{v}_e(\boldsymbol{s}, t)^0 \in \Re^n \tag{6.23}$$

where $\boldsymbol{q}$ is the vector of joint states of the robotic manipulator system (see Chapter 2), $\boldsymbol{J}(\boldsymbol{q})^{\#}$ denotes the generalized pseudo-inverse [219] of the Jacobian $\boldsymbol{J}(\boldsymbol{q})$, $\boldsymbol{v}_e(\boldsymbol{s}, t)$ is the task-space version of $\boldsymbol{v}(\boldsymbol{s}, t)$ calculated by (6.21) and $\boldsymbol{v}_e(\boldsymbol{s}, t)^0$ denotes secondary tasks (e.g., maintaining manipulator's joint limits, increasing manipulability) to be regulated independently since they do not contribute to the end-effector's velocity [145] (i.e., they belong to the null space of the Jacobian $\boldsymbol{J}(\boldsymbol{q})$)[2].

6.3.3 Stability Analysis

The main results of this chapter are summarized in the following theorem, where it is proven that the aforementioned control protocol solves the image based visual servoing problem with prescribed performance under field of view constraints, despite the inevitable camera calibration and depth measurement errors.

Theorem 6.1. *Consider n fixed visual features in the workspace and a pinhole camera that aims at attaining desired values for the feature coordinates on the image plane, while satisfying its own field of view constraints. Under the assumption that all visual*

[2]For more details on task priority based control and redundancy resolution for UVMSs the reader is referred to [145] and [151].

features initially lie within the field of view of the camera, the proposed image based visual servoing protocol (6.21) guarantees local practically asymptotic stabilization of the feature errors:

$$-\underline{M}_i^u \rho_i^u(t) < e_i^u(t) < \bar{M}_i^u \rho_i^u(t),\ i = 1, \ldots, n \tag{6.24}$$

$$-\underline{M}_i^v \rho_i^v(t) < e_i^v(t) < \bar{M}_i^v \rho_i^v(t),\ i = 1, \ldots, n \tag{6.25}$$

for all $t \geq 0$, as well as the boundedness of all closed loop signals.

Proof. We first define the normalized image feature error vector:

$$\boldsymbol{\xi} = [\xi_1^u, \xi_1^v, \ldots, \xi_n^u, \xi_n^v]^T .$$

Differentiating with respect to time and substituting (6.4) and (6.21), the closed loop dynamical system may be written in compact form as:

$$\dot{\boldsymbol{\xi}} = h(t, \boldsymbol{\xi}) \triangleq \operatorname{diag}(\rho(t))^{-1} \left(-k \boldsymbol{L} \hat{\boldsymbol{L}}^{+} \boldsymbol{E} - \operatorname{diag}(\dot{\boldsymbol{\rho}}(t)) \boldsymbol{\xi} \right) \tag{6.26}$$

where $\boldsymbol{\rho}(t) \triangleq [\rho_1^u(t), \rho_1^v(t), \ldots, \rho_n^u(t), \rho_n^v(t)]^T$. Let us also define the open set $\Omega_\xi \triangleq \left(-\underline{M}_1^u, \bar{M}_1^u\right) \times \left(-\underline{M}_1^v, \bar{M}_1^v\right) \times \cdots \times \left(-\underline{M}_n^u, \bar{M}_n^u\right) \times \left(-\underline{M}_n^v, \bar{M}_n^v\right)$. In what follows, we proceed in two phases. First, the existence of a maximal solution $\boldsymbol{\xi}(t)$ of (6.26) over the set Ω_ξ for a time interval $[0,\ \tau_{max})$ is ensured, i.e., $\boldsymbol{\xi}(t) \in \Omega_\xi,\ \forall t \in [0,\ \tau_{max})$. Then, we prove that the proposed control scheme (6.21) guarantees, for all $t \in [0,\ \tau_{max})$: a) the boundedness of all closed loop signals as well as that b) $\boldsymbol{\xi}(t)$ remains strictly within a compact subset of Ω_ξ, which leads by contradiction to $\tau_{max} = \infty$ and consequently to the satisfaction of (6.9)-(6.10), thus completing the proof.

Phase A. The set Ω_ξ is nonempty and open. Moreover, (6.15)-(6.16) lead to $-\underline{M}_i^u < \xi_i^u(0) < \bar{M}_i^u$ and $-\underline{M}_i^v < \xi_i^v(0) < \bar{M}_i^v,\ i = 1, \ldots, n$. Thus, we conclude that $\boldsymbol{\xi}(0) \in \Omega_\xi$. Additionally, $h(t, \xi)$, as defined in (6.26), is continuous on t and locally Lipschitz on ξ over the set Ω_ξ. Therefore, the hypotheses of Theorem-11.2 (see Appendix-11.9) hold and the existence of a maximal solution $\boldsymbol{\xi}(t)$ of (6.26) on a time interval $[0,\ \tau_{max})$ such that $\boldsymbol{\xi}(t) \in \Omega_\xi,\ \forall t \in [0,\ \tau_{max})$ is ensured.

Phase B. We have proven in Phase A that $\xi(t) \in \Omega_\xi,\ \forall t \in [0,\ \tau_{max})$ and more specifically that $\boldsymbol{\xi}_i^u(t) \in (-\underline{M}_i^u, \bar{M}_i^u)$ and $\xi_i^v(t) \in (-\underline{M}_i^v, \bar{M}_i^v),\ i = 1, \ldots, n$ for all $t \in [0,\ \tau_{max})$. Thus, the transformed errors E_i^u, E_i^v, $i = 1, \ldots, n$, as designated in (6.22), are well defined for all $t \in [0,\ \tau_{max})$. Hence, we may adopt, based on the transformed errors (6.22), the following task function [239]:

$$\boldsymbol{\varepsilon} = \hat{\boldsymbol{L}}^{+} \boldsymbol{E}.$$

Contrary to [239], where $\hat{L}^+$ is assumed constant, in this chapter we consider a more generic case where $\hat{L}^+$ is state dependent, with positive and fixed focal length and depth estimates. Thus, the time derivative of the task function becomes:

$$\begin{aligned}\dot{\boldsymbol{\varepsilon}} &= \frac{d\hat{L}^+}{dt}E + \hat{L}^+\dot{E} \\ &= \frac{d\hat{L}^+}{dt}E + \hat{L}^+\frac{\partial E}{\partial \xi}\dot{\xi} \\ &= \frac{d\hat{L}^+}{dt}E + \hat{L}^+\frac{\partial E}{\partial \xi}\left(\frac{\partial \xi}{\partial e}\dot{e} + \frac{\partial \xi}{\partial t}\right) \\ &= \frac{d\hat{L}^+}{dt}E + \hat{L}^+\frac{\partial E}{\partial \xi}\left(\frac{\partial \xi}{\partial e}LV + \frac{\partial \xi}{\partial t}\right)\end{aligned} \tag{6.27}$$

Following [231], we also obtain:

$$\frac{d\hat{\boldsymbol{L}}^+}{dt}\boldsymbol{E} = \boldsymbol{O}(\boldsymbol{e},t)\boldsymbol{v}$$

where $\boldsymbol{O}(\boldsymbol{e},t)$ is a 6×6 matrix satisfying $\boldsymbol{O}(\boldsymbol{e},t)|_{e=0} = \boldsymbol{0}_{6\times 6}$, $\forall t \geq 0$. Hence, (6.27) becomes:

$$\dot{\boldsymbol{\varepsilon}} = \Big(\boldsymbol{O}(\boldsymbol{e},t) + \hat{\boldsymbol{L}}^+\left(\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{\xi}}\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{e}}\right)\boldsymbol{L}\Big)\boldsymbol{v} + \hat{\boldsymbol{L}}^+\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{\xi}}\frac{\partial \boldsymbol{\xi}}{\partial t}$$

and substituting the control law:

$$\boldsymbol{v} = -k\hat{\boldsymbol{L}}^+\boldsymbol{E} \triangleq -k\boldsymbol{\varepsilon}$$

we get:

$$\dot{\boldsymbol{\varepsilon}} = -k\Big(\boldsymbol{O}(\boldsymbol{e},t) + \hat{\boldsymbol{L}}^+\left(\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{\xi}}\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{e}}\right)\boldsymbol{L}\Big)\boldsymbol{\varepsilon} + \hat{\boldsymbol{L}}^+\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{\xi}}\frac{\partial \boldsymbol{\xi}}{\partial t}. \tag{6.28}$$

Finally, linearizing (6.28) for $\varepsilon = 0$, we obtain similarly to [231]:

$$\dot{\boldsymbol{\varepsilon}} = -(k\boldsymbol{A}(t) - \boldsymbol{B}(t))\boldsymbol{\varepsilon} + \boldsymbol{C}(t),$$

where:

$$\boldsymbol{A}(t) = \hat{\boldsymbol{L}}^+\left(\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{\xi}}\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{e}}\right)\boldsymbol{L}\Big|_{\varepsilon=0} \tag{6.29}$$

$$\boldsymbol{B}(t) = \frac{\partial}{\partial \varepsilon}\left(\hat{\boldsymbol{L}}^+\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{\xi}}\frac{\partial \boldsymbol{\xi}}{\partial t}\right)\Big|_{\varepsilon=0} \tag{6.30}$$

$$\boldsymbol{C}(t) = \hat{\boldsymbol{L}}^+\frac{\partial \boldsymbol{E}}{\partial \boldsymbol{\xi}}\frac{\partial \boldsymbol{\xi}}{\partial t}\Big|_{\varepsilon=0}. \tag{6.31}$$

Notice also that by construction $\frac{\partial E}{\partial \xi}\frac{\partial \xi}{\partial e}$ is a diagonal positive definite matrix. Thus, following similar arguments with [231], we conclude that $A(t) = \hat{L}^+\left(\frac{\partial E}{\partial \xi}\frac{\partial \xi}{\partial e}\right)L\Big|_{\varepsilon=0}$ is Hurwitz close to the origin for any positive and fixed focal length and depth estimates. Moreover,

it can be easily verified that the matrix $\boldsymbol{B}(t)$ and the vector $\boldsymbol{C}(t)$ are bounded for all $t \geq 0$ and vanish as time proceeds owing to the decreasing property of the performance functions. Hence, invoking Lemma 4.5 (pp.193) in [240], we conclude that $\boldsymbol{\varepsilon}(t)$ remains ultimately bounded for a sufficiently high gain value k and all $t \in [0,\ \tau_{max})$ within a neighborhood of $\varepsilon = 0$, i.e., $||\boldsymbol{\varepsilon}\,(t)\,|| \leq \bar{\boldsymbol{\varepsilon}}$. Moreover, in a neighborhood of $\varepsilon = 0$, we have $||\varepsilon|| \triangleq ||\hat{\boldsymbol{L}}^{+}\boldsymbol{E}|| \neq 0$ if $\boldsymbol{e} \neq 0$ or equivalently if $\boldsymbol{E} \neq 0$, since $\hat{\boldsymbol{L}}^{+}$ is full rank [232]. Hence, there exists $\bar{E} > 0$ such that :

$$||E\,(t)\,|| \leq \bar{E},\ \forall t \in [0,\ \tau_{max}). \tag{6.32}$$

In this way, taking the inverse logarithmic function in (6.22), we get:

$$\begin{array}{l} -\underline{M}_i^u < \underline{\xi}_i^u < \xi_i^u\,(t) < \bar{\xi}_i^u < \bar{M}_i^u \\ -\underline{M}_i^v < \underline{\xi}_i^v < \xi_i^v\,(t) < \bar{\xi}_i^v < \bar{M}_i^v \end{array},\ i = 1, \ldots, n \tag{6.33}$$

for all $t \in [0,\ \tau_{max})$, where:

$$\begin{array}{ll} \underline{\xi}_i^u = -\underline{M}_i^u \frac{\exp(\bar{E})-1}{\exp(\bar{E})+\frac{\underline{M}_i^u}{\bar{M}_i^u}}, & \bar{\xi}_i^v = \bar{M}_i^u \frac{\exp(\bar{E})-1}{\exp(\bar{E})+\frac{\bar{M}_i^u}{\underline{M}_i^u}} \\ \underline{\xi}_i^v = -\underline{M}_i^v \frac{\exp(\bar{E})-1}{\exp(\bar{E})+\frac{\underline{M}_i^v}{\bar{M}_i^v}}, & \bar{\xi}_i^v = \bar{M}_i^v \frac{\exp(\bar{E})-1}{\exp(\bar{E})+\frac{\bar{M}_i^v}{\underline{M}_i^v}} \end{array}$$

Finally, it can be easily proven from (6.22) that the control input (6.21) remains also bounded for all $t \in [0,\ \tau_{max})$.

Up to this point, what remains to be shown is that τ_{max} can be extended to ∞. Notice by (6.33) that $\boldsymbol{\xi}(t) \in \Omega'_{\xi},\ \forall t \in [0, \tau_{max})$, where the set:

$$\Omega'_{\xi} = \left[\underline{\xi}_1^u, \bar{\xi}_1^u\right] \times \left[\underline{\xi}_1^v, \bar{\xi}_1^v\right] \times \cdots \times \left[\underline{\xi}_n^u, \bar{\xi}_n^u\right] \times \left[\underline{\xi}_n^v, \bar{\xi}_n^v\right]$$

is a nonempty and compact subset of Ω_{ξ}. Hence, assuming $\tau_{max} < \infty$ and since $\Omega'_{\xi} \subset \Omega_{\xi}$, Proposition 11.1 (see Appendix-11.9) dictates the existence of a time instant $t' \in [0, \tau_{max})$ such that $\boldsymbol{\xi}(t') \notin \Omega'_{\xi}$, which is a clear contradiction. Therefore, $\tau_{max} = \infty$. As a result, all closed loop signals remain bounded and moreover $\boldsymbol{\xi}(t) \in \Omega'_{\xi} \subset \Omega_{\xi},\ \forall t \geq 0$. Finally, from (6.33) we conclude the satisfaction of (6.9)-(6.10) for all $t \geq 0$ and consequently prescribed transient and steady state performance without violating the field of view constraints, which completes the proof. □

Remark 6.5. From the aforementioned proof, it can be deduced that the proposed image based visual servoing scheme achieves its goals (i.e., prescribed performance and field of view constraints) without residing on the need of rendering $\bar{E}$ arbitrarily small (see (6.32)), by adopting an extreme value for the control gain k. More specifically, notice

that (6.33) and consequently (6.9)-(6.10), which encapsulate the prescribed performance notion and the field of view constraints, hold no matter how large the finite bound $\bar{E}$ is. Thus, contrary to what is the common practice in the related literature (i.e., the control gains are tuned towards satisfying a desired performance, nonetheless without any a priori guarantees), the actual performance of the proposed IBVS scheme is solely determined by the performance functions $\underline{M}_i^u \underline{\rho}_i^u(t)$, $\bar{M}_i^u \bar{\rho}_i^u(t)$, $\underline{M}_i^v \underline{\rho}_i^v(t)$, $\bar{M}_i^v \bar{\rho}_i^v(t)$, $i = 1, \ldots, n$. Hence, the selection of the control gain k is significantly simplified to adopting those values that lead to reasonable control effort.

Remark 6.6. Contrary to the existing works in the related literature, where the depth and camera calibration errors influence severely the performance of the visual servoing, in the proposed controller the achieved performance is a priori determined by the selection of $\underline{M}_i^u \underline{\rho}_i^u(t)$, $\bar{M}_i^u \bar{\rho}_i^u(t)$, $\underline{M}_i^v \underline{\rho}_i^v(t)$, $\bar{M}_i^v \bar{\rho}_i^v(t)$, $i = 1, \ldots, n$. However, it should be stressed that the aforementioned errors affect the region of attraction of the closed loop system around the origin, thus leading to local stability results. Studying the effect of camera calibration and depth distribution errors to obtain the magnitude of the robust initialization domain goes beyond the scope of this chapter and is left open for future research.

6.4 Simulation Study

To validate the theoretical findings and verify the efficiency of the proposed IBVS scheme, a comparative simulation study with a conventional IBVS controller [69] was conducted using the dynamic simulation environment built in MATLAB® presented in Section 2.4, where we assumed that the supposed camera system is located at the end effector of the UVMS (see Fig. 6.3). Therefore, the control input calculated by (6.21) is expressed in the joint space using the (6.23) (see Remark-6.4). a simple model based PD controller is designed and employed in dynamic level in order to track that control inputs at joint space level.

In the following simulations, the sampling period was equal to 0.033 sec, representing in this way a real time operation with a pinhole camera of 30 FPS. The target comprised of four feature points on a vertical plane, forming a square with edge 0.1 m. The desired pose of the target with respect to the camera frame O_c was $\boldsymbol{p}_{d/O_c} = [0\ 0\ 0.4\ 0\ 0\ 0]^\top$. It should be noticed that the initial pose of the target with respect to the camera frame (i.e., $\boldsymbol{p}_{init/O_c} = [1.3, 0.0, 1.37, -0.4, 0.0, -0.4]^\top$ was a rather difficult pose for conventional image based visual servoing control [220] (see Fig 6.3). The desired feature coordinates

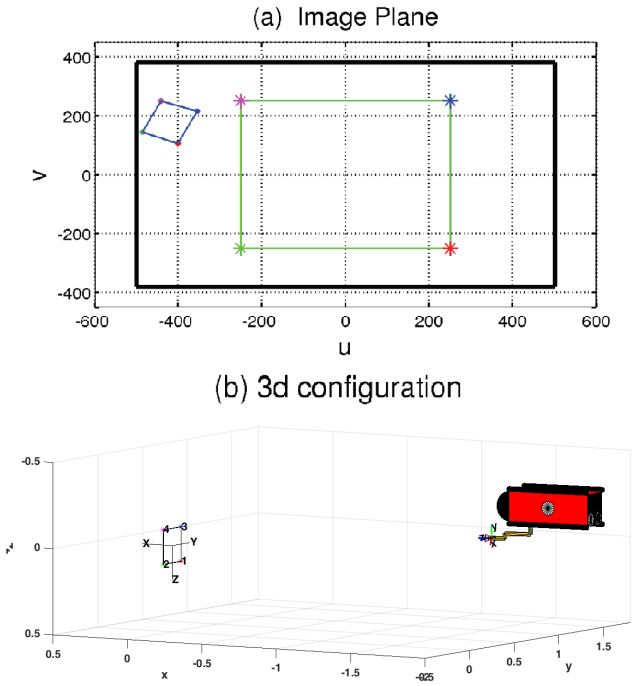


FIGURE 6.3: The simulation setup comprises a camera, attached at the end effector of a UVMS, observing a planar target. The initial and the desired position of the target on the image plane are illustrated by blue and green lines respectively. (b) The initial and the desired pose of the camera attached at the UVMS end effector.

s^* at the desired pose of the camera are:

$$s^* = \begin{bmatrix} -250 & 250 & -250 & 250 \\ -250 & -250 & 250 & 250 \end{bmatrix}.$$

The parameters $\underline{M}_i^u$, $\bar{M}_i^u$, $\underline{M}_i^v$, $\bar{M}_i^v$, $i = 1, \ldots, 4$ are chosen such that all features are retained within the camera field of view for all time. In particular, the following upper and lower bounds of the image plane:

$$u_{\min} = -500 \quad u_{\max} = 500$$
$$v_{\min} = -380 \quad v_{\max} = 380$$

were adopted in (6.13) and (6.14) to extract the values of the parameters $\underline{M}_i^u$, $\bar{M}_i^u$, $\underline{M}_i^v$, $\bar{M}_i^v$, $i = 1, \ldots, 4$ that are shown in Table 6.1. Moreover, the maximum allowable steady state error was set equal to $\rho_\infty = 5$ pixels. Thus, each feature will be ultimately confined within a square of 10 pixels edge, centered at the desired position on the image plane. Moreover, the decreasing rate l was chosen equal to $l = 0.2$ to enforce an exponential convergence dictated by $\exp(-0.2t)$. Finally, In order to test the robustness of the proposed scheme, in all subsequent simulations studies, the dynamics of the UVMS were affected by external disturbances in the form of slowly time varying sea currents

TABLE 6.1

Feature	$\underline{M}_i^u$	$\bar{M}_i^u$	$\underline{M}_i^v$	$\bar{M}_i^v$
1	250	750	130	630
2	750	250	130	630
3	250	750	630	130
4	750	250	630	130

modeled by the corresponding velocities $v_x^c = 0.1\sin(\frac{\pi}{30}t)\frac{m}{s}$ and $v_y^c = 0.1\cos(\frac{\pi}{30}t)\frac{m}{s}$.

6.4.1 Results

The robustness and guaranteed convergence properties of the proposed IBVS scheme as well as its efficiency in handling the camera field of view constraints are demonstrated via a comparative simulation study with a conventional IBVS scheme [69]. The initial feature coordinates in following simulations were:

$$s\left(0\right) = \begin{bmatrix} 327 & 406 & 282 & 363 \\ 75 & 111 & 182 & 213 \end{bmatrix}.$$

It should be noticed that the aforementioned initial configuration can be considered as rather challenging for conventional IBVS schemes, owing to the large distance between the camera and the target as well as to the rotation of the camera frame that is needed to converge to the desired configuration. Two cases with "*known*" and "*unknown*" image feature depth values were considered. For each case, the comparison was performed via two simulations. The proposed IBVS scheme was employed in the first simulation, and a conventional IBVS scheme was used in the latter. In both simulation studies, comparisons are made to show the efficacy and superior performance of the proposed scheme in handling field of view constraints versus the conventional IBVS scheme. Finally, the control gains for both cases equal and is $\boldsymbol{K} = \text{diag}\{5\}$.

Case I: "Known" depth

In this study, the depth measurement for each image feature was available in both the proposed and the conventional IBVS algorithms. The results are presented in Fig.6.4-Fig.6.7. In Fig.6.4 The evolution of UVMS/Camera system in 3D space employing the proposed IBVS scheme is illustrated. It can be seen that the UVMS/Camera system has reached to the desired position with respect to the object. In Fig.6.5 the evolution of the image feature errors employing the proposed IBVS scheme is presented. As it was expected, the feature coordinate errors in the scenario with the proposed IBVS scheme were retained within the corresponding performance envelopes and consequently the features were constantly kept within the camera field of view, as depicted in Fig.6.7(a).

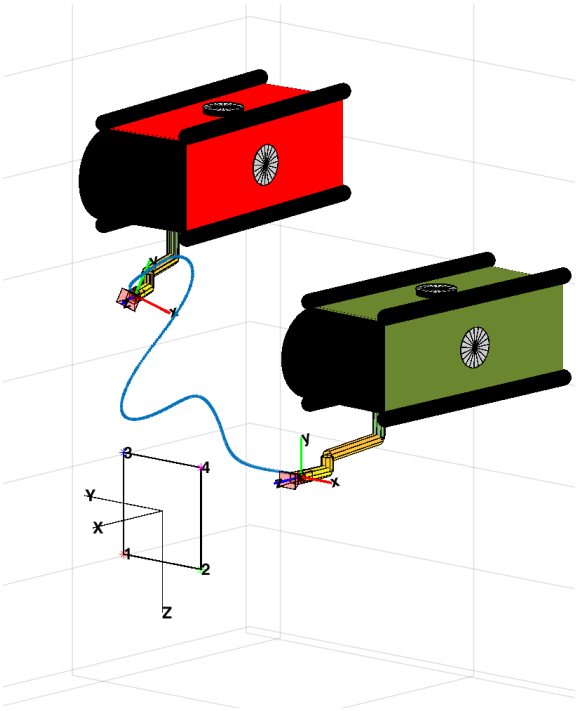


FIGURE 6.4: **Case I:** The evolution of UVMS/Camera system in 3D space employing the **proposed** IBVS scheme. The initial and final position of the UVMS/Camera system is indicated by red and green color respectively.

Apparently, the conventional IBVS scheme failed because image features left the camera FOV Fig.6.7(b). On the contrary, the proposed IBVS scheme retained all image features strictly within the camera FOV and satisfied at the same time the transient and steady state performance specifications imposed by the selected performance functions (see Fig.6.5). Finally, the required control input signals employing the proposed IBVS scheme are illustrated in Fig.6.6.

Case II: "Unknown" depth

In this case study, the same challenging initial configuration was considered and the depth measurements for each image feature were replaced in the control algorithms by the corresponding desired values $z_i^* = 0.4,\ i = 1, \ldots, 4$. The results are illustrated in Fig.6.9-Fig.6.11. In Fig.6.8 The evolution of UVMS/Camera system in 3D space employing the proposed IBVS scheme is illustrated. It can be seen that the UVMS/Camera system has reached to the desired position with respect to the object. In Fig.6.9 the evolution of the image feature errors employing the proposed IBVS scheme are presented respectively. As it was expected from the theoretical findings of this chapter, even in the case of "unknown" depth measurements, the feature coordinate errors were retained

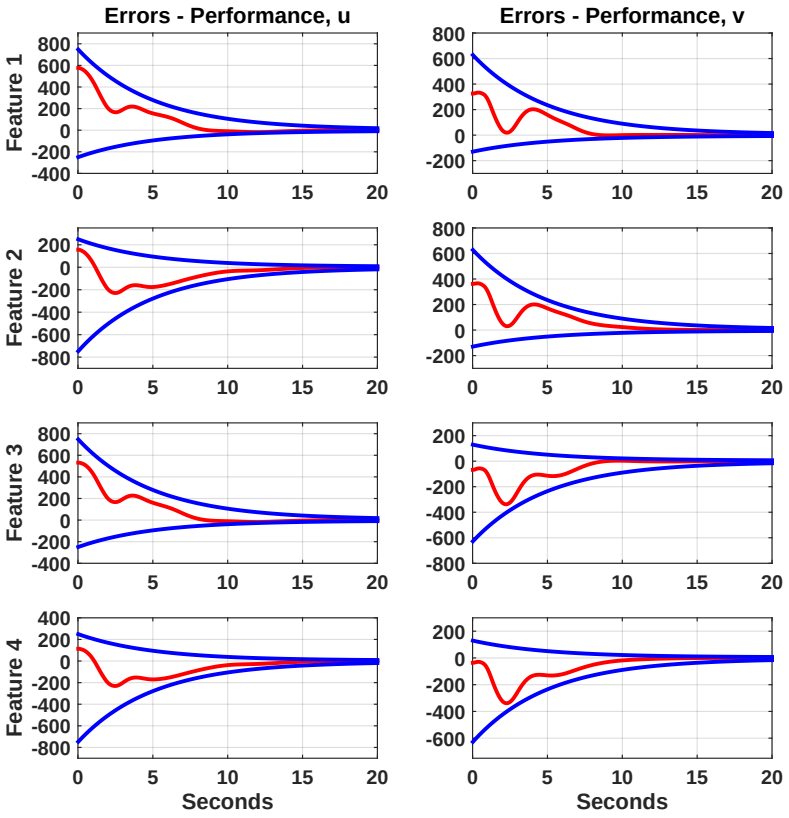


FIGURE 6.5: **Case I:** The evolution of the feature coordinate errors along with the corresponding imposed performance bounds employing the **proposed** IBVS scheme.

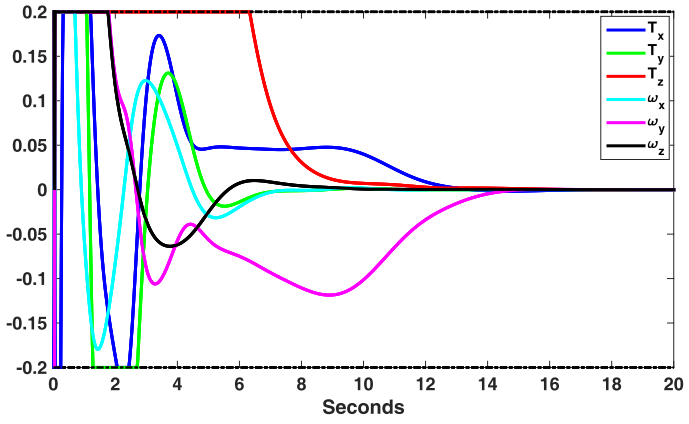
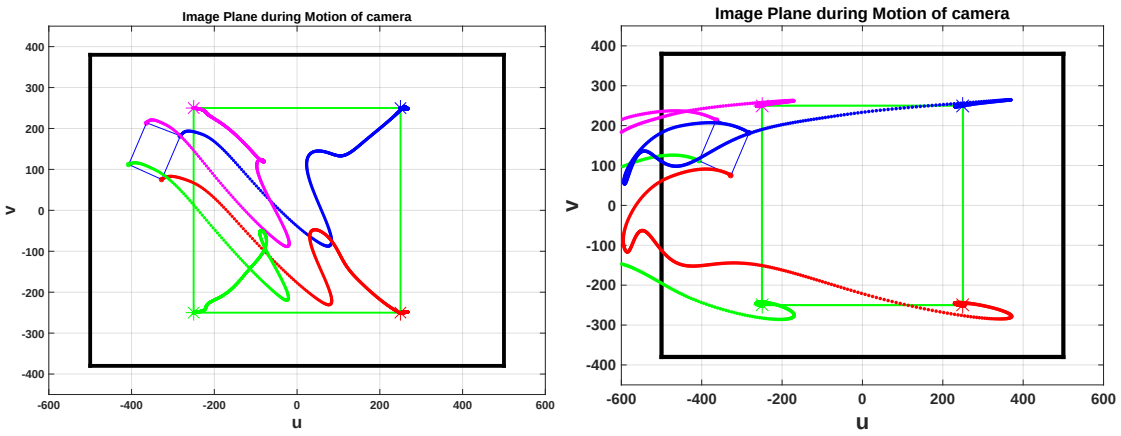


FIGURE 6.6: **Case I:** The control input signals employing the **proposed** IBVS scheme.

in the corresponding performance envelopes and consequently the features were constrained within the camera field of view (see Fig.6.11(a)). Moreover, it can be seen from Fig.6.9 that the convergence properties of the proposed IBVS scheme remained unaltered. On the contrary, noticing the evolution of the image features in Fig.6.11(b) for the case of the conventional IBVS, we conclude that the particular simulation failed as the features escaped the image boundaries. The required control input signals employing the proposed IBVS scheme are depicted in Fig.6.10. Finally, it should be noted that the



( ) a ( ) b

FIGURE 6.7: **Case I:** The evolution of the features on the image plane for (a) the **proposed** IBVS scheme and (b) the **conventional** IBVS scheme. The image boundaries are illustrated by black lines. The conventional IBVS scheme failed because image features left the camera filed of view.

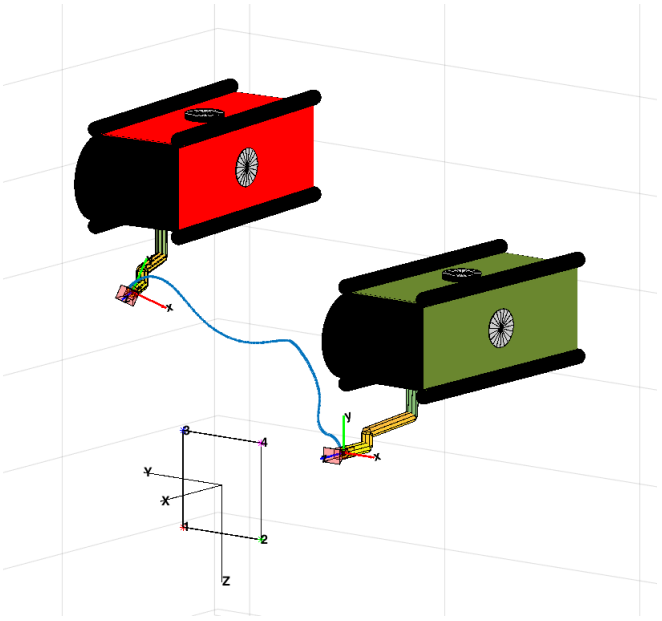


FIGURE 6.8: **Case II:** The evolution of UVMS/Camera system in 3D space employing the **proposed** IBVS scheme. The initial and final position of the UVMS/Camera system is indicated by red and green color respectively.

control gains were kept unaltered in both Cases I and II, thus revealing that the control gain selection in the proposed IBVS scheme has been significantly simplified since it is decoupled by both the closed loop transient and steady state response as well as the satisfaction of the camera field of view constraints.

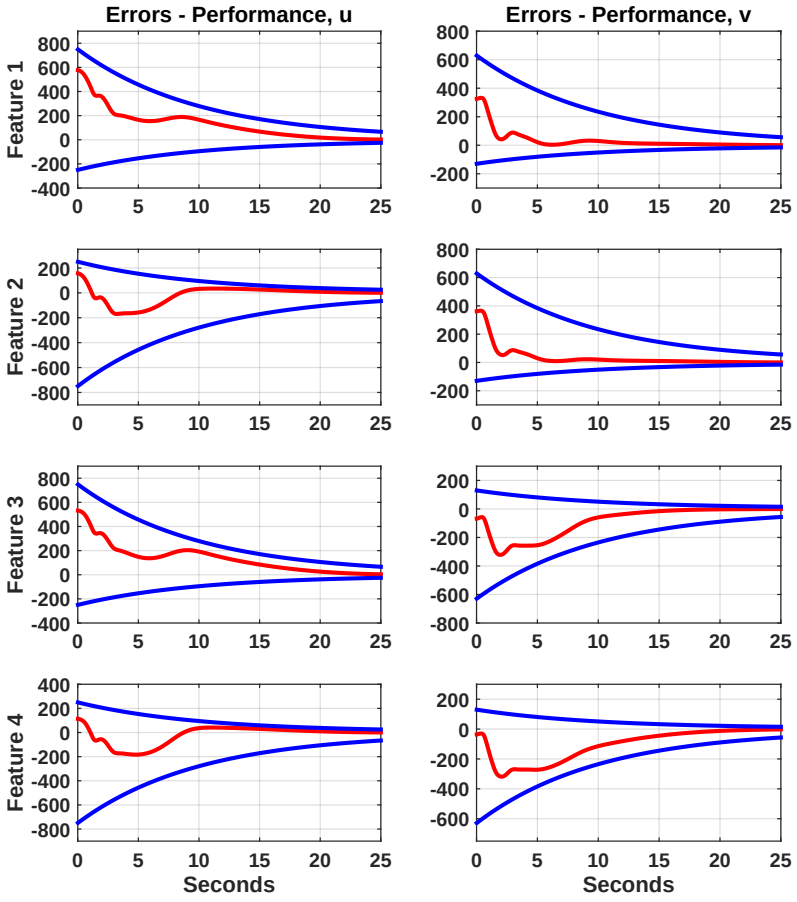


FIGURE 6.9: **Case II:** The evolution of the feature coordinate errors along with the corresponding imposed performance bounds employing the **proposed** IBVS scheme.

6.4.2 Video

The proposed IBVS control scheme was experimentally tested using a redundant manipulator system equipped with a USB camera, validating the theoretical findings. That experimental study can be found at the following url: https://youtu.be/A1Lqv8xeCRA. Alternatively, it can be found in the attached dvd as file: video_ch6.mpg

6.5 Conclusions

In the chapter we presented a novel robust image based visual servoing scheme that achieves prescribed transient and steady state performance on the image feature coordinate errors and satisfies the camera field of view constraints. The proposed controller exhibits the following important characteristics. It is of low complexity and thus it can be easily integrated on embedded control systems. Additionally, the priori guaranteed

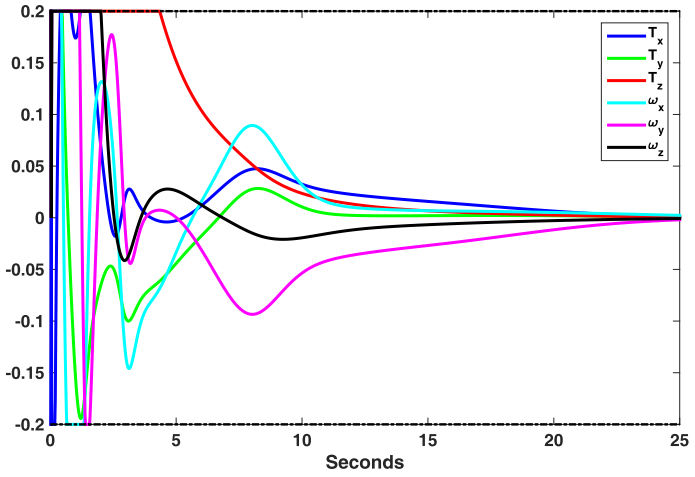
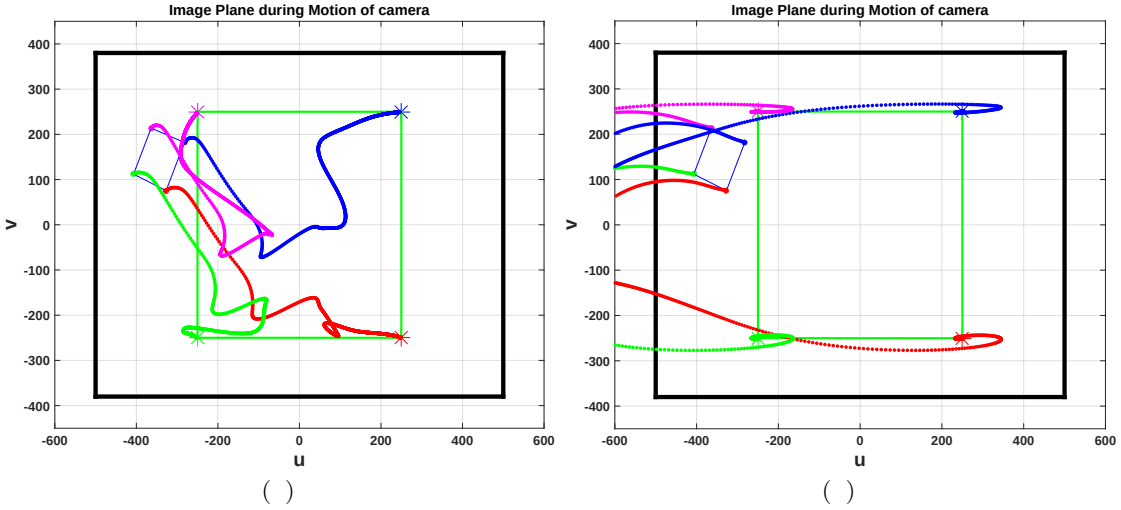


FIGURE 6.10: **Case II:** The control input signals employing the **proposed** IBVS scheme.



( ) ( )

FIGURE 6.11: **Case II:** The evolution of the features on the image plane for (a) the **proposed** IBVS scheme and (b) the **conventional** IBVS scheme. The image boundaries are illustrated by black lines. The conventional IBVS scheme failed because image features left the camera filed of view.

performance that is imposed by certain designer-specified performance functions, simplifies significantly the selection of the controller parameters. Gain tuning is only confined to achieving reasonable control effort. Finally, the depth and camera calibration errors influence minimally the achieved performance of the visual servoing since they affect only the region of attraction of the closed loop system.

Part IV - Interaction & Cooperative Manipulation

Chapter 7

A Robust Interaction Control Approach for Underwater Vehicle Manipulator Systems

7.1 Introduction

When a UVMS interacts with the environment via its end-effector (See Fig.7.1) , the redundancy resolution formalism becomes a tricky problem compared to the unconstrained motion tasks. In this case, the primary task is defined by the velocity tracking along the tangent space and force tracking along the normal space [143]. It should be noticed that only a small number of publications and research studies have addressed the problems of force-motion control and redundancy resolution within the same control framework. In particular, the force control scheme proposed in [241], handles the force tracking task as the secondary objective while the motion of the end effector is defined as the primary objective task. Moreover, the aforementioned method needs reseting when the contact is lost. In a recent work [143], a model free control scheme was proposed for a UVMS in rigid contact with the environment while simultaneously dealing with the aforementioned issues. More specifically, by means of a priority based inverse kinematic formulation proposed in [144], the UVMS operational limitations along with the contact maintenance task are formulated as inequality constraints within a cascade of quadratic programming problems. However, beyond the known implementation drawbacks of sliding mode control technique (chattering and high gain excitation), it requires the accurate knowledge of the bounds of the fluid disturbances. Moreover, the transient and steady state performance of the system is not considered at all. In this chapter, we propose a force-motion control strategy for a UVMS in compliant contact with a planar surface.

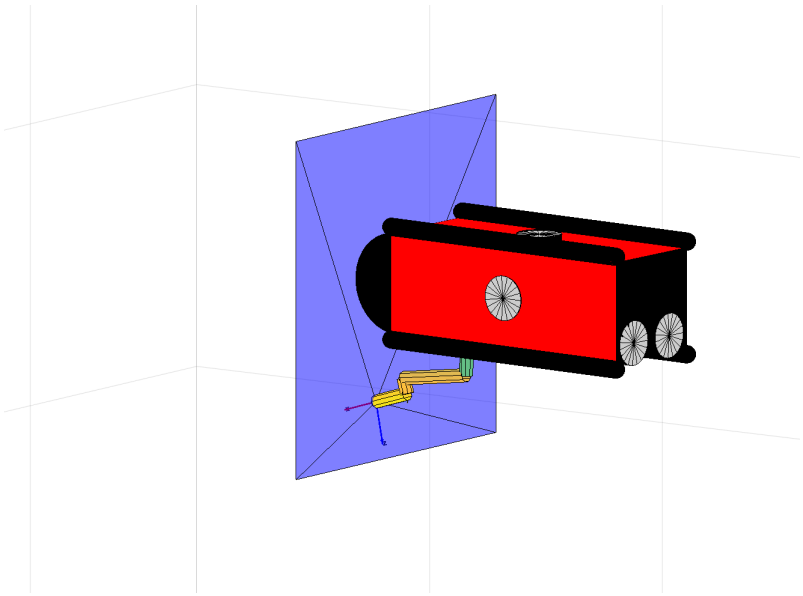


FIGURE 7.1: Workspace including the UVMS in compliant contact with a planar surface.

The purpose of the controller is to simultaneously track a desired trajectory along the planar surface and a force along the normal direction. The proposed method does not require any knowledge of either the UVMS dynamic parameters, or the stiffness model, or the disturbance profile. Furthermore, various performance issues such as: i) maintaining the contact, ii) tracking the desired trajectory, iii) inferior overshooting of the interaction force error and iv) robust steady state response are achieved. Moreover, the novel formulation of the problem, allows the tracking of the desired force and position trajectories to be considered equally prioritized, while it enables us to treat various other operational limitations (e.g., joint limits) as secondary tasks, and thus fully decouple them. In particular, the proposed control strategy tackles all aforementioned challenges and further guarantees predefined behavior in terms of overshoot, convergence rate and maximum steady state error. Furthermore, the robustness of the proposed control strategy with respect to external disturbances is enhanced. Finally, the complexity of the proposed control law is significantly low. It is actually a static scheme involving only a few calculations to derive the control signal, which enables its onboard implementation straightforwardly.

The rest of the chapter is organized as follows: In Section-7.2, the problem is rigorously formulated. The analytic description of the proposed method along with the corresponding stability proof are presented in Section-7.3. The efficiency of the proposed approach is illustrated and validated via simulated and experimental results in Section-7.4 . Finally, Section-7.6 concludes the chapter.

7.2 Problem Formulation

Consider an n degrees of freedom UVMS in compliant contact with a planar surface. Let $\boldsymbol{q} = [\boldsymbol{q}_a^\top,\ \boldsymbol{q}_m^\top]^\top \in \mathbb{R}^n$ be the state variables of the UVMS, where $\boldsymbol{q}_a = [\boldsymbol{\eta}_1^\top, \boldsymbol{\eta_2}^\top]^\top \in \mathbb{R}^6$ involves the position vector $\boldsymbol{\eta}_1 = [x_v, y_v, z_v]^\top$ and the orientation $\boldsymbol{\eta}_2 = [\phi, \theta, \psi]^\top$ of the vehicle expressed in the Euler-angles representation with respect to (w.r.t) an inertial frame $\{I\}$ and $\boldsymbol{q}_m \in \mathbb{R}^{n-6}$ is the angle vector of the manipulator's joints. Consider also the frame $\{E\}$ attached at the end-effector of the UVMS described by a position vector $\boldsymbol{x}_e = [x_e, y_e, z_e]^\top \in \mathbb{R}^3$ and a rotation matrix $\boldsymbol{R}_e = [\boldsymbol{n}_e, \boldsymbol{o}_e, \boldsymbol{\alpha}_e]$ w.r.t the inertial frame $\{I\}$. Let also $\boldsymbol{\omega}_e$ be the rotational velocity of the end-effector that satisfies $\boldsymbol{S}(\boldsymbol{\omega}_e) = \dot{\boldsymbol{R}}_e \boldsymbol{R}_e^\top$, where

$$\boldsymbol{S}(\boldsymbol{d}) = \begin{bmatrix} 0 & -d_z & d_y \\ d_z & 0 & -d_x \\ -d_y & d_x & 0 \end{bmatrix}$$

is the skew-symmetric matrix of the vector $\boldsymbol{d} = [d_x\ d_y\ d_z]^\top$. Let also $\dot{\boldsymbol{x}} = [\dot{\boldsymbol{x}}_e^\top, \boldsymbol{\omega}_e^\top]^\top \in \mathbb{R}^6$ denote the velocity of the end-effector frame. Without loss of generality, we have [10]:

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\boldsymbol{\zeta} \tag{7.1}$$

where $\boldsymbol{\zeta} = [\boldsymbol{v}^\top, \dot{\boldsymbol{q}}_{m,i}^\top]^\top \in \mathbb{R}^n$ is the overall velocity vector that involves the body velocities of the vehicle $\boldsymbol{v}$ and the joint velocities of the manipulator $\dot{\boldsymbol{q}}_{m,i},\ i \in \{1, \ldots, n-6\}$ and $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{6\times n}$ is the geometric Jacobian Matrix [10].

We assume that the UVMS has initially established contact with a planar surface, whose normal and tangential vectors expressed w.r.t the inertial frame $\{I\}$ are known. Thus, for the sake of simplicity, we consider the inertial frame $\{I\}$ attached at some point on the surface with its x-axis normal to the surface pointing inwards (See Fig.7.2). Now, let us denote the unit vector normal to the contact surface and the generalized normal vector as $\boldsymbol{n}_s = [1\ 0\ 0]^\top \in \mathbb{R}^3$ and $\boldsymbol{n} = [\boldsymbol{n}_s^\top\ \boldsymbol{0}_3^\top]^\top \in \mathbb{R}^6$ respectively. We also assume that the end-effector is rigid, thus the contact compliance arises from the planar surface[1]. Hence, the deformation χ is given as a function of $\boldsymbol{x}_e$ as follows:

$$\chi = \boldsymbol{n}_s^\top \boldsymbol{x}_e = x_e \tag{7.2}$$

and its derivative is calculated by:

$$\dot{\chi} = \boldsymbol{n}_s^\top \dot{\boldsymbol{x}}_e = \dot{x}_e \tag{7.3}$$

[1]In case of UVMS with soft tip, the compliance may arise either from the tip side or the surface or both. Thus, the deformation can also be derived without affecting the subsequent analysis.

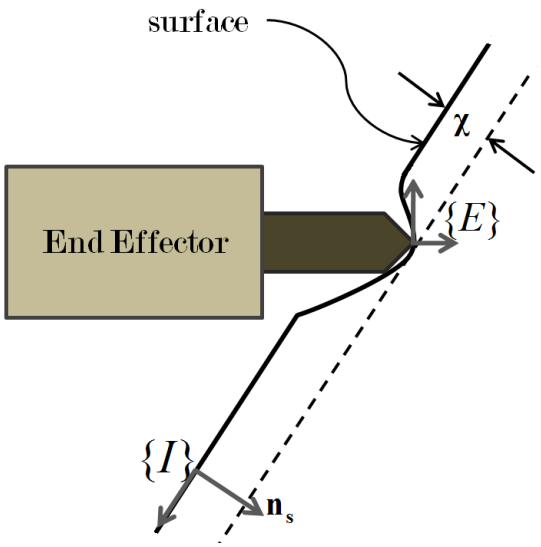


FIGURE 7.2: A graphical illustration of the UVMS end-effector in compliant contact with a planar surface.

During an intervention task, the UVMS exerts an interaction wrench $\boldsymbol{\lambda} \in \mathbb{R}^6$ at the contact, which can be measured by a force/torque sensor attached to its end-effector. This interaction wrench can be decomposed into: (i) $\boldsymbol{n}\boldsymbol{n}^\top\boldsymbol{\lambda}$ that is normal to the surface and (ii) $(\boldsymbol{I}_{6\times6} - \boldsymbol{n}\boldsymbol{n}^\top)\boldsymbol{\lambda}$ involving tangential forces and torques, owing to tangential deformations and friction terms. In this chapter, we assume that the normal force magnitude $f = \boldsymbol{n}^\top\boldsymbol{\lambda}$ is a positive and continuously differentiable nonlinear function of the material deformation χ:

$$f = \Phi(\chi), \quad \forall\chi \geq 0. \tag{7.4}$$

The aforementioned general formulation includes several force deformation models such as the Hertz model [242] ($\Phi(\chi) = k\chi^{\frac{3}{2}}, k > 0$) or the quadratic model $\Phi(\chi) = k\chi^2, k > 0$ [243]. The time derivative of the normal force magnitude in view of (7.3) is then given by:

$$\dot{f} = \partial\Phi(\chi)\dot{x}_e \tag{7.5}$$

where $\partial\Phi(\chi) = \frac{d\Phi}{d\chi}$ is strictly positive for all $\chi \geq \underline{\chi}^* > 0$, where $\underline{\chi}^*$ is any strictly real positive number. Thus, there is an unknown strictly positive constant $\partial\Phi^*$ such that:

$$\partial\Phi(\chi) \geq \partial\Phi^* > 0, \forall\chi \geq \underline{\chi}^*. \tag{7.6}$$

Without loss of generality, the dynamics of the UVMS in complaint contact with the environment can be formulated as [10]:

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{q},\boldsymbol{\zeta})\boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{q},\boldsymbol{\zeta})\boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{J}^{\top}(\boldsymbol{q})\boldsymbol{\lambda} + \boldsymbol{\delta}(\boldsymbol{q},\boldsymbol{\zeta},t) = \boldsymbol{\tau} \tag{7.7}$$

where $\boldsymbol{\delta}(\boldsymbol{q},\boldsymbol{\zeta},t)$ encapsulates bounded unmodeled terms and external disturbances (sea waves and currents). Moreover, $\boldsymbol{\tau} \in \mathbb{R}^n$ denotes the control input at the joint/thruster level, $\boldsymbol{M}(\boldsymbol{q})$ is the positive definite inertial matrix, $\boldsymbol{C}(\boldsymbol{q},\boldsymbol{\zeta})$ represents coriolis and centrifugal terms, $\boldsymbol{D}(\boldsymbol{q},\boldsymbol{\zeta})$ models dissipative effects, $\boldsymbol{g}(\boldsymbol{q})$ encapsulates the gravity and buoyancy effects and $\boldsymbol{J}^{\top}(\boldsymbol{q})\boldsymbol{\lambda}$ represents the effect of the external forces/torques applied at the end-effector owing to the contact.

Finally, the problem to be solved in this chapter is formulated as follows:

Problem 7.1. *Given a UVMS, a desired force profile along the normal to the surface direction $f^d(t)$, a desired position trajectory $[y_e^d(t),\ z_e^d(t)]^{\top} \in \mathbb{R}^2$ on the planar surface, as well as a smooth rotation matrix target $\boldsymbol{R}^d(t) = [\boldsymbol{n}^d(t), \boldsymbol{o}^d(t), \boldsymbol{\alpha}^d(t)]$, design a feedback control law, without incorporating any information regarding either the UVMS dynamics or the force deformation model, such that the following are satisfied:*

1. *Predefined behavior in terms of overshoot, convergence rate and maximum steady state error on the force/position/orientation tracking errors;*
2. *Contact maintenance;*
3. *Bounded closed loop signals;*
4. *Robustness against external disturbances.*

7.3 Control Methodology

Typically, in a force/position control problem the robot end-effector should track a force trajectory along the normal to the surface direction, a desired position trajectory on the surface and possibly attain a desired orientation related to the contact surface. Hence,

we define the force, position, orientation and the overall errors as:

$$e_f = f - f^d \in \mathbb{R}, \tag{7.8}$$

$$\boldsymbol{e}_p \triangleq \begin{bmatrix} e_y \\ e_z \end{bmatrix} = \begin{bmatrix} y_e - y_e^d \\ z_e - z_e^d \end{bmatrix} \in \mathbb{R}^2, \tag{7.9}$$

$$\boldsymbol{e}_o \triangleq \begin{bmatrix} e_{o_1} \\ e_{o_2} \\ e_{o_3} \end{bmatrix} = \frac{1}{2}\Big(\boldsymbol{n}_e \times \boldsymbol{n}^d + \boldsymbol{o}_e \times \boldsymbol{o}^d + \boldsymbol{\alpha}_e \times \boldsymbol{\alpha}^d\Big) \in \mathbb{R}^3, \tag{7.10}$$

$$\boldsymbol{e} \triangleq [e_f, e_y, e_z, e_{o_1}, e_{o_2}, e_{o_3}]^\top \in \mathbb{R}^6 \tag{7.11}$$

Notice that for the orientation error $\boldsymbol{e}_o$, we have employed the outer product formulation [244, 245] of the end-effector rotation matrix $\boldsymbol{R}_e$ and the desired rotation matrix $\boldsymbol{R}^d$ to relax the representation singularity issue that is inherent in case Euler angles are adopted [2]. Differentiating (7.8)-(7.10) with respect to time and in view of (7.5), we obtain:

$$\dot{e}_f = \partial f(\chi)\dot{x}_e - \dot{f}^d, \tag{7.12}$$

$$\dot{\boldsymbol{e}}_p = \begin{bmatrix} \dot{e}_y \\ \dot{e}_z \end{bmatrix} = \begin{bmatrix} \dot{y}_e - \dot{y}_e^d \\ \dot{z}_e - \dot{z}_e^d \end{bmatrix}, \tag{7.13}$$

$$\dot{\boldsymbol{e}}_o = \begin{bmatrix} \dot{e}_{o_1} \\ \dot{e}_{o_2} \\ \dot{e}_{o_3} \end{bmatrix} = \boldsymbol{L}\boldsymbol{\omega}_e - \boldsymbol{L}\boldsymbol{\omega}^d, \tag{7.14}$$

where $\boldsymbol{L}$ is defined as:

$$\boldsymbol{L} = \frac{1}{2}\Big[\boldsymbol{S}(\boldsymbol{n}_e)\boldsymbol{S}(\boldsymbol{n}^d) + \boldsymbol{S}(\boldsymbol{o}_e)\boldsymbol{S}(\boldsymbol{o}^d) + \boldsymbol{S}(\boldsymbol{\alpha}_e)\boldsymbol{S}(\boldsymbol{\alpha}^d)\Big] \tag{7.15}$$

which is full rank when the relative orientation between the frames $\boldsymbol{R}_e$ and $\boldsymbol{R}^d$ is confined less than 90^o for an angle-axis local parametrization and hence is not restrictive for practical cases [245].

7.3.1 Control Design

An appropriate methodology to meet the control objectives of this chapter is the prescribed performance control technique [246–248], which is adapted here in order to achieve predefined transient and steady state response bounds for the errors. By prescribed performance control we mean that the force, position and orientation errors evolve strictly within a predefined region that is bounded by decaying functions of time,

[2] Other representations, such as unit quaternions, can also be considered in order to describe the orientation errors. For more details the reader is referred to [244].

which is mathematically expressed as:

$$-\underline{M}_i\rho_i(t) < e_i(t) < \overline{M}_i\rho_i(t),\ i \in \{f, y, z, o_1, o_2, o_3\},\ \forall t \geq 0 \tag{7.16}$$

with

$$\rho_i(t) = \left(1 - \frac{\rho_i^{\infty}}{\max\{\underline{M}_i, \overline{M}_i\}}\right)\exp(-l_i t) + \left(\frac{\rho_i^{\infty}}{\max\{\underline{M}_i, \overline{M}_i\}}\right), i \in \{f, y, z, o_1, o_2, o_3\}. \tag{7.17}$$

The constants $\underline{M}_i$, $\overline{M}_i$, are selected such that (7.16) is satisfied at $t = 0$ (i.e., $-\underline{M}_i < e_i(0) < \overline{M}_i$). The constant $\rho_i^{\infty} = \lim_{t\to\infty}\rho_i(t)$ represents the maximum allowable size of $e_i(t)$ at the steady state, which can be set arbitrarily small to a value reflecting the resolution of the measurement device, thus achieving practical convergence of $e_i(t)$ to zero. Furthermore, the decreasing rate of $\rho_i(t)$, which is affected by the constant l_i introduces a lower bound of the required speed of convergence of $e_i(t)$, while the maximum overshoot is prescribed less than $\underline{M}_i\rho_i(0)$ or $\overline{M}_i\rho_i(0)$. Thus, the appropriate selection of the performance functions $\rho_i(t)$, $i \in \{f, y, z, o_1, o_2, o_3\}$ as well as of the design constants $\underline{M}_i$, $\overline{M}_i$, $i \in \{f, y, z, o_1, o_2, o_3\}$ encapsulates performance characteristics for the corresponding tracking errors e_i, $i \in \{f, y, z, o_1, o_2, o_3\}$.

In particular, meeting the performance bounds for the force error, as described in (7.16), allows us further to guarantee a priori that contact with the surface is never lost (i.e., $f(t) \geq \underline{f}^* > 0,\ \forall t \geq 0$ for a positive constant $\underline{f}^*$) and excessive interaction forces are avoided (i.e, $f(t) \leq \overline{f}^* > 0,\ \forall t \geq 0$ for a positive constant $\overline{f}^* > \underline{f}^* > 0$). In this spirit, $\underline{M}_f$, $\overline{M}_f$ and $\rho_f(t)$ are selected to further satisfy: $\inf_{t\geq 0}\{-\underline{M}_f\rho_f(t) + f^d(t)\} > \underline{f}^*$, $\sup_{t\geq 0}\{\overline{M}_f\rho_f(t) + f^d(t)\} < \overline{f}^*$. Hence, owing to (7.17) it can be verified that the satisfaction of the performance bounds for the force error guarantees further that $0 < \inf_{t\geq 0}\{-\underline{M}_f\rho_f(t) + f^d(t)\} \leq f(t) \leq \sup_{t\geq 0}\{\overline{M}_f\rho_f(t) + f^d(t)\},\ \forall t \geq 0$. Finally, in view of (7.6), there exist constants $\underline{\partial f}$, $\overline{\partial f}$ such that $0 < \underline{\partial f} \leq \partial f(\chi) \leq \overline{\partial f}$.

In the sequel, we propose a state feedback control protocol that does not incorporate any information regarding the UVMS dynamic model (7.7) or the deformation model and achieves force/position/orientation tracking of the corresponding smooth and bounded desired trajectories with prescribed transient and steady state response. The overall control architecture is illustrated in Fig.7.3.

Level I-a: Select the performance functions $\rho_i(t)$ and the corresponding positive parameters $\underline{M}_i$, $\overline{M}_i$ such that $-\underline{M}_i < e_i(0) < \overline{M}_i$, $\forall i \in \{f, y, z, o_1, o_2, o_3\}$ and the desired transient and steady state performance specifications are properly encapsulated.

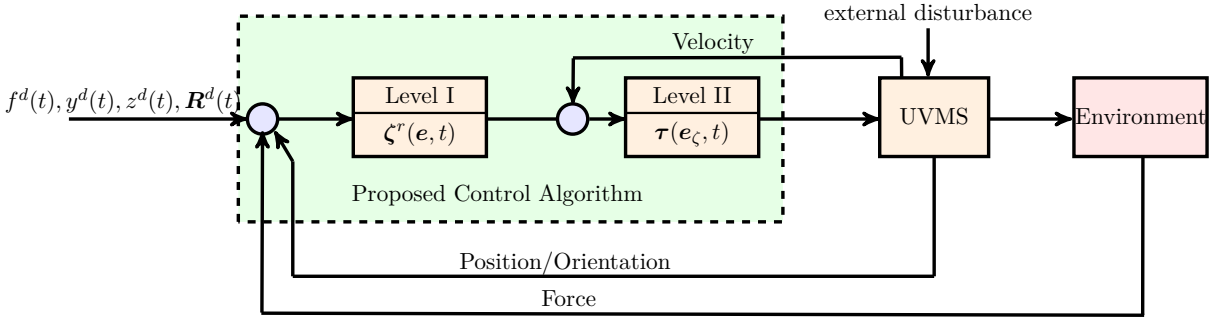


FIGURE 7.3: The closed loop block diagram of the proposed control scheme.

Level I-b: Define the transformed errors ε_i as:

$$\varepsilon_i(\xi_i) = \ln\left(\frac{1+\frac{\xi_i}{M_i}}{1-\frac{\xi_i}{M_i}}\right),\ i \in \{f, y, z, o_1, o_2, o_3\}, \tag{7.18}$$

where

$$\xi_i(t) = \frac{e_i(t)}{\rho_i(t)},\ i \in \{f, y, z, o_1, o_2, o_3\}, \tag{7.19}$$

denote the normalized force/position/orientation errors and select the end-effector reference velocity:

$$\dot{\boldsymbol{x}}^r(\boldsymbol{\xi}_x, t) = -\boldsymbol{K}\mathbf{P}^{-1}(t)\text{diag}[\boldsymbol{I}_{3\times 3}, \boldsymbol{L}^{-1}]\boldsymbol{R}(\boldsymbol{\xi}_x)\boldsymbol{\varepsilon}(\boldsymbol{\xi}_{\boldsymbol{x}}) \tag{7.20}$$

where

$$\boldsymbol{R}(\boldsymbol{\xi}_x) = \underset{i\in\{f,y,z,o_1,o_2,o_3\}}{\text{diag}}\left[\frac{\frac{1}{M_i}+\frac{1}{M_i}}{\left(1-\frac{\xi_i}{M_i}\right)\left(1+\frac{\xi_i}{M_i}\right)}\right], \tag{7.21}$$

$$\boldsymbol{\varepsilon}(\boldsymbol{\xi}_{\boldsymbol{x}}) \triangleq [\varepsilon_f(\xi_f), \varepsilon_y(\xi_y), \varepsilon_z(\xi_z), \varepsilon_{o_1}(\xi_{o_1}), \varepsilon_{o_2}(\xi_{o_2}), \varepsilon_{o_3}(\xi_{o_3})]^\top, \tag{7.22}$$

with $\boldsymbol{\xi}_x = [\xi_f, \xi_y, \xi_z, \xi_{o_1}, \xi_{o_2}, \xi_{o_3}]^\top$, $\mathbf{P}(t) \triangleq \underset{i\in\{f,y,z,o_1,o_2,o_3\}}{\text{diag}}[\rho_i(t)]$ and $\boldsymbol{K} = \underset{i\in\{f,y,z,o_1,o_2,o_3\}}{\text{diag}}[k_i]$ is a positive diagonal gain matrix. Subsequently, the task-space desired motion profile $\dot{\boldsymbol{x}}^r$ can be expressed equivalently in the configuration space via:

$$\boldsymbol{\zeta}^r(t) = \boldsymbol{J}(\boldsymbol{q})^{\#}\dot{\boldsymbol{x}}^r + \big(\boldsymbol{I}_{n\times n} - \boldsymbol{J}(\boldsymbol{q})^{\#}\boldsymbol{J}(\boldsymbol{q})\big)\dot{\boldsymbol{x}}^0 \in \mathbb{R}^n \tag{7.23}$$

where $\boldsymbol{J}(\boldsymbol{q})^{\#}$ denotes the generalized pseudo-inverse [219] of the Jacobian $\boldsymbol{J}(\boldsymbol{q})$ and $\dot{\boldsymbol{x}}^0$ denotes secondary tasks (e.g., maintaining manipulator's joint limits, increasing manipulability) to be regulated independently since they do not contribute to the end-effector's

velocity [145] (i.e., they belong to the null space of the Jacobian $\boldsymbol{J}(\boldsymbol{q})$)[3].

Level II-a: Define the velocity error vector as:

$$\boldsymbol{e}_{\zeta}(t) \triangleq [e_{\zeta_1}(t), \ldots, e_{\zeta_n}(t)]^{\top} = \boldsymbol{\zeta}(t) - \boldsymbol{\zeta}^{r}(t) \in \mathbb{R}^n \tag{7.24}$$

and select the corresponding performance functions:

$$\rho_{\zeta_i}(t) = (\rho_{\zeta_i}^{0} - \rho_{\zeta_i}^{\infty}) \exp(-l_{\zeta_i} t) + \rho_{\zeta_i}^{\infty}, \ i = 1, \ldots, n \tag{7.25}$$

with $\rho_{\zeta_i}^{0} > |e_{\zeta_i}(0)|$, $\rho_{\zeta_i}^{\infty} > 0$ and $l_{\zeta_i} > 0, i = 1, \ldots n$. Notice that similarly to the force/-position/orientation errors we intend to enforce transient and steady state response on the velocity errors $e_{\zeta_i}(t), i = 1, \ldots, n$ as well by satisfying:

$$-\rho_{\zeta_i}(t) < e_{\zeta_i}(t) < \rho_{\zeta_i}(t), \ \forall t \geq 0 \ i = 1, \ldots, n \tag{7.26}$$

Level II-b: Define the transformed velocity error vector:

$$\boldsymbol{\varepsilon}_{\zeta}(\boldsymbol{\xi}_{\zeta}) \triangleq [\varepsilon_{\zeta_1}(\xi_{\zeta_1}), \ldots, \varepsilon_{\zeta_n}(\xi_{\zeta_n})]^{\top} = \left[\ln\left(\frac{1+\xi_{\zeta_1}}{1-\xi_{\zeta_1}}\right), \ldots, \ln\left(\frac{1+\xi_{\zeta_n}}{1-\xi_{\zeta_n}}\right) \right]^{\top} \tag{7.27}$$

where

$$\boldsymbol{\xi}_{\zeta}(t) \triangleq [\xi_{\zeta_1}, \ldots, \xi_{\zeta_n}]^{\top} = \mathbf{P}_{\zeta}^{-1}(t) \boldsymbol{e}_{\zeta}(t) \tag{7.28}$$

denotes the normalized velocity error vector, with $\mathbf{P}_{\zeta}(t) = \underset{i=1,\ldots,n}{\text{diag}} [\rho_{\zeta_i}(t)]$ and design the state feedback control law:

$$\boldsymbol{\tau}(\boldsymbol{e}_{\zeta}(t), t) = -\boldsymbol{K}_{\zeta} \mathbf{P}_{\zeta}^{-1}(t) \boldsymbol{R}_{\zeta}(\boldsymbol{\xi}_{\zeta}) \boldsymbol{\varepsilon}_{\zeta}(\boldsymbol{\xi}_{\zeta}) \tag{7.29}$$

where

$$\boldsymbol{R}_{\zeta}(\boldsymbol{\xi}_{\zeta}) = \underset{i=1,\ldots,n}{\text{diag}} \left[\frac{2}{1-\xi_{\zeta_i}^2} \right] \tag{7.30}$$

and $\boldsymbol{K}_{\zeta} > 0$ is a diagonal gain matrix.

Theorem 7.1. *The proposed state feedback control law (7.18)-(7.30) enforces tracking of: i) the desired normal force trajectory $f^d(t)$, ii) the desired position trajectory $[y_e^d(t), \ z_e^d(t)]^{\top} \in \mathbb{R}^2$ on the planar surface as well as iii) the smooth rotation matrix target $\boldsymbol{R}^d(t) = [\boldsymbol{n}^d(t), \boldsymbol{o}^d(t), \boldsymbol{\alpha}^d(t)]$ with the desired transient and steady state performance.*

[3] For more details on task priority based control and redundancy resolution for UVMSs the reader is referred to [145] and [151].

Proof. First, let us define the overall normalized error vector $\boldsymbol{\xi} = [\boldsymbol{\xi}_x^\top, \boldsymbol{\xi}_\zeta^\top]^\top$. Differentiating (7.19) and (7.28) with respect to time and substituting the system dynamics (7.7), (7.12)-(7.14), (7.20), (7.24) and (7.29), we obtain the closed loop system dynamics:

$$\begin{aligned}
\dot{\boldsymbol{\xi}}_x &\triangleq h_x(\boldsymbol{\xi}_x, t) \\
&= -\mathbf{P}^{-1}(t)\text{diag}\Big[\partial\Phi(\chi), \boldsymbol{I}_{5\times5}\Big]\boldsymbol{K}\mathbf{P}^{-1}(t)\boldsymbol{R}(\boldsymbol{\xi}_x)\boldsymbol{\varepsilon}(\boldsymbol{\xi}_x) \\
&\quad + \mathbf{P}^{-1}(t)\Bigg[\text{diag}\Big[\partial\Phi(\chi), \boldsymbol{I}_{2\times2}, \boldsymbol{L}\Big]\boldsymbol{J}\mathbf{P}_\zeta(t)\boldsymbol{\xi}_\zeta \\
&\quad - \Big[\dot{f}^d, \dot{y}^d, \dot{z}^d, [\boldsymbol{L}\boldsymbol{\omega}^d]^\top\Big]^\top - \dot{\mathbf{P}}(t)\boldsymbol{\xi}_x\Bigg]
\end{aligned} \tag{7.31}$$

$$\begin{aligned}
\dot{\boldsymbol{\xi}}_\zeta(\xi_\zeta, t) &\triangleq h_\zeta(\boldsymbol{\xi}_\zeta, t) \\
&= -\mathbf{P}_\zeta^{-1}(t)\boldsymbol{M}^{-1}\boldsymbol{K}_\zeta\mathbf{P}_\zeta^{-1}(t)\boldsymbol{R}_\zeta(\boldsymbol{\xi}_\zeta)\boldsymbol{\varepsilon}_\zeta(\boldsymbol{\xi}_\zeta) - \\
&- \mathbf{P}_\zeta^{-1}(t)\Big[\boldsymbol{M}^{-1}\Big(\boldsymbol{C}\cdot(\mathbf{P}_\zeta(t)\boldsymbol{\xi}_\zeta + \boldsymbol{\zeta}^r) + \boldsymbol{D}\cdot(\mathbf{P}_\zeta(t)\boldsymbol{\xi}_\zeta + \boldsymbol{\zeta}^r) \\
&+ \boldsymbol{g} + \boldsymbol{J}^\top\boldsymbol{\lambda} + \boldsymbol{\delta}(t)\Big) + \dot{\mathbf{P}}_\zeta(t)\boldsymbol{\xi}_\zeta + \dot{\boldsymbol{\zeta}}^r\Big]
\end{aligned} \tag{7.32}$$

which can be written in compact form as:

$$\dot{\boldsymbol{\xi}} \triangleq h(\boldsymbol{\xi}, t) = [h_x^\top(\boldsymbol{\xi_x}, t), h_\zeta^\top(\boldsymbol{\xi_\zeta}, t)]^\top \tag{7.33}$$

Let us define the open set $\Omega_\xi = \Omega_{\xi_x} \times \Omega_{\xi_\zeta}$ with $\Omega_{\xi_x} \triangleq (-\underline{M}_f, \overline{M}_f) \times (-\underline{M}_y, \overline{M}_y) \times (-\underline{M}_z, \overline{M}_z) \times (-\underline{M}_{o_1}, \overline{M}_{o_1}) \times (-\underline{M}_{o_2}, \overline{M}_{o_2}) \times (-\underline{M}_{o_3}, \overline{M}_{o_3})$ and $\Omega_{\xi_\zeta} \triangleq (-1, 1)^n$. In what follows, we proceed in two phases. First we ensure the existence of a unique maximal solution $\boldsymbol{\xi}(t)$ of (7.33) over the set Ω_ξ for a time interval $[0, t_{\max})$ (i.e., $\boldsymbol{\xi}(t) \in \Omega_\xi, \forall t \in [0, t_{\max})$). Then, we prove that the proposed controller guarantees, for all $t \in [0, t_{\max})$ the boundedness of all closed loop signals as well as that $\boldsymbol{\xi}(t)$ remains strictly within the set Ω_ξ, which leads by contradiction to $t_{\max} = \infty$ and consequently to the satisfaction of (7.16) and (7.26), thus completing the proof.

Phase A: The set Ω_ξ is nonempty and open. Moreover (7.16) and (7.26) leads to $-\underline{M}_i < \xi_i(0) < \overline{M}_i$, $i \in \{f, y, z, o_1, o_2, o_3\}$ and $-1 < \xi_{\zeta_i}(0) < 1$, $i = 1, \ldots, n$. Thus, we guarantee that $\boldsymbol{\xi}_x(0) \in \Omega_{\xi_x}$ and $\boldsymbol{\xi}_\zeta(0) \in \Omega_{\xi_\zeta}$. Additionally, $h(\boldsymbol{\xi}, t)$, as defined in (7.33), is continuous on t and locally Lipschitz on $\boldsymbol{\xi}$ over Ω_ξ. Therefore, the hypotheses of the Theorem-11.2 hold and the existence of a maximal solution $\boldsymbol{\xi}(t)$ of (7.33) on a time interval $[0, t_{\max})$ such that $\boldsymbol{\xi}(t) \in \Omega_\xi,\ \forall t \in [0, t_{\max})$ is ensured.

Phase B: In Phase A, we proved that $\boldsymbol{\xi}(t) \in \Omega_\xi,\ \forall t \in [0, t_{\max})$, thus it can be concluded that:

$$\xi_i(t) = \frac{e_i(t)}{\rho_i(t)} \in (-\underline{M}_i, \overline{M}_i),\ i \in \{f, y, z, o_1, o_2, o_3\} \tag{7.34}$$

$$\xi_{\zeta_i}(t) = \frac{e_{\zeta_i}}{\rho_{\zeta_i}} \in (-1, 1),\ i = 1, \ldots, n \tag{7.35}$$

for all $t \in [0, t_{\max})$, from which we obtain that $e_i(t)$ and $e_{\zeta_i}(t)$ are lower and upper bounded by $-\underline{M}_i\rho_i(t), \overline{M}_i\rho_i(t)$ and $-\rho_{\zeta_i}(t), \rho_{\zeta_i}(t)$, respectively. Therefore, the transformed error vectors $\varepsilon_i(\xi_i)$ and $\varepsilon_{\zeta_i}(\xi_{\zeta_i})$ designated in (7.18) and (7.27), respectively, are well-defined for all $t \in [0, t_{\max})$. Hence, consider the positive definite and radially unbounded function:

$$V(\boldsymbol{\varepsilon}) = \frac{1}{2}\boldsymbol{\varepsilon}^\top(\boldsymbol{\xi}_x)\boldsymbol{\varepsilon}(\boldsymbol{\xi}_x) = \frac{1}{2}\varepsilon_f^2 + \frac{1}{2}||\boldsymbol{\varepsilon}_p||^2 + \frac{1}{2}||\boldsymbol{\varepsilon}_o||^2 \tag{7.36}$$

with $\boldsymbol{\varepsilon}_p = [\varepsilon_{p_1}, \varepsilon_{p_2}]^\top$ and $\boldsymbol{\varepsilon}_o = [\varepsilon_{o_1}, \varepsilon_{o_2}, \varepsilon_{o_3}]^\top$. Differentiating $V(\boldsymbol{\varepsilon})$ with respect to time and substituting (7.31) results in:

$$\begin{aligned}\dot{V} =& -\boldsymbol{\varepsilon}^\top(\boldsymbol{\xi}_x)\boldsymbol{R}(\boldsymbol{\xi}_x)\mathbf{P}^{-1}(t)\mathrm{diag}\Big[\partial\Phi(\chi), \boldsymbol{I}_{5\times5}\Big]\boldsymbol{K}\mathbf{P}^{-1}(t)\boldsymbol{R}(\boldsymbol{\xi}_x)\boldsymbol{\varepsilon}^\top(\boldsymbol{\xi}_x) \\ &+ \boldsymbol{\varepsilon}^\top(\boldsymbol{\xi}_x)\boldsymbol{R}(\boldsymbol{\xi}_x)\mathbf{P}^{-1}(t)\boldsymbol{B}_x\end{aligned} \tag{7.37}$$

where

$$\boldsymbol{B}_x = \mathrm{diag}\Big[\partial\Phi(\chi), \boldsymbol{I}_{2\times2}, \boldsymbol{L}\Big]\boldsymbol{J}\mathbf{P}_\zeta(t)\boldsymbol{\xi}_\zeta - \Big[\dot{f}^d, \dot{y}^d, \dot{z}^d, [\boldsymbol{L}\boldsymbol{\omega}^d]^\top\Big]^\top - \dot{\mathbf{P}}(t)\boldsymbol{\xi}_x \tag{7.38}$$

It is well known that the Jacobian $\boldsymbol{J}$ is bounded by definition. Moreover, since, $\dot{f}^d(t), \dot{y}^d(t)$, $\dot{z}^d(t), w^d(t)$, $\rho_i(t)\ i \in \{f, y, z, o_1, o_2, o_3\}$ and $\rho_{\zeta_i}(t),\ i = 1, \ldots, n$ are bounded by construction and ξ_i, ξ_{ζ_i} are also bounded within the compact sets Ω_{ξ_x} and Ω_{ξ_ζ} owing to (7.35), we conclude the existence of a position constant $\bar{\varepsilon}$ such that:

$$|\boldsymbol{\varepsilon}(\boldsymbol{\xi}_x(t))| \leq \bar{\varepsilon},\ \forall t \in [0, t_{\max}) \tag{7.39}$$

Furthermore, from (7.18), and invoking the inverse logarithmic function, we obtain:

$$-\underline{M}_i < \underline{\xi}_i \leq \xi_i(t) \leq \bar{\xi}_i < \overline{M}_i,\ \forall t \in [0, t_{\max}),\ i \in \{f, o_1, o_2, p_1, p_2, p_3\} \tag{7.40}$$

where:

$$\underline{\xi}_i = -\underline{M}_i \frac{\exp(\bar{\varepsilon}) - 1}{\exp(\bar{\varepsilon}) + \frac{\underline{M}_i}{\overline{M}_i}}, \quad \bar{\xi}_i = \overline{M}_i \frac{\exp(\bar{\varepsilon}) - 1}{\exp(\bar{\varepsilon}) + \frac{\overline{M}_i}{\underline{M}_i}} \tag{7.41}$$

Owing to (7.40) and (7.23) it can concluded that the reference velocity vector $\boldsymbol{\zeta}^r$ remains bounded for all $t \in [0, t_{\max})$ as well. Moreover, invoking $\boldsymbol{\zeta} = \boldsymbol{\zeta}^r(t) + \mathbf{P}_\zeta(t)\boldsymbol{\xi}_\zeta$ from (7.28), we also conclude the boundedness of $\boldsymbol{\zeta}$ for all $t \in [0, t_{\max})$. Finally, differentiating $\boldsymbol{\zeta}^r(t)$ w.r.t time and employing (7.31), (7.35) and (7.40), we conclude the boundedness of $\dot{\boldsymbol{\zeta}}^r(t)$, $\forall t \in [0, t_{\max})$ too.

Now let us consider the positive definite and radially unbounded function $V_\zeta(\boldsymbol{\varepsilon}_\zeta) = \frac{1}{2}||\boldsymbol{\varepsilon}_\zeta||^2$. Differentiating V_ζ with respect to time, substituting (7.32) and employing the continuity of $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{g}$, $\boldsymbol{\lambda}$, $\boldsymbol{\delta}$, $\boldsymbol{\xi}_x$, $\boldsymbol{\xi}_\zeta$,$\dot{\mathbf{P}}_\zeta$, $\dot{\boldsymbol{\zeta}}^r$, $\forall t \in [0, t_{\max})$, we obtain:

$$\dot{V}_\zeta \leq ||\mathbf{P}_\zeta^{-1}\boldsymbol{R}_\zeta(\boldsymbol{\xi}_\zeta)\boldsymbol{\varepsilon}_\zeta|| \Big(B_\zeta - \lambda_M \boldsymbol{K}_\zeta ||\mathbf{P}_\zeta^{-1}\boldsymbol{R}_\zeta(\boldsymbol{\xi}_\zeta)\boldsymbol{\varepsilon}_\zeta|| \Big)$$

$\forall t \in [0, t_{\max})$, where λ_M is the minimum eigenvalue of the positive definite matrix $\boldsymbol{M}^{-1}$ and B_ζ is a positive constant independent of $t_{\max}$, that satisfies:

$$B_\zeta \geq ||\boldsymbol{M}^{-1}\Big(\boldsymbol{C}\cdot(\mathbf{P}_\zeta\boldsymbol{\xi}_\zeta + \boldsymbol{\zeta}^r(t)) + \boldsymbol{D}\cdot(\mathbf{P}_\zeta\boldsymbol{\xi}_\zeta + \boldsymbol{\zeta}^r(t)) + \boldsymbol{g} + \boldsymbol{J}^\top\boldsymbol{\lambda} + \boldsymbol{\delta}(t) + \dot{\mathbf{P}}_\zeta\boldsymbol{\xi}_\zeta + \dot{\boldsymbol{\zeta}}^r\Big)||$$

Thus, we conclude that:

$$||\boldsymbol{\varepsilon}_\zeta(\boldsymbol{\xi}_\zeta)|| \leq \bar{\varepsilon}_\zeta \ , \forall t \in [0, t_{\max}) \tag{7.42}$$

Furthermore, from (7.30) and invoking $|\varepsilon_{\zeta_i}| \leq \bar{\varepsilon}_\zeta$, we obtain:

$$-1 < \underline{\xi}_{\zeta_i} \leq \xi_{\zeta_i}(t) \leq \bar{\xi}_{\zeta_i} < 1, \ \forall t \in [0, t_{\max}), \ i = 1, \ldots, n \tag{7.43}$$

where

$$\underline{\xi}_{\zeta_i} = \frac{-\exp(\bar{\varepsilon}_\zeta) - 1}{-\exp(\bar{\varepsilon}_\zeta) + 1}, \quad \bar{\xi}_{\zeta_i} = \frac{\exp(\bar{\varepsilon}_\zeta) - 1}{\exp(\bar{\varepsilon}_\zeta) + 1} \tag{7.44}$$

which also leads to the boundedness of the control law (7.29) for all $t \in [0, t_{\max})$.

Subsequently, we will show that $t_{\max}$ can be extended to infinity. Obviously, notice by (7.40) and (7.43) that $\boldsymbol{\xi}(t) \in \Omega'_\xi \triangleq \Omega'_{\xi_x} \times \Omega'_{\xi_\zeta}, \forall t \in [0, t_{\max})$, where:

$$\begin{aligned}
\Omega'_{\xi_x} =& [\underline{\xi}_f, \bar{\xi}_f] \times [\underline{\xi}_y, \bar{\xi}_y], \times [\underline{\xi}_z, \bar{\xi}_z], \times [\underline{\xi}_{o_1}, \bar{\xi}_{o1}], \times [\underline{\xi}_{o_2}, \bar{\xi}_{o2}], \times [\underline{\xi}_{o_3}, \bar{\xi}_{o3}] \\
\Omega'_{\xi_\zeta} =& [\underline{\xi}_{\zeta_1}, \bar{\xi}_{\zeta_1}] \times \ldots \times [\underline{\xi}_{\zeta_n}, \bar{\xi}_{\zeta_n}],
\end{aligned}$$

are nonempty and compact subsets of Ω_{ξ_x} and Ω_{ξ_ζ}, respectively. Hence, assuming that $t_{\max} < \infty$ and since $\Omega_\xi^{'} \subset \Omega_\xi$, Proposition-11.1 dictates the existence of a time instant $t^{'} \in [0, t_{\max})$ such that $\boldsymbol{\xi}(t^{'}) \notin \Omega_\xi^{'}$, which is a clear contradiction. Therefore, $t_{\max} = \infty$. Thus, all closed loop signals remain bounded and moreover $\boldsymbol{\xi}(t) \in \Omega_\xi^{'}, \forall t \geq 0$. Finally, from (7.19) and (7.40) we conclude that:

$$-\underline{M}_i \rho_i(t) < -\underline{M}_i \frac{\exp(\bar{\varepsilon}) - 1}{\exp(\bar{\varepsilon}) + \frac{\underline{M}_i}{\overline{M}_i}} \rho_i(t) \leq e_i(t) \leq \overline{M}_i \frac{\exp(\bar{\varepsilon}) - 1}{\exp(\bar{\varepsilon}) + \frac{\overline{M}_i}{\underline{M}_i}} \rho_i(t) < \overline{M}_i \rho_i(t) \quad (7.45)$$

for $i \in \{f, y, z, o_1, o_2, o_3\}$ and for all $t \geq 0$, which completes the proof. □

Remark 7.1. From the aforementioned proof, it is worth noticing that the proposed control scheme does not incorporate any information regarding the matrices $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{g}$, the deformation model $\Phi(\chi)$ or the external disturbances $\boldsymbol{\delta}(t)$, which all affect only the size of $\bar{\varepsilon}$ and $\bar{\varepsilon}_\zeta$ but leave unaltered the achieved convergence properties as (7.45) dictates. In fact, the actual transient and steady state performance is determined by the selection of the performance functions $\rho_i(t)$ and performance parameters $\underline{M}_i$, $\overline{M}_i$, $i \in \{f, y, z, o_1, o_2, o_3\}$. More specifically, for any initial force/position/orientation tracking error, the performance functions $\rho_i(t)$ and parameters $\underline{M}_i$, $\overline{M}_i$, $i \in \{f, y, z, o_1, o_2, o_3\}$ are selected such that: i) $-\underline{M}_i < e_i(0) < \overline{M}_i$ and ii) the desired transient and steady state performance specifications are incorporated.

Remark 7.2. It should be noted that the selection of the control gains affects both the quality of evolution of the errors e_i, $i \in \{f, y, z, o_1, o_2, o_3\}$ inside the corresponding performance envelopes as well as the control input characteristics (e.g., decreasing the gain values leads to increased oscillatory behavior within the prescribed performance envelope described by (7.45), which is improved when adopting higher values, enlarging, however, the control effort both in magnitude and rate). Additional fine tuning might be needed in real-time scenarios to retain the required control input signals within the feasible range that can be implemented by the actuators. Similarly, the control input constraints impose an upper bound on the required speed of convergence of $\rho_i(t)$, $i \in \{f, y, z, o_1, o_2, o_3\}$. Hence, the selection of the control gains k_i , $i \in \{f, y, z, o_1, o_2, o_3\}$ and $\boldsymbol{K}_\zeta$ can have positive influence on the overall closed loop system response. More specifically, notice that (7.39) and (7.42) provide bounds on $\boldsymbol{\varepsilon}(t)$ and $\boldsymbol{\varepsilon}_\zeta(t)$. Therefore, invoking (7.20) and (7.29) we can select the control gains $k_i, i \in \{f, y, z, o_1, o_2, o_3\}$ and $\boldsymbol{K}_\zeta$ such that $\boldsymbol{\zeta}^r$ and $\boldsymbol{\tau}$ are retained within certain bounds. Nevertheless, (7.39) and (7.42) involve via the terms $\boldsymbol{B}_x$ and $\boldsymbol{B}_\zeta$ the parameters of the model, the external disturbances and the desired performance specifications. Thus, an upper bound of the dynamic parameters of the system as well as of the exogenous disturbances should be given in order to extract any relationships between the achieved performance and the input constraints. Finally, in the same vein, the selection of the velocity performance functions $\rho_{\zeta_i}(t), i = 1, \ldots, n$

affects similarly both the evolution of the force/position/orientation errors within the corresponding performance envelopes as well as the control input characteristics.

7.4 Results

In this section, the theoretical findings are verified via both simulation and experimental studies. The simulation results were conducted using the dynamic simulation environment built in MATLAB® presented in Section 2.4. The experimental results were conducted in a test tank employing a small UVMS.

7.4.1 Simulation study

We considered a scenario involving 3D motion, where the end-effector of the UVMS is in compliant contact with a planar surface with stiffness model $f = k\chi^2$, $k = 300\frac{N}{m^2}$ which is unknown to the controller. The sampling time is equal to 0.1 *sec*, which is common in a real time operation with an underwater robotic system. The initial configuration is depicted in Fig.7.1. We assumed that the UVMS is in contact with the compliant environment exerting a force normal to the surface $f(0) = 0.5\ N$. The desired force profile along the normal to the surface direction is set as $f^d(t) = 4N$. Moreover, the UVMS should track a desired position trajectory on the surface and attain a perpendicular orientation (i.e., $\boldsymbol{R}^d = I_{3\times3}$) with respect to the surface. The constants $\underline{f}^*$ and $\overline{f}^*$ are chosen as $\underline{f}^* = 0.1N$ and $\overline{f}^* = 5N$ in order to ensure a priori that the contact with the surface is never lost and that excessive interaction forces are avoided. In this spirit, as it was described earlier, we set $\underline{M}_f = 3.9$, $\overline{M}_f = 1$. All other performance parameters at the kinematic level are given in Table-7.1. Moreover, the control gains were selected as

TABLE 7.1: Performance parameters- First level

Parameter	Value
$\overline{M}_{p_{1,2}}$, $\underline{M}_{p_{1,2}}$	0.2
$\overline{M}_{o_{1,2,3}}$, $\underline{M}_{o_{1,2,3}}$	0.3
ρ^{∞}_{f}	0.4
$\rho^{\infty}_{p_{1,2}}$	0.03
$\rho^{\infty}_{o_{1,2,3}}$	0.07
$l_i, i\{f, p_{1,2}, o_{1,2,3}\}$	2.0

$k_i = 0.2,\ j \in \{f, p_1, p_2, o_1, o_2, o_3\}$. The design parameters regarding to the second level were chosen as: $\rho^0_{\zeta_i} = 1.0$, $\boldsymbol{K}_\zeta = 2I_{n\times n}$. Notice that the dynamic parameters of the UVMS as well as the stiffness of the planar surface were considered unknown for the controller. Furthermore, the secondary task velocities (7.23) were designed appropriately to avoid

the violation of the manipulator joint limits. However, more complicated secondary tasks (e.g., manipulability) could also be incorporated following the task priority based control techniques [145, 173]. In addition, in order to test the robustness of the proposed scheme, in the subsequent simulation study the dynamics of the UVMS were affected by external disturbances in the form of slowly time varying sea currents acting along x, y and z axes modeled by the corresponding velocities $v_i^{\{I\}} = 0.1\sin(\frac{\pi}{25}t)\frac{m}{s}$, $i \in \{x, y, z\}$. Finally, bounded measurement noise of normal distribution with 5% standard deviation was considered during the simulation study.

The results are depicted in Fig.7.4-Fig.7.7. The evolution of the position trajectory on the surface is presented in Fig.7.4. It can be easily observed that the actual position of the end-effector (indicated by red color) converges to the desired one (indicated by green color) and follows the desired trajectory profile. Fig.7.5 presents the evolution of the actual force exerted by the UVMS with respect to the desired force profile. Moreover, it can be seen that the force exerted by the UVMS remained inside the desired region, securing that the contact is never lost and the exerted force never overshoots the predefined value. The evolution of the errors at the first and second level of the proposed controller are illustrated in Fig.7.6 and Fig.7.7, respectively. It can be concluded that even with the influence of external disturbances as well as measurements noise, the errors in all directions converge close to zero and remain bounded below and above by the corresponding performance functions.

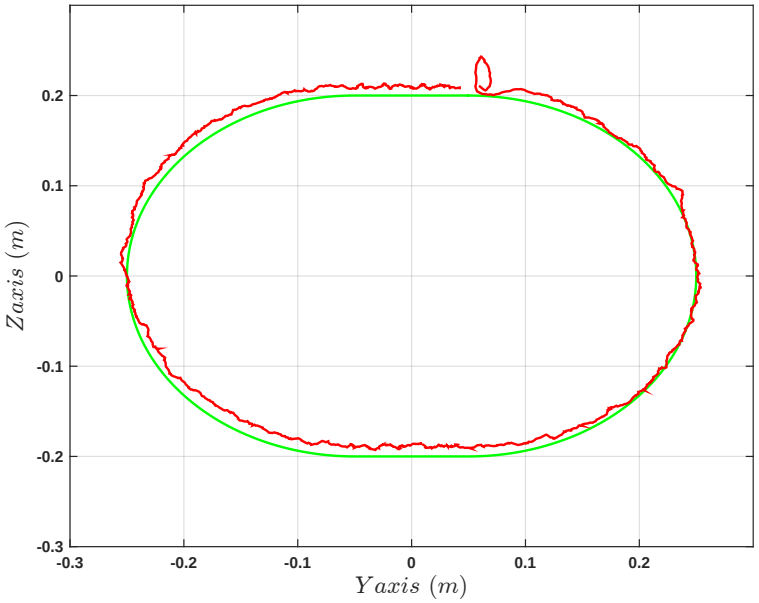


FIGURE 7.4: The evolution of the position trajectory on the surface. The desired trajectory and the actual position of the end-effector on the surface are indicated by green and red color respectively. The end-effector position converges and follows the desired trajectory profile.

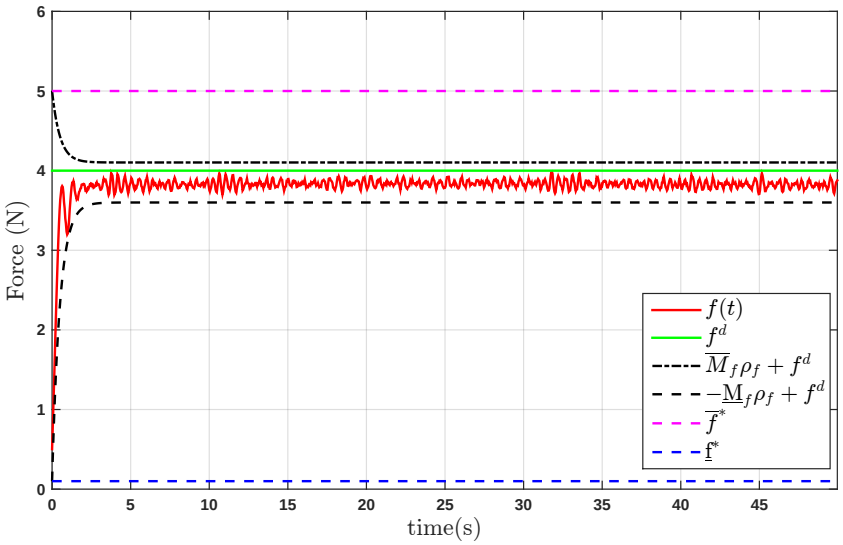


FIGURE 7.5: The evolution of the force trajectory. The desired constant force and the actual force exerted by the UVMS are indicated by green and red color respectively.

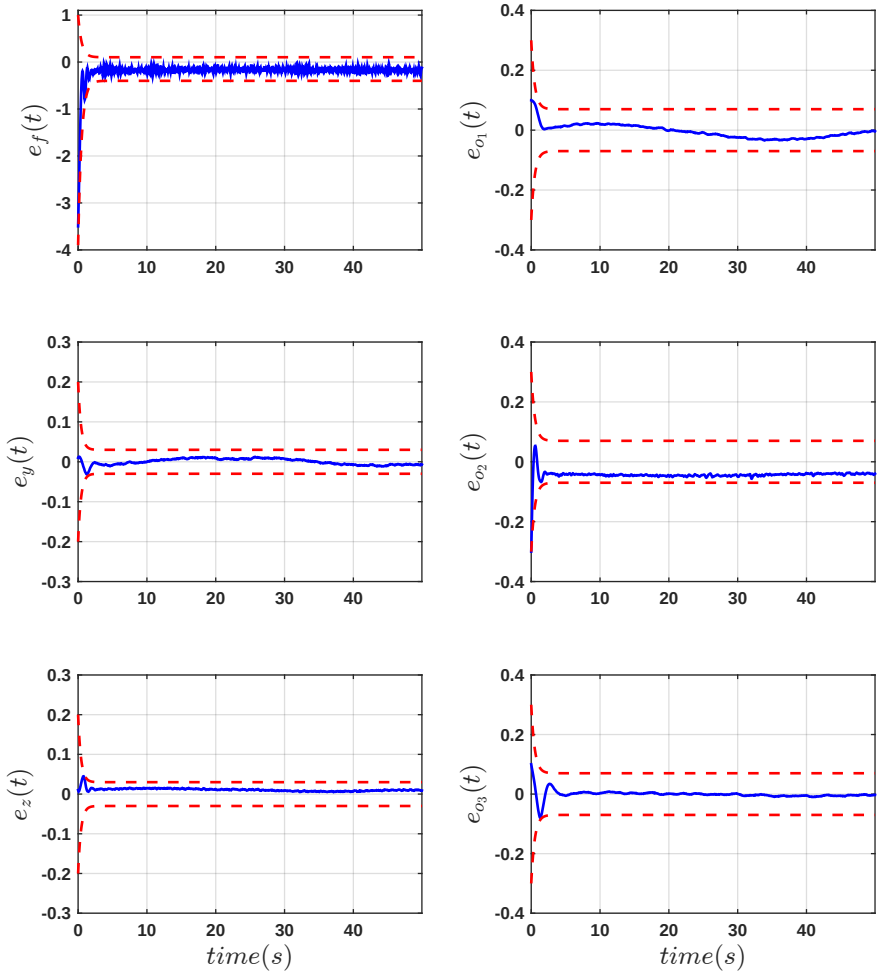


FIGURE 7.6: The evolution of the errors at the first level of the proposed control scheme. The errors and performance bounds are indicated by blue and red color respectively.

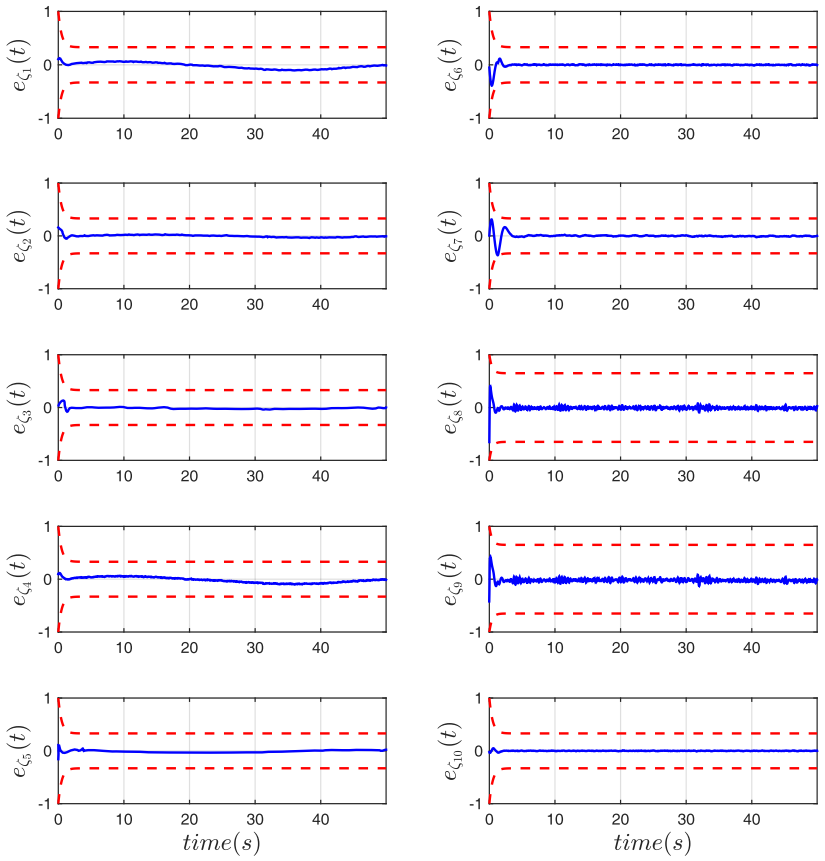


FIGURE 7.7: The evolution of the errors at the second level of the proposed control scheme. The errors and performance bounds are indicated by blue and red color respectively.

7.4.2 Experimental Results

This section demonstrates the efficacy of the proposed position/force control scheme via an experimental procedure with a small UVMS.

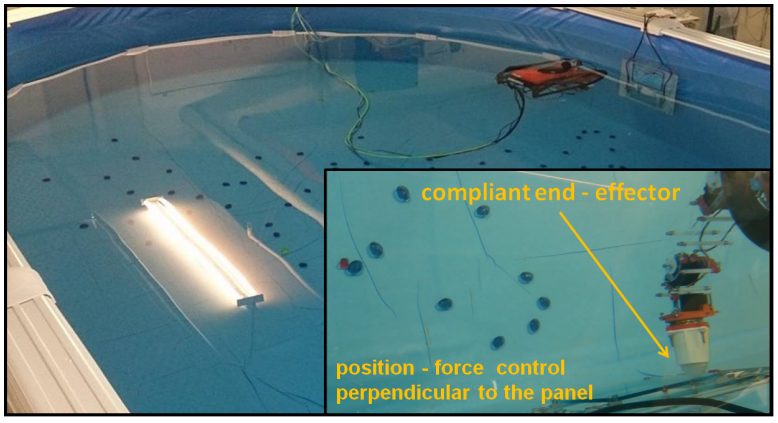


FIGURE 7.8: The *NTUA, Control Systems Lab* test tank and Position–Force Control Scenario

7.4.2.1 Experimental Setup

The experiments were carried out inside the *NTUA, Control Systems Lab* test tank, with dimensions $5m \times 3m \times 1.5m$ (Fig. 7.8). In the bottom of the tank, sparse visual features have been added in order to improve optical-flow velocity estimations. The vehicle used in this chapter is a 4 DoFs Seabotix LBV [249], actuated in Surge, Sway, Heave and Yaw via a 4 thruster set configuration. The vehicle is equipped with a small custom-made 4 DoFs underwater manipulator, with revolute joints and a compliant (spring-based) end-effector. The system is equipped with two cameras: i) a down-looking Sony PlayStation Eye camera, with 640×480 pixels at 30 frames per second (fps) enclosed in a water-proof housing and ii) a forward looking camera (Seabotix LBV default camera) with 640×480 pixels at 25 frames per second (fps). The vehicle is also equipped with an $SBG\ IG-500A$ AHRS, delivering temperature-compensated 3D acceleration, angular velocity and orientation measurements at $100Hz$. Finally, a marker localization system based on the *ArUco* library [190] is employed in order to determine the pose of the vehicle with respect to the panel (see Fig. 7.9). A state estimation algorithm based on the Complementary Filters notion, as in [191], delivers the pose, velocity and acceleration estimates of the underwater vehicle by fusing data from the available on-board sensors while the position and velocity of the arm joints are available via the on-board encoders. The force measurements along the desired direction (perpendicular to the panel) are

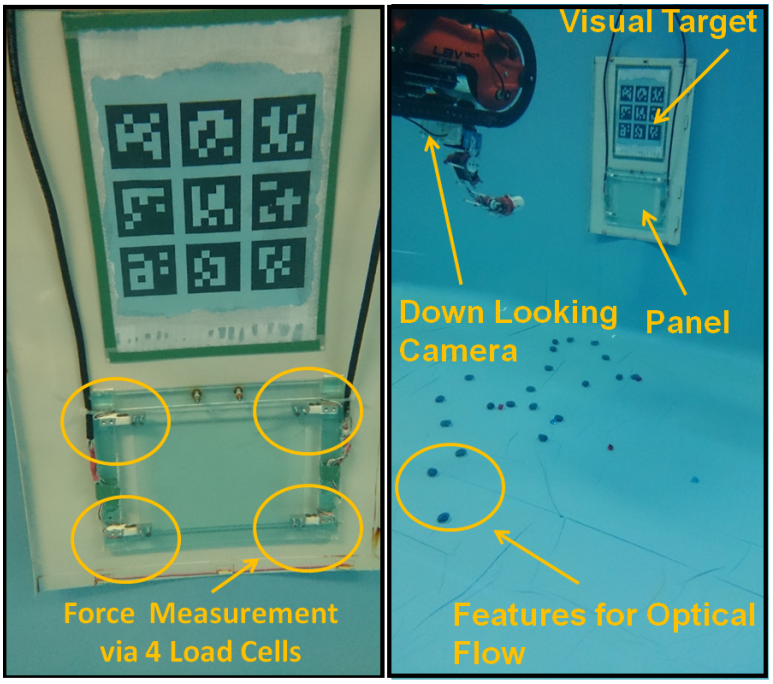


FIGURE 7.9: a) Load cells mounted on the panel, b)Visual features for state estimation

acquired by 4 load–cells, properly mounted on the base corners of the panel, as shown in Fig. 7.9. At each time instant, the sum of the 4 load–cells is incorporated as force feedback to the system. The load–cells are connected directly to the control PC, via a

PhidgetBridge data I/O device. Finally, the software implementation of the state estimation algorithm as well as of the proposed position/force control scheme was conducted in C++ and Python under the Robot Operating System (ROS) [192].

7.4.2.2 Experimental results

This subsection demonstrates experimental results obtained by the proposed position/-force control scheme. In this scenario, the UVMS end-effector should follow a line trajectory along the panel, while maintaining a normal constant force to the panel. We consider that the end-effector has already achieved contact with the panel. The desired force is set to $f^d = 5N$, while the end-effector must perform a periodical motion between $-0.05m$ and $0.05m$ along the panel y local axis, while the position along z local axis should remain constant. The state responses are shown in Fig. 7.10-7.12, while the trajectory of the end-effector on the plane is depicted in Fig. 7.13. As it can be easily observed, the proposed position/force control scheme fulfilled the interaction task successfully.

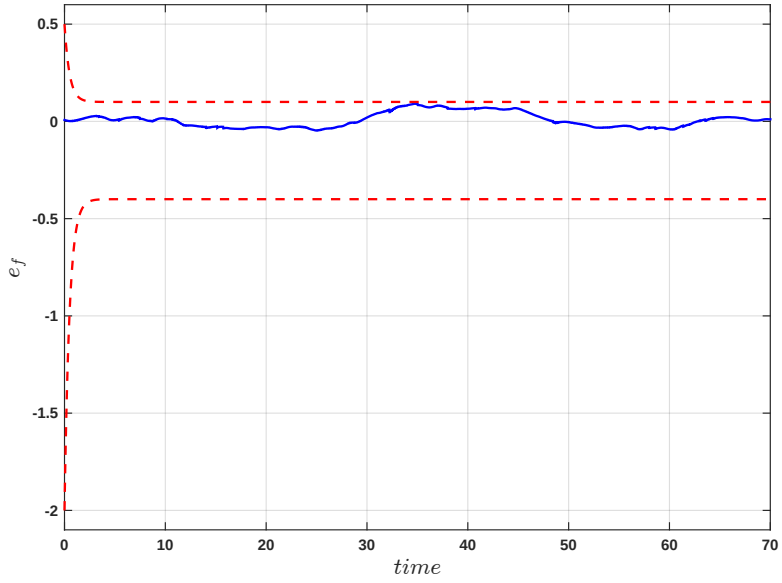


FIGURE 7.10: The evolution of the force error during the experiment. Perpendicular force f along x panel local axis.

7.5 Video

A video demonstrating the aforementioned experimental result of the proposed methodology can be found in a HD video at the following url: https://youtu.be/PCHLsUb-vM0. Alternatively, it can be found in the attached dvd as file: video_ch7.mpg

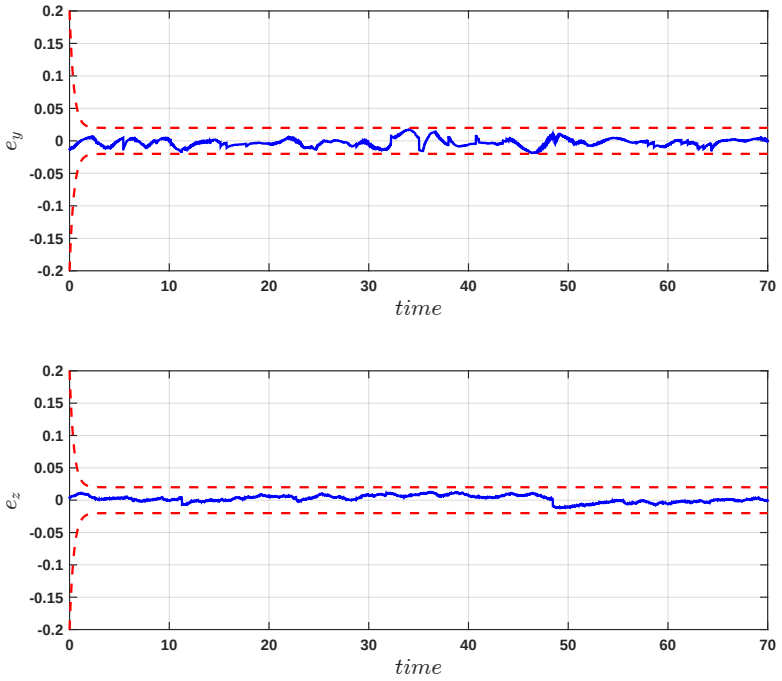


FIGURE 7.11: The evolution of the end-effector position errors along y and z direction during the experiment.

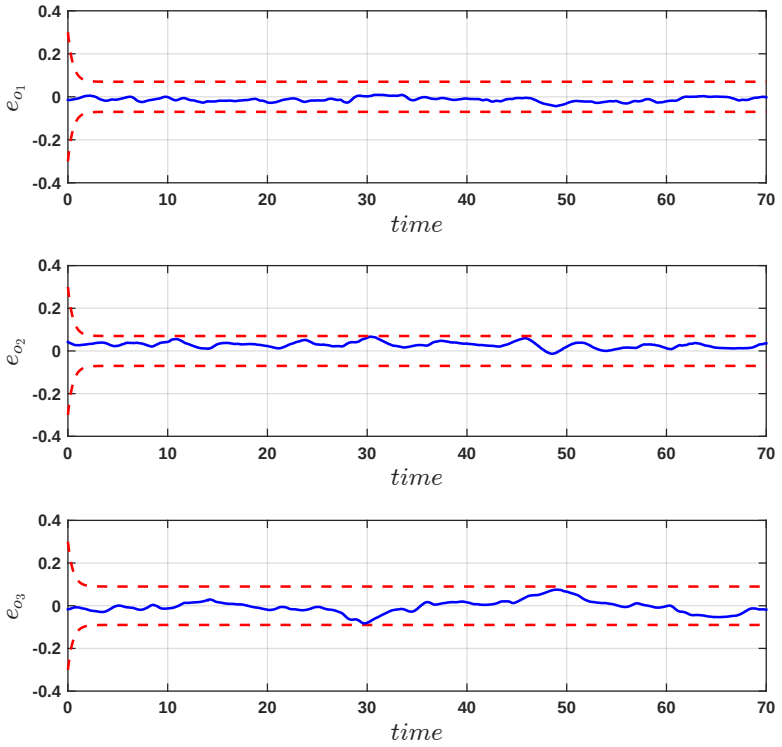


FIGURE 7.12: The evolution of the end-effector orientation errors direction during the experiment.

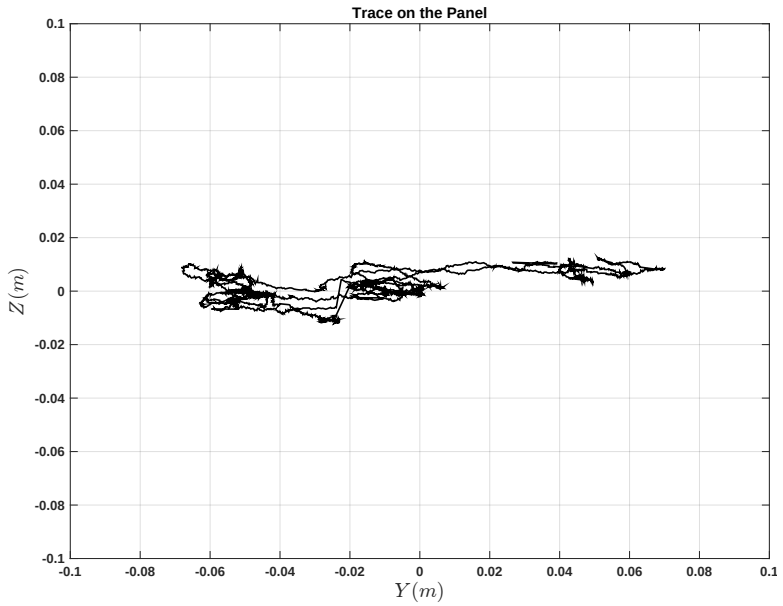


FIGURE 7.13: The trajectory of the end-effector w.r.t the horizontal plane. Perpendicular force is kept constant at all times.

7.6 Conclusions

In this chapter we presented a robust force/position control scheme for a UVMS in compliant contact with the environment, with great applications in underwater robotics (e.g. sampling of the sea organisms, underwater welding, object handling). The proposed control scheme does not required any a priori knowledge of the UVMS dynamical parameters or the stiffness model. It guarantees a predefined behavior in terms of desired overshoot, transient and steady state response and it is robust with respect to external disturbances and measurement noises. Moreover, the proposed controller exhibits the following important characteristics: i) it is of low complexity and thus can be easily used in most UVMSs ii) the performance of the proposed scheme (e.g., desired overshoot, steady state response) is a priori and explicitly imposed by certain designer-specified performance functions, and is fully decoupled by the control gains selection, thus simplifying the control design.

Chapter 8

Cooperative Impedance Control for Multiple Underwater Vehicle Manipulator Systems under Lean Communication

In this chapter we address the problem of cooperative object transportation for multiple Underwater Vehicle Manipulator Systems (UVMSs) in a constrained workspace involving static obstacles, where the coordination relies solely on implicit communication arising from the physical interaction of the robots with the commonly grasped object. We propose a novel distributed leader-follower architecture, where the leading UVMS, which has knowledge of the object's desired trajectory, tries to achieve the desired tracking behavior via an impedance control law, navigating in this way, the overall formation towards the goal configuration while avoiding collisions with the obstacles. On the other hand, the following UVMSs estimate locally the object's desired trajectory via a novel prescribed performance estimation law and implement a similar impedance control law achieving in this way tracking of the desired trajectory despite the uncertainty and external disturbance in the object and the UVMS dynamics respectively. The feedback relies on each UVMS's force/torque measurements and no explicit data is exchanged online among the robots, thus reducing the required communication bandwidth and increasing robustness. Moreover, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities. Finally, various simulation studies clarify the proposed method and verify its efficiency.

8.1 Introduction

In this chapter, the problem of decentralized cooperative object transportation considering multiple UVMSs in a constrained workspace with static obstacles is addressed. The challenge lays in replacing explicit communication with implicit, by incorporating sensor data that result from the physical interaction of the robots with the commonly grasped object (i.e., we assume that each UVMS is equipped with a force/torque sensor attached on its end-effector). The proposed scheme is based on a leader-follower architecture, where the leader, which is aware of the object's desired trajectory, implements it via an impedance control law. On the other hand, the followers estimate in a distributed way, the desired trajectory via observing the object motion and impose a similar impedance law. All impedance laws linearize the dynamics and incorporate coefficients for load sharing. The estimation process is based on the prescribed performance methodology [246] that drives the estimation error to an arbitrarily small residual set. Moreover, we design adaptive control laws in order to compensate for the parametric uncertainty of the UVMSs dynamics as well as the external disturbances. Finally, it should be noticed that the proposed scheme exploits information i.e., force/torque at the end-effector, position and velocity measurements acquired solely by onboard sensors (e.g., via a fusion algorithm on measurements of various sensors such as IMU, USBL and DVL), avoiding thus any tedious inter-robot explicit communication.

One should bear in mind that although underwater vehicles are equipped with acoustic modems to communicate with the surface control station[1], employing implicit communication based cooperative control protocols is clearly motivated by the limited bandwidth of acoustic communication devices. Moreover, in order to achieve collision avoidance, either the leader has to transmit online the desired object trajectory to the followers, or all UVMSs should obtain a mutually agreed desired trajectory of the object, which necessitates for an accurate common localization system [171] that is extremely challenging and prone to errors in underwater environments. On the contrary, in the proposed scheme, it is worth noting that each follower estimates locally and in a distributed way, the desired object trajectory relatively to its inertial frame, employing exclusively its own measurements (position, velocity and force/toque). In this way, although the proposed control strategy does not remove all practical needs for communication in underwater intervention tasks, (e.g., for safety, adaptability and efficiency) nevertheless, it relieves the team of robots from intense inter-robot communication during the execution of the collaborative tasks. This, consequently, increases significantly the robustness of the cooperative scheme and furthermore avoids any restrictions imposed by the acoustic communication

[1]For example, all cooperating UVMSs need to know the initial position of the object to be transported, in order to reach and grasp it, and coordinate discrete phases of the tasks via simple high level messages (e.g., "Ok, I've grasped it", "Let us proceed").

bandwidth (e.g., the number of participating UVMSs). Additionally, we extend the current state of art in implicit communication-based cooperative manipulation [250, 251], via a more robust estimation algorithm that converges even though the desired object's acceleration profile is non-zero (i.e., for an arbitrary object's desired trajectory profile as long as it is bounded and smooth). Finally, the customizable ultimate bounds allow us to achieve practical stabilization of the estimation error, with accuracy limited only by the sensors' resolution.

The rest of the manuscript is organized as follows: Section II introduces the problem and describes the system model. The proposed control methodology is presented in Section III. Section IV validates our approach via simulated paradigms. Finally, Section V concludes our work.

8.2 Problem Formulation

Consider $N+1$ UVMSs under a single leader - multiple followers architecture, rigidly grasping an object[2] within a constrained workspace with static obstacles (see Fig **??**). We also assume that each vehicle is actuated in all 6 Degrees of Freedoms (DoFs) and is equipped with an n DoFs manipulator. Thus, each UVMS is fully-actuated at its end-effector frame. This assumption implies that all UVMSs are able to exert arbitrary forces and torques on the object along and around any direction. It should also be noted that in the proposed scheme, only the leading robot is aware of the obstacles' position in the workspace and the object's desired configuration $\boldsymbol{x}_o^d$. On the other hand, the followers estimate locally in a distributed way the object's desired trajectory profile and manipulate the object in coordination with the leader based solely on their own sensory information. Moreover, we assume that the UVMSs are equipped with appropriate sensors, that allow them to measure their position and velocity (e.g., employing a fusion technique based on measurements by various onboard sensors such as USBL, IMU, DVL and depth-sensor), as well as the interaction forces/torques with the object via a force/toque sensor. Additionally, the geometric parameters of the both UVMSs and the commonly grasped object are considered known, whereas their dynamic parameters are completely unknown. Moreover, the control of each UVMS will be designed based on a commonly agreed frame on a specific feature of the object, which could be identified employing a visual detection system, owing to the fact that the limited underwater visibility is not an issue when all robots are close to the object of interest. Finally, notice that owing to: i) the strict communication constraints (i.e., online inter-robot communication

[2]The end-effector frame of each UVMS is always constant relative to the object's body fixed frame.

is not permitted), ii) the model uncertainties of UVMSs (common problem in underwater robotics) and iii) the constrained workspace, the problem becomes very challenging, with no previously reported results in the related literature.

8.2.1 Kinematics

Consider $N+1$ UVMSs operating in a bounded workspace $\mathscr{W} \subseteq \mathbb{R}^3$. We denote the coordinates of the commonly agreed body-fixed frame on the object as well as the leader's and followers' task space (i.e., end-effector) coordinates by $\boldsymbol{x}_O = [\boldsymbol{\eta}_{1,O}^\top, \boldsymbol{\eta}_{2,O}^\top]^\top$, $\boldsymbol{x}_L = [\boldsymbol{\eta}_{1,L}^\top, \boldsymbol{\eta}_{2,L}^\top]^\top$ and $\boldsymbol{x}_{F_i} = [\boldsymbol{\eta}_{1,F_i}^\top, \boldsymbol{\eta}_{2,F_i}^\top]^\top,\ i \in \mathscr{N} = \{1, \ldots, N\}$ respectively. More specifically, $\boldsymbol{\eta}_{1,i} = [x_i, y_i, z_i]^\top$ and $\boldsymbol{\eta}_{2,i} = [\phi_i, \theta_i, \psi_i]^\top,\ i \in \{O, L, F_1, \ldots, F_N\}$ denote the position and the orientation expressed in Euler angles representation with respect to the inertial frame. Alternatively, the orientation coordinates $\boldsymbol{\eta}_{2,i}\ i \in \{O, L, F_1, \ldots, F_N\}$ expressed in Euler angles may be described by a rotation matrix $\boldsymbol{R}_i = [\boldsymbol{n}_i, \boldsymbol{o}_i, \boldsymbol{\alpha}_i] \in \mathbb{R}^3$ that is mainly employed owing to its physical meaning (i.e., the inertial coordinate frame after three successive rotations of ψ_i, θ_i, ϕ_i angles about its z, y and x axes respectively ends up parallel to the object-fixed coordinate) [245]. Thus, the rotation matrix $\boldsymbol{R}_i\ i \in \{O, L, F_1, \ldots, F_N\}$ may be expressed via $\boldsymbol{\eta}_{2,i}$ as follows:

$$\boldsymbol{R}_i = \begin{bmatrix} c_{\psi_i} c_{\theta_i} & c_{\psi_i} s_{\theta_i} s_{\phi_i} - s_{\psi_i} c_{\phi_i} & c_{\psi_i} s_{\theta_i} c_{\phi_i} + s_{\psi_i} s_{\phi_i} \\ s_{\psi_i} c_{\theta_i} & s_{\psi_i} s_{\theta_i} s_{\phi_i} + c_{\psi_i} c_{\phi_i} & s_{\psi_i} s_{\theta_i} c_{\phi_i} - c_{\psi_i} s_{\phi_i} \\ -s_{\theta_i} & c_{\theta_i} s_{\phi_i} & c_{\theta_i} c_{\phi_i} \end{bmatrix} \tag{8.1}$$

where $s_\star = \sin(\star)$ and $c_\star = \cos(\star)$. Let $\boldsymbol{q}_i = [\boldsymbol{q}_{v,i}^\top,\ \boldsymbol{q}_{m,i}^\top]^\top \in \mathbb{R}^{6+n}, i \in \mathscr{K} = \{L, F_1, \ldots, F_N\}$ be the joint state variables of each UVMS, where $\boldsymbol{q}_{v,i} \in \mathbb{R}^6$ is the vector that involves the position and the orientation of the vehicle and $\boldsymbol{q}_{m,i} \in \mathbb{R}^n$ is the vector of the angular positions of the manipulator's joints. Let also define the object as well as the leader's and followers' end effector generalized velocities by $\boldsymbol{v}_O = [\dot{\boldsymbol{\eta}}_{1,O}^\top, \boldsymbol{\omega}_O^\top]^\top$, $\boldsymbol{v}_L = [\dot{\boldsymbol{\eta}}_{1,L}^\top, \boldsymbol{\omega}_L^\top]^\top$ and $\boldsymbol{v}_i = [\dot{\boldsymbol{\eta}}_{1,i}^\top, \boldsymbol{\omega}_i^\top]^\top,\ i \in \{F_1, \ldots, F_N\}$ respectively, where $\dot{\boldsymbol{\eta}}_{1,i}$ and $\boldsymbol{\omega}_i$ denote the linear and angular velocity respectively. Without any loss of generality, for the augmented UVMS system we get [10]:

$$\boldsymbol{v}_i = \boldsymbol{J}_i(\boldsymbol{q}_i)\boldsymbol{\zeta}_i,\ i \in \mathscr{K} \tag{8.2}$$

where $\boldsymbol{\zeta}_i = [\boldsymbol{v}_i^\top, \dot{\boldsymbol{q}}_{m,i}^\top]^\top \in \mathbb{R}^{6+n}$ is the velocity vector involving the body velocities of the vehicle as well as the joint velocities of the manipulator with $\boldsymbol{v}_i$ to be the velocity of the vehicle expressed in the body-fixed frame and $\boldsymbol{J}_i(\boldsymbol{q}_i)$ is the geometric Jacobian matrix

[10]. Furthermore, owing to the rigid grasp of the object, the following equations hold:

$$\boldsymbol{x}_i = \boldsymbol{x}_O + \begin{bmatrix} {}^I\boldsymbol{R}_O \boldsymbol{l}_i \\ \boldsymbol{\alpha}_i \end{bmatrix}, \ i \in \mathscr{K} \tag{8.3}$$

where the vectors $\boldsymbol{l}_i = [l_{ix}, l_{iy}, l_{iz}]^\top$ and $\boldsymbol{\alpha}_i = [\alpha_{ix}, \alpha_{iy}, \alpha_{iz}]^\top, \ i \in \mathscr{K}$ represent the *constant* relative position and orientation of the end-effector w.r.t the object, expressed in the object's frame and ${}^I\boldsymbol{R}_O$ denotes the rotation matrix which describes the orientation of the object expressed in the inertial frame $\{I\}$. Thus, using (8.3) each UVMS can compute the object's position w.r.t inertial frame $\{I\}$, since the object geometric parameters are considered known. Furthermore, along with the fact that, due to the grasping rigidly, it holds that $\boldsymbol{\omega}_i = \boldsymbol{\omega}_O, \ i \in \mathscr{K}$, one obtains:

$$\boldsymbol{v}_i = \boldsymbol{J}_{iO} \boldsymbol{v}_O, \ i \in \mathscr{K} \tag{8.4}$$

where $\boldsymbol{J}_{iO}, \ i \in \mathscr{K}$ denotes the Jacobian from the end-effector of each UVMS to the object's center of mass, that is defined as:

$$\boldsymbol{J}_{iO} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & -\boldsymbol{S}(\boldsymbol{l}_i) \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \in \mathbb{R}^{6\times6}, \quad i \in \mathscr{K}$$

where $\boldsymbol{S}(\boldsymbol{l}_i)$ is the skew-symmetric matrix of vector $\boldsymbol{l}_i = [l_{ix}, l_{iy}, l_{iz}]^\top$ defined as:

$$\boldsymbol{S}(\boldsymbol{l}_i) = \begin{bmatrix} 0 & -l_{iz} & l_{iy} \\ l_{iz} & 0 & -l_{ix} \\ -l_{iy} & l_{ix} & 0 \end{bmatrix} \in \mathbb{R}^{3\times3}, \ i \in \mathscr{K}$$

Notice that $\boldsymbol{J}_{iO}, \ i \in \mathscr{K}$ are always full-rank owing to the grasp rigidity and hence obtain a well defined inverse. Thus, the object's velocity can be easily computed via the inverse of (8.4). Moreover, from (8.4), one obtains the acceleration relation:

$$\dot{\boldsymbol{v}}_i = \boldsymbol{J}_{iO} \dot{\boldsymbol{v}}_O + \dot{\boldsymbol{J}}_{iO} \boldsymbol{v}_O, \ i \in \mathscr{K} \tag{8.5}$$

which will be used in the subsequent analysis.

8.2.2 Dynamics

8.2.2.1 UVMS Dynamics

The dynamics of a UVMS after straightforward algebraic manipulations can be written as [10]:

$$\boldsymbol{M}_{q_i}(\boldsymbol{q}_i)\dot{\boldsymbol{\zeta}}_i + \boldsymbol{C}_{q_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{\zeta}_i + \boldsymbol{D}_{q_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{\zeta}_i + \boldsymbol{g}_{q_i}(\boldsymbol{q}_i) + \boldsymbol{d}_{q_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t) = \boldsymbol{\tau}_i + \boldsymbol{J}_i{}^{\top}\boldsymbol{\lambda}_i \quad (8.6)$$

for $i \in \mathscr{K}$, where $\boldsymbol{\lambda}_i$ is the vector of *measured* interaction forces and torques exerted at the end-effector by the object, $\boldsymbol{\tau}_i$ denotes the vector of control inputs (forces and torques), $\boldsymbol{M}_{q_i}(\boldsymbol{q}_i)$ is the inertial matrix, $\boldsymbol{C}_{q_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)$ represents coriolis and centrifugal terms, $\boldsymbol{D}_{q_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)$ models dissipative effects, $\boldsymbol{g}_i(\boldsymbol{q}_i)$ encapsulates the gravity and buoyancy effects and $\boldsymbol{d}_{q_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t)$ is a bounded vector representing unmodeled friction, uncertainties and external disturbances. In view of (8.2) we have:

$$\dot{\boldsymbol{v}}_i = \boldsymbol{J}_i(\boldsymbol{q}_i)\dot{\boldsymbol{\zeta}}_i + \boldsymbol{J}_i^d(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{\zeta}_i, \ i \in \mathscr{K} \quad (8.7)$$

where $\boldsymbol{J}_i^d(\boldsymbol{\zeta}_i, \boldsymbol{q}_i) \in \mathbb{R}^{6\times(6+n)}$ represents the Jacobian derivative function, with $\boldsymbol{J}_i^d(\boldsymbol{\zeta}_i, \boldsymbol{q}_i) \triangleq \dot{\boldsymbol{J}}_i(\boldsymbol{q}_i)$. Then, by employing the differential kinematics as well as (8.7), we obtain from (8.6) the transformed task space dynamics [252]:

$$\boldsymbol{M}_{v_i}(\boldsymbol{q}_i)\dot{\boldsymbol{v}}_i + \boldsymbol{C}_{v_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{v}_i + \boldsymbol{D}_{v_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{v}_i + \boldsymbol{g}_{v_i}(\boldsymbol{q}_i) + \boldsymbol{d}_{v_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t) = \boldsymbol{u}_i + \boldsymbol{\lambda}_i \quad (8.8)$$

with the corresponding task space terms $\boldsymbol{M}_{v_i} \in \mathbb{R}^{6\times6}$, $\boldsymbol{C}_{v_i} \in \mathbb{R}^{6\times6}$, $\boldsymbol{D}_{v_i} \in \mathbb{R}^{6\times6}$, $\boldsymbol{g}_{v_i} \in \mathbb{R}^6$, $\boldsymbol{d}_{v_i} \in \mathbb{R}^6$ and $\boldsymbol{u}_i$ to be the vector of task space generalized forces/torques. It is worth noting that the vector of control inputs $\boldsymbol{\tau}_i, \ i \in \mathscr{K}$ can be related to the task space wrench $\boldsymbol{u}_i \in \mathbb{R}^6, \ i \in \mathscr{K}$ via:

$$\boldsymbol{\tau}_i = \boldsymbol{J}_i^{\top}(\boldsymbol{q}_i)\boldsymbol{u}_i + (\boldsymbol{I}_{6+n} - \boldsymbol{J}_i^{\top}(\boldsymbol{q}_i)\tilde{\boldsymbol{J}}_i^{\top}(\boldsymbol{q}_i))\boldsymbol{\tau}_{i0} \quad (8.9)$$

where, $\tilde{\boldsymbol{J}}_i^{\top}(\boldsymbol{q}_i)$ is the generalized pseudo-inverse of $\boldsymbol{J}_i$ [252] and the vector $\boldsymbol{\tau}_{i0}$ does not contribute to the end effector's wrench $\boldsymbol{u}_i$ (i.e., they belong to the null space of the Jacobian $\boldsymbol{J}_i$) and can be regulated independently to achieve secondary tasks (e.g., maintaining manipulator's joint limits, increasing the manipulability)[3]. Invoking the kinematic relation (8.3)-(8.5), we may express the aforementioned dynamics (8.8) with

[3] For more details on task priority based control and redundancy resolution for UVMSs the reader is referred to [145] and [151].

respect to the object's coordinates as follows:

$$\boldsymbol{M}_i(\boldsymbol{q}_i)\dot{\boldsymbol{v}}_O + \boldsymbol{C}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{v}_O + \boldsymbol{D}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{v}_O + \boldsymbol{g}_i(\boldsymbol{q}_i) + \boldsymbol{d}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t) = \boldsymbol{J}_{iO}^{\top}\boldsymbol{u}_i + \boldsymbol{J}_{iO}^{\top}\boldsymbol{\lambda}_i \tag{8.10}$$

where:

$$\begin{aligned}
\boldsymbol{M}_i(\boldsymbol{q}_i) &= \boldsymbol{J}_{iO}^{\top}\boldsymbol{M}_{v_i}(\boldsymbol{q}_i)\boldsymbol{J}_{iO} \\
\boldsymbol{C}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i) &= \boldsymbol{J}_{iO}^{\top}\left[\boldsymbol{C}_{v_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{J}_{iO} + \boldsymbol{M}_{v_i}(\boldsymbol{q}_i)\dot{\boldsymbol{J}}_{iO}\right] \\
\boldsymbol{D}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i) &= \boldsymbol{J}_{iO}^{\top}\boldsymbol{D}_{v_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\boldsymbol{J}_{iO} \\
\boldsymbol{g}_i(\boldsymbol{q}_i) &= \boldsymbol{J}_{iO}^{\top}\boldsymbol{g}_{v_i}(\boldsymbol{q}_i) \\
\boldsymbol{d}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t) &= \boldsymbol{J}_{iO}^{\top}\boldsymbol{d}_{v_i}(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t)
\end{aligned}$$

Now, the following common properties will be employed in the analysis:

Property 8.1. *The matrix $\boldsymbol{M}_i(\boldsymbol{q}_i)$, $i \in \mathscr{K}$ is positive definite and the matrix $\dot{\boldsymbol{M}}_i(\boldsymbol{q}_i) - 2\boldsymbol{C}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)$, $i \in \mathscr{K}$ is skew-symmetric. We have that a quadratic form of a skew-symmetric matrix is always equal to 0. Hence, for the matrices $\dot{\boldsymbol{M}}_i(\boldsymbol{q}_i) - 2\boldsymbol{C}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)$, the following holds [14, 252, 253]:*

$$\boldsymbol{s}^{\top}\left[\dot{\boldsymbol{M}}_i(\boldsymbol{q}_i) - 2\boldsymbol{C}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i)\right]\boldsymbol{s} = 0, \forall \boldsymbol{s} \in \mathbb{R}^6$$

Property 8.2. *The uncertainty of the UVMS model appears linearly in the dynamics (8.10), in terms of an unknown but constant parameter vector $\boldsymbol{\theta}_i \in \mathbb{R}^{q_i}$, $i \in \mathscr{K}$ in the following way[253, 254]:*

$$\boldsymbol{M}_i(\boldsymbol{a}_i)\boldsymbol{d}_i + \boldsymbol{C}_i(\boldsymbol{a}_i, \boldsymbol{b}_i)\boldsymbol{c}_i + \boldsymbol{D}_i(\boldsymbol{a}_i, \boldsymbol{b}_i)\boldsymbol{c}_i + \boldsymbol{g}_i(\boldsymbol{a}_i) = \boldsymbol{\Omega}_i(\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{c}_i, \boldsymbol{d}_i)\boldsymbol{\theta}_i$$

for $i \in \mathscr{K}$, where $\boldsymbol{\Omega}_i(\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{c}_i, \boldsymbol{d}_i) \in \mathbb{R}^{6\times q_i}$, $i \in \mathscr{K}$ is a regressor matrix of known functions of $\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{c}_i, \boldsymbol{d}_i \in \mathbb{R}^6$ independent of $\boldsymbol{\theta}_i$.

Now, we introduce the following assumption regarding the unmodeled dynamics/external disturbances.

Assumption 8.1. There exists positive, finite unknown constant $\boldsymbol{\theta}_{d,i} \in \mathbb{R}^{q_i}$, $i \in \mathscr{K}$ and known bounded function $\boldsymbol{\Delta}_i \in \mathbb{R}^{6\times q_i}$, $i \in \mathscr{K}$, such that

$$\boldsymbol{d}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t) = \boldsymbol{\Delta}_i(\boldsymbol{\zeta}_i, \boldsymbol{q}_i, t)\boldsymbol{\theta}_{d,i},\ i \in \mathscr{K}$$

8.2.2.2 Object Dynamic

Without any loss of generality, we consider the following second order dynamic for the object, which can be derived based on the Newton-Euler formulations:

$$\dot{\boldsymbol{x}}_O = \boldsymbol{J}_O(\boldsymbol{\eta}_{2,O})\boldsymbol{v}_O \tag{8.11}$$

$$\boldsymbol{M}_O(\boldsymbol{x}_O)\dot{\boldsymbol{v}}_O + \boldsymbol{C}_O(\dot{\boldsymbol{x}}_O, \boldsymbol{x}_O)\boldsymbol{v}_O + \boldsymbol{D}_O(\dot{\boldsymbol{x}}_O, \boldsymbol{x}_O)\boldsymbol{v}_O + \boldsymbol{g}_O = \boldsymbol{\lambda}_O + \boldsymbol{\lambda}_e \tag{8.12}$$

where $\boldsymbol{M}_O(\boldsymbol{x}_O)$ is the positive definite inertia matrix, $\boldsymbol{C}_O(\dot{\boldsymbol{x}}_O, \boldsymbol{x}_O)$ is the Coriolis matrix, $\boldsymbol{g}_O$ is the vector of gravity and buoyancy effects, $\boldsymbol{D}_O(\dot{\boldsymbol{x}}_O, \boldsymbol{x}_O)$ models dissipative effects, $\boldsymbol{\lambda}_O$ is the vector of generalized forces acting on the object's center of mass and $\boldsymbol{\lambda}_e$ is a vector representing uncertainties and external disturbances. Moreover, $\boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O})$ is the object representation Jacobian $\boldsymbol{J}_O(\boldsymbol{\eta}_{2,O}) = \text{diag}\{\boldsymbol{I}_3, \boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O})\}$:

$$\boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O}) = \begin{bmatrix} 1 & \sin(\phi_O)\tan(\theta_O) & cos(\phi_O)\tan(\theta_O) \\ 0 & \cos(\phi_O) & -\sin(\theta_O) \\ 0 & \frac{\sin(\phi_O)}{\cos(\theta_O)} & \frac{cos(\phi_O)}{\cos(\theta_O)} \end{bmatrix}, \tag{8.13}$$

Moreover, The kineto-statics duality along with the grasp rigidity suggest that the force $\boldsymbol{\lambda}_O$ acting on the object's center of mass and the generalized forces $\boldsymbol{\lambda}_i,\ i \in \mathscr{K}$, exerted by the UVMSs at the grasping points, are related through:

$$\boldsymbol{\lambda}_O = \boldsymbol{G}^\top \boldsymbol{\lambda} \tag{8.14}$$

where:

$$\boldsymbol{G} = \left[[\boldsymbol{J}_{LO}]^\top, [\boldsymbol{J}_{F_1O}]^\top, \ldots, [\boldsymbol{J}_{F_NO}]^\top \right]^\top \in \mathbb{R}^{6(N+1)\times 6} \tag{8.15}$$

is the full column-rank grasp matrix and $\boldsymbol{\lambda} = [\boldsymbol{\lambda_L}^\top, \boldsymbol{\lambda_{F_1}}^\top, \ldots, \boldsymbol{\lambda_{F_N}}^\top]^\top$ is the vector of overall interaction forces and torques.

Remark 8.1. Wrenches that lie on the null space of the grasp matrix $\boldsymbol{G}^\top$ do not contribute to the object dynamics. Therefore, we may incorporate in the control scheme an extra component $\boldsymbol{\lambda}_{int,i} = (\boldsymbol{I} - (\boldsymbol{G}^\top)^\# \boldsymbol{G}^\top)\boldsymbol{\lambda}^d_{int},\ i \in \mathscr{K}$, that belongs to the null space of $\boldsymbol{G}^\top$, in order to regulate the steady state internal forces, where $(\boldsymbol{G}^\top)^\#$ denotes the generalized inverse of $\boldsymbol{G}^\top$. Notice that owing to the rigid grasp, $l_i,\ i \in \mathscr{K}$ remain constant. Thus, since $l_i,\ i \in \mathscr{K}$ are considered known to the team of UVMSs[4], if $\boldsymbol{\lambda}^d_{int}$ is chosen constant, no communication is needed during task execution in order to compute $\boldsymbol{G}^\top,\ (\boldsymbol{G}^\top)^\#$ and $\boldsymbol{\lambda}_{int,i}$.

[4]This can be achieved by using the acoustic modems before beginning the task execution.

8.2.3 Description of the Workspace

Consider the team of $N+1$ UVMSs operating in a bounded workspace $\mathscr{W} \subseteq \mathbb{R}^3$ with boundary $\partial\mathscr{W}$. The object of interest is a rigid body which is required to be transported cooperatively by the robot team from an initial to a goal position. Without any loss of the generality, the obstacles, the robots as well as the workspace are all modeled by spheres (i.e., we adopt the spherical world representation [185]). However, the proposed control strategy could be extended to more general and complex geometries following the analysis in [185]. In this spirit, let $\mathscr{B}(\boldsymbol{x}_O, r_0)$ be a closed ball that covers the volume of the object and has radius r_0. We also define the closed balls $\mathscr{B}(\boldsymbol{x}_i, \bar{r}), i \in \mathscr{K}$, centered at the end-effector of each UVMS that cover the robot volume for all possible configurations. Notice that the value of $\bar{r}$ can be calculated easily for each UVMS based solely on its own design parameters. We also assume that the distance among the grasping points on the given object is at least $2\bar{r}$. In particular, the distance $2\bar{r}$ denotes the minimum allowed distance at which two bounding spheres $\mathscr{B}(\boldsymbol{x}_i, \bar{r})$ and $\mathscr{B}(\boldsymbol{x}_j, \bar{r})$ $i, j \in \mathscr{K}, i \neq j$ do not collide (see Fig. 8.1). Furthermore, we define a ball area $\mathscr{B}(\boldsymbol{x}_O, R)$ located at

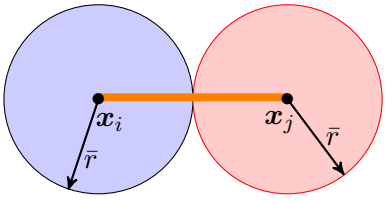


FIGURE 8.1: Graphical representation of the minimum allowed distance $\bar{r}$.

$\boldsymbol{x}_O$ with radius $R = \bar{r} + r_o$ that includes the whole volume of the robotic team and the object (see Fig. 8.2). Finally, the $\mathscr{M}$ static obstacles within the workspace are defined as closed spheres described by $\pi_m = \mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m}),\ m \in \{1, \ldots, \mathscr{M}\}$, where $\boldsymbol{p}_{\pi_m} \in \mathbb{R}^3$ is the center and the $r_{\pi_m} > 0$ the radius of the obstacle π_m. Obviously, the ultimate goal of the proposed cooperative control strategy is to transport the object from the initial configuration to the desired one, without colliding with the obstacles and the boundary of workspace. Additionally, based on the property of spherical world [185], for each pair of obstacles $m, m' \in \{1, \ldots, \mathscr{M}\}$ the following inequality holds:

$$||\boldsymbol{p}_{\pi_m} - \boldsymbol{p}_{\pi'_m}|| > 2R + r_{\pi_m} + r_{\pi'_m}$$

which intuitively means that the obstacles m and m' are disjoint in a such a way that the whole team of UVMSs including the object can pass through the free space between them. Therefore, there exists a feasible trajectory $\boldsymbol{x}_O(t)$ for the whole team that connects the initial configuration $\boldsymbol{x}_O(t_0)$ with $\boldsymbol{x}_O^d$ such as:

$$\mathscr{B}(\boldsymbol{x}_O(t), R) \cap \{\mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m}) \cup \partial\mathscr{W}\} = \emptyset,\ \forall m \in \{1, \ldots, \mathscr{M}\} \tag{8.16}$$

Hence, the problem that we aim to solve in this chapter is stated as follows:

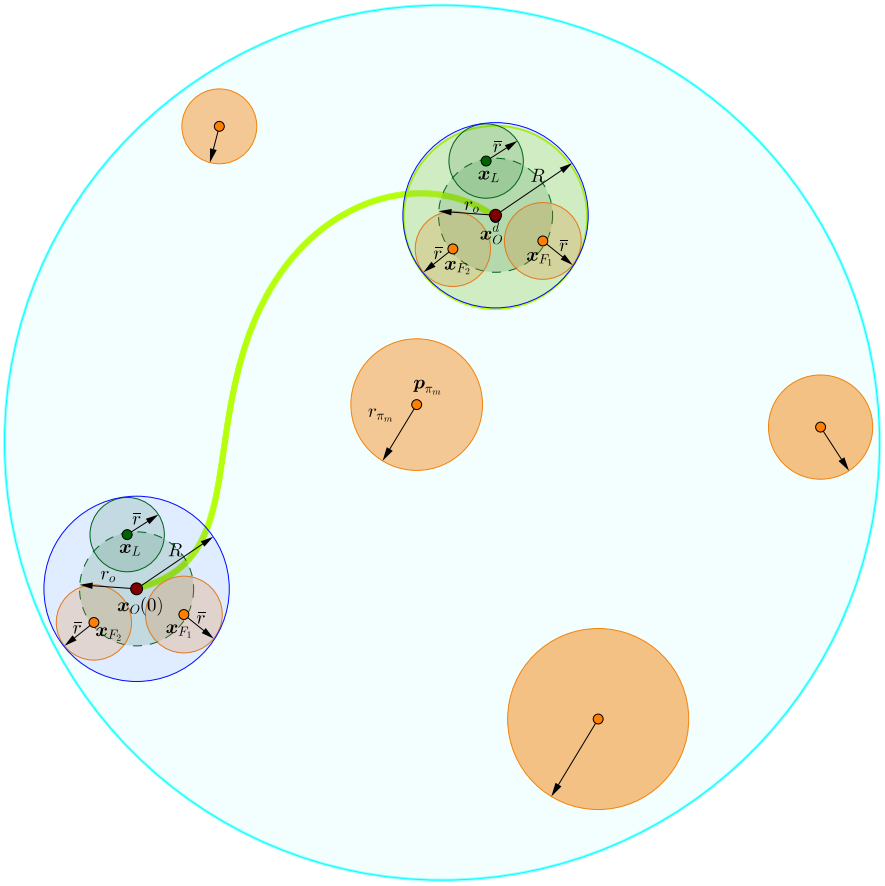


FIGURE 8.2: Graphical representation of a feasible trajectory of the team of UVMS carrying object from the initial position $\boldsymbol{x}_O(t_0)$ to the desired position $\boldsymbol{x}_O^d$. The boundary of workspace $\partial\mathscr{W}$ is illustrated in cyan. Red circles indicate the obstacles within the workspace $\mathscr{W}$. A feasible trajectory of the whole team is depicted in green.

Problem: Given $N+1$ UVMSs operating in a constrained workspace $\mathscr{W}$, design distributed control protocols $\boldsymbol{u}_i, \quad \in \mathscr{K}$ that navigate safely the whole robotic team to the desired configuration without colliding with the obstacles and the boundary of the workspace, while satisfying the following specifications:

1. Impose no strict requirements regarding the underwater communication bandwidth;

2. Enforce robustness against the parametric uncertainty of the UVMS dynamic model.

8.3 Control Methodology

We assume that the leading UVMS is aware of both the desired configuration of the object as well as of the obstacles position in the workspace. Thus, its control objective is to navigate the overall formation towards the goal configuration while avoiding collisions with the static obstacles that lie within the workspace. Towards this direction and in view of (8.16), we assume that there is a feasible trajectory within the workspace which is known only for the leader. On the other hand, the followers are not aware of the object's desired configuration. However, even though explicit inter-robot communication is not permitted, the followers will estimate the object's desired trajectory profile via their own state measurements (sensor fusion of locally onboard navigation system sensors, e.g., DVL, IMU, USBL). Towards this direction, acceleration residuals owing to the lack of acceleration measurements for the object will be compensated by adopting a robust prescribed performance estimator that guarantees ultimate boundedness of the estimation error with predefined transient and steady state specifications. Finally, an adaptive control scheme will be designed to achieve the asymptotic tracking of the estimated trajectory profile, thus increasing greatly the robustness of the overall control scheme and avoiding high interaction forces among the object and the robots.

Remark 8.2. The desired/feasible object trajectory within the workspace $\mathscr{W}$ can be generated based on the Navigation Functions concept originally proposed by Rimon and Koditschek in [185] as follows:

$$\phi_O(\boldsymbol{x}_O;\boldsymbol{x}_O^d)=\frac{\gamma(\boldsymbol{x}_O-\boldsymbol{x}_O^d)}{[\gamma^k(\boldsymbol{x}_O-\boldsymbol{x}_O^d)+\beta(\boldsymbol{x}_O)]^{\frac{1}{k}}} \tag{8.17}$$

where $\phi_O:\mathscr{W}-\bigcap_{m=1}^{\mathscr{M}}\mathscr{B}(\boldsymbol{p}_{\pi_m},r_{\pi_m})\longrightarrow[0,1)$ denotes the potential that derives a safe motion vector field within the free space $\mathscr{W}-\bigcap_{m=1}^{\mathscr{M}}\mathscr{B}(\boldsymbol{p}_{\pi_m},r_{\pi_m})$. Moreover, $k>1$ is a design constant, $\gamma(\boldsymbol{x}_O-\boldsymbol{x}_O^d)>0$ with $\gamma(\boldsymbol{0})=0$ represents the attractive potential field to the goal configuration $\boldsymbol{x}_O^d$ and $\beta(\boldsymbol{x}_O)>0$ with:

$$\lim_{\boldsymbol{x}_O\to\left\{\substack{\text{Boundary}\\\text{Obstacles}}\right.}\beta(\boldsymbol{x}_O)=0$$

represents the repulsive potential field by the workspace boundary and the obstacle regions. In that respect, it was proven in [185] that $\phi_O(\boldsymbol{x}_O,\boldsymbol{x}_O^d)$ has a global minimum at $\boldsymbol{x}_O^d$ and no other local minima for sufficiently large k. Thus, a feasible path that leads from any initial obstacle-free configuration[5] to the desired configuration might be generated by following the negated gradient of $\phi_O(\boldsymbol{x}_O,\boldsymbol{x}_O^d)$. Consequently, the desired

[5]Except from a set of measure zero[185].

velocity profile at leader's side is designed as follows:

$$\boldsymbol{v}_{O_L}^d(t) = -K_{NF}\boldsymbol{J}_O^{-1}(\boldsymbol{\eta}_{2,O})\nabla_{\boldsymbol{x}_O}\phi_O(\boldsymbol{x}_O(t), \boldsymbol{x}_O^d) \tag{8.18}$$

where $K_{NF} > 0$ is a positive gain. Moreover, given the initial configuration, the leading UVMS may easily calculate the desired trajectory and velocity profile denoted by $\boldsymbol{x}_{O_L}^d(t)$ and $\boldsymbol{v}_{O_L}^d(t)$ respectively, by propagating the model (8.11).

via a propagation procedure based on (8.17) and (8.18) can calculates a map of desired trajectory and velocity of the object denoted by $\boldsymbol{x}_{O_L}^d(t)$ and $\boldsymbol{v}_{O_L}^d(t)$ respectively. Notice that the generation of the feasible trajectory of the object is out of scops of this work and we assume that the desired trajectory of the object is available in previous for the leading UVMS.

8.3.1 Control Design

In the sequel, we propose an decentralized control scheme that guarantees the asymptotic stabilization of the object to the goal configuration $\boldsymbol{x}_O^d$. Before proceeding the analysis, we introduce the load sharing coefficients $c_i,\ i \in \mathscr{K}$ that are subject to the following design constraints:

$$c_i \in (0,1), \forall i \in \mathscr{K},\ \text{and} \sum_{i\in\mathscr{K}} c_i = 1 \tag{8.19}$$

Thus, before losing any generality, for simplify the analysis we set:

$$c_i = \frac{1}{N+1}, \qquad i \in \mathscr{K} \tag{8.20}$$

which satisfy the constraints of (8.19). In view of the object dynamics (8.12), it can be concluded that the vector of external disturbances $\boldsymbol{\lambda}_e$ is unknown. Thus, in order to design the impedance control scheme, each UVMS must estimate the aforementioned vector in a distributed way (since there is not explicit communication between UVMSs). Moreover, based on the object dynamics (8.12), the vector of external disturbances is impossible to be estimated in a decentralized way by relying only on local measurements (i.e., local force torque measurements at UVMS's end effector), since it depends on the applying force from all member of the UVMS teams on the object. Therefore, an online estimation method based on the object momentum concept [255] is designed in the sequel. First, in view of the load coefficients (8.20) and the (8.14), the object dynamic

of (8.12) can be rewritten as:

$$\sum_{i\in\mathscr{K}}\left\{\boldsymbol{M}_{O_i}(\boldsymbol{x}_O)\dot{\boldsymbol{v}}_O+\boldsymbol{C}_{O_i}(\dot{\boldsymbol{x}}_O,\boldsymbol{x}_O)\boldsymbol{v}_O+\boldsymbol{D}_{O_i}(\dot{\boldsymbol{x}}_O,\boldsymbol{x}_O)\boldsymbol{v}_O+\boldsymbol{g}_{O_i}\right\}=\sum_{i\in\mathscr{K}}\boldsymbol{J}_{iO}^{\top}\boldsymbol{\lambda}_i+\sum_{i\in\mathscr{K}}\boldsymbol{\lambda}_{e_i} \tag{8.21}$$

where: $\boldsymbol{M}_{O_i}=c_i\boldsymbol{M}_O$, $\boldsymbol{C}_{O_i}=c_i\boldsymbol{C}_O$, $\boldsymbol{D}_{O_i}=c_i\boldsymbol{D}_O$, $\boldsymbol{g}_{O_i}=c_i\boldsymbol{g}_O$ and $\boldsymbol{\lambda}_{e_i}=c_i\boldsymbol{\lambda}_e$. In order to estimate locally the $\boldsymbol{\lambda}_{e_i}$ for the UVMS i, $i\in\mathscr{K}$ we define the object momentum equivalent momentum [255]: $\boldsymbol{\mu}_i=\boldsymbol{M}_{O_i}\boldsymbol{v}_O$ and the vector $\boldsymbol{\zeta}_i(t)\in\mathbb{R}^6$ as:

$$\boldsymbol{\zeta}_i(t)=\boldsymbol{K}_\mu\left(\boldsymbol{\mu}_i(t)+\int_{t_0}^{t}\Big(\boldsymbol{C}_{O_i}\boldsymbol{v}_O+\boldsymbol{D}_{O_i}\boldsymbol{v}_O+\boldsymbol{g}_{O_i}-\boldsymbol{\zeta}_i(d\tau)\Big)d\tau\right) \tag{8.22}$$

whose time derivative is given by:

$$\dot{\boldsymbol{\zeta}}_i(t)=-\boldsymbol{K}_\mu\boldsymbol{\zeta}_i(t)+c_i\boldsymbol{K}_\mu\Big(\sum_{i\in\mathscr{K}}\boldsymbol{J}_{iO}^{\top}\boldsymbol{\lambda}_i+\boldsymbol{\lambda}_e\Big) \tag{8.23}$$

where $\boldsymbol{K}_\mu$ is a positive definite matrix gain. Notice that the for a properly large matrix $\boldsymbol{K}_\mu$, we obtain:

$$\boldsymbol{\zeta}_i(t)\approx c_i\Big(\sum_{i\in\mathscr{K}}\boldsymbol{J}_{iO}^{\top}\boldsymbol{\lambda}_i+\boldsymbol{\lambda}_e\Big) \tag{8.24}$$

which intuitively means that the $\boldsymbol{\zeta}_i(t)$ represents the effect of overall external forces exerted on the object (i.e., external disturbances and the forces exerted by all the UVMS team on the object). Consequently, an estimation of $\boldsymbol{\lambda}_{e_i}=c_i\boldsymbol{\lambda}_e$ can be given as:

$$\boldsymbol{\lambda}_{e_i}\approx\boldsymbol{\zeta}_i(t)-\boldsymbol{J}_{iO}^{\top}\boldsymbol{\lambda}_i,\ i\in\mathscr{K} \tag{8.25}$$

Now let us assume that each UVMS is expected to exerts the following desired force/-torque on the object:

$$\boldsymbol{\lambda}_i^d=\boldsymbol{\lambda}_{int,i}^d-\boldsymbol{J}_{iO}^{-\top}(\boldsymbol{M}_{O_i}\boldsymbol{y}_i^{cmd}+\boldsymbol{C}_{O_i}\boldsymbol{v}_O+\boldsymbol{D}_{O_i}\boldsymbol{v}_O+\boldsymbol{g}_{O_i}-\boldsymbol{\lambda}_{e_i}) \tag{8.26}$$

where $\boldsymbol{\lambda}_{int,i}^d$ is the desired inertial forces (See Remark-8.1) and the $\boldsymbol{y}_i^{cmd}$ is a pre-designed input given by:

$$\boldsymbol{y}_i^{cmd}=\dot{\boldsymbol{v}}_{O_i}^d+\boldsymbol{M}_{d_O}^{-1}\Big[-\boldsymbol{D}_{d_O}\tilde{\boldsymbol{v}}_{O_i}-\boldsymbol{K}_{d_O}\tilde{\boldsymbol{e}}_{O_i}\Big] \tag{8.27}$$

where $\boldsymbol{M}_{d_O}$, $\boldsymbol{D}_{d_O}$ and $\boldsymbol{K}_{d_O}$ are the desired inertia, damping and stiffness matrices for the object dynamics, $\tilde{\boldsymbol{v}}_{O_i}(t)=\boldsymbol{v}_O-\boldsymbol{v}_{O_i}^d$ denotes the velocity error and $\tilde{\boldsymbol{e}}_{O_i}$ is the object

pose error, defined as:

$$\tilde{\boldsymbol{e}}_{O_i} = \begin{bmatrix} \boldsymbol{\eta}_{1,O} - \boldsymbol{\eta}^d_{1,O_i} \\ \tilde{\boldsymbol{\epsilon}}_{O_i} \end{bmatrix} \tag{8.28}$$

where

$$\tilde{\boldsymbol{\epsilon}}_{O_i} = \frac{1}{2}\Big(\boldsymbol{n}_O \times \boldsymbol{n}^d_{O_i} + \boldsymbol{o}_O \times \boldsymbol{o}^d_{O_i} + \boldsymbol{\alpha}_O \times \boldsymbol{\alpha}^d_{O_i}\Big) \in \mathbb{R}^3 \tag{8.29}$$

is the orientation error expressed in the outer product formulation [244]. In view of (8.21), it can be concluded that if all robots cooperatively apply the desired wrench vector (8.26) to the object, then

$$\boldsymbol{M}_{d_O}\dot{\tilde{\boldsymbol{v}}}_{O_i} + \boldsymbol{D}_{d_O}\tilde{\boldsymbol{v}}_{O_i} + \boldsymbol{K}_{d_O}\tilde{\boldsymbol{e}}_{O_i} = \boldsymbol{0} \tag{8.30}$$

which intuitively means that the aforementioned selection of $\boldsymbol{\lambda}^d_i$ cancels the object's nonlinearities, ensures adequate internal forces via $\boldsymbol{\lambda}^d_{int,i}$ and achieves the desired dynamics of the object. Thus, the control objective for each UVMS $i \in \mathscr{K}$ is to enforce $\lim_{t\to\infty} \boldsymbol{w}_i(t) = 0$, where the error signal $\boldsymbol{w}(t)$ is constructed as:

$$\boldsymbol{w}_i(t) = \boldsymbol{M}_d\dot{\tilde{\boldsymbol{v}}}_{O_i} + \boldsymbol{D}_d\tilde{\boldsymbol{v}}_{O_i} + \boldsymbol{K}_d\tilde{\boldsymbol{e}}_{O_i} - \boldsymbol{J}^\top_{iO}\boldsymbol{\lambda}^d_i,\ i \in \mathscr{K} \tag{8.31}$$

where $\boldsymbol{M}_d$, $\boldsymbol{D}_d$ and $\boldsymbol{K}_d$ are the desired inertia, damping and stiffness matrices for the robot dynamics. Thus, we get an augmented impedance error:

$$\tilde{\boldsymbol{w}}_i = \boldsymbol{K}_f\boldsymbol{w}_i = \dot{\tilde{\boldsymbol{v}}}_{O_i} + \boldsymbol{K}_g\tilde{\boldsymbol{v}}_{O_i} + \boldsymbol{K}_p\tilde{\boldsymbol{e}}_{O_i} - \boldsymbol{K}_f\boldsymbol{J}^\top_{iO}\boldsymbol{\lambda}^d_i \tag{8.32}$$

where $\boldsymbol{K}_f = \boldsymbol{M}_d^{-1}$, $\boldsymbol{K}_g = \boldsymbol{K}_f\boldsymbol{D}_d$, and $\boldsymbol{K}_p = \boldsymbol{K}_f\boldsymbol{K}_d$. We also choose two positive-definite matrices $\boldsymbol{F}$ and $\boldsymbol{Y}$ such that:

$$\boldsymbol{F} + \boldsymbol{Y} = \boldsymbol{K}_g$$
$$\dot{\boldsymbol{F}} + \boldsymbol{Y}\boldsymbol{F} = \boldsymbol{K}_p$$

and define the filtered force/torque measurement:

$$\dot{\boldsymbol{f}}_i + \boldsymbol{Y}\boldsymbol{f}_i = \boldsymbol{K}_f\boldsymbol{J}^\top_{iO}\boldsymbol{\lambda}^d_i,\ i \in \mathscr{K}. \tag{8.33}$$

Thus, we may rewrite (8.32) as:

$$\tilde{\boldsymbol{w}}_i = \dot{\tilde{\boldsymbol{v}}}_{O_i} + (\boldsymbol{F} + \boldsymbol{Y})\tilde{\boldsymbol{v}}_{O_i} + (\dot{\boldsymbol{F}} + \boldsymbol{Y}\boldsymbol{F})\tilde{\boldsymbol{e}}_{O_i} - \dot{\boldsymbol{f}}_i - \boldsymbol{Y}\boldsymbol{f}_i. \tag{8.34}$$

Now we define the auxiliary variables $\boldsymbol{z}_i,\ i \in \mathscr{K}$ as:

$$\boldsymbol{z}_i = \tilde{\boldsymbol{v}}_{O_i} + \dot{\boldsymbol{F}}\check{\boldsymbol{e}}_{O_i} - \boldsymbol{f}_i,\ i \in \mathscr{K}. \tag{8.35}$$

Hence, the augmented impedance error becomes:

$$\tilde{\boldsymbol{w}}_i = \dot{\boldsymbol{z}}_i + \boldsymbol{Y}\boldsymbol{z}_i,\ i \in \mathscr{K} \tag{8.36}$$

which represents a stable low pass filter. Therefore, if we achieve $\lim_{t\to\infty} \boldsymbol{z}_i(t) = 0$, then the initial control objective is readily met, i.e, $\lim_{t\to\infty} \boldsymbol{w}_i(t) = 0$. In this respect, let us define the augmented state variable:

$$\boldsymbol{v}^r_{O_i} = \boldsymbol{v}^d_{O_i} - \boldsymbol{F}\check{\boldsymbol{e}}_{O_i} + \boldsymbol{f}_i,\ i \in \mathscr{K} \tag{8.37}$$

Thus, (8.35) and (8.37) immediately result in:

$$\boldsymbol{z}_i = \boldsymbol{v}_O - \boldsymbol{v}^r_{O_i},\ i \in \mathscr{K} \tag{8.38}$$

from which the dynamics (8.10) becomes:

$$\boldsymbol{M}_i\dot{\boldsymbol{z}}_i + \boldsymbol{C}_i\boldsymbol{z}_i + \boldsymbol{D}_i\boldsymbol{z}_i = \boldsymbol{J}^\top_{iO}\boldsymbol{u}_i + \boldsymbol{J}^\top_{iO}\boldsymbol{\lambda}_i - \Big[\boldsymbol{M}_i\dot{\boldsymbol{v}}^r_{O_i} + \boldsymbol{C}_i\boldsymbol{v}^r_{O_i} + \boldsymbol{D}_i\boldsymbol{v}^r_{O_i} + \boldsymbol{g}_i + \boldsymbol{d}_i\Big].$$

Invoking Property 2 and Assumption 8.1, we arrive at the open loop dynamics:

$$\boldsymbol{M}_i\dot{\boldsymbol{z}}_i+\boldsymbol{C}_i\boldsymbol{z}_i+\boldsymbol{D}_i\boldsymbol{z}_i=\boldsymbol{J}^\top_{iO}\boldsymbol{u}_i+\boldsymbol{J}^\top_{iO}\boldsymbol{\lambda}_i-\boldsymbol{\Delta}_i(\boldsymbol{\zeta}_i,\boldsymbol{q}_i,t)\boldsymbol{\theta}_{d,i}-\boldsymbol{\Omega}_i(\boldsymbol{q}_i,\boldsymbol{\zeta}_i,\boldsymbol{v}^r_{O_i},\dot{\boldsymbol{v}}^r_{O_i})\boldsymbol{\theta}_i,\ i \in \mathscr{K}. \tag{8.39}$$

Therefore, we design the following impedance control scheme:

$$\boldsymbol{u}_i = -\boldsymbol{\lambda}_i + \boldsymbol{J}^{-\top}_{iO}\Big[\boldsymbol{\Omega}_i(\boldsymbol{q}_i,\boldsymbol{\zeta}_i,\boldsymbol{v}^r_{O_i},\dot{\boldsymbol{v}}^r_{O_i})\hat{\boldsymbol{\theta}}_i + \boldsymbol{\Delta}_i(\boldsymbol{\zeta}_i,\boldsymbol{q}_i,t)\hat{\boldsymbol{\theta}}_{d,i} - \boldsymbol{K}\boldsymbol{z}_i\Big],\ i \in \mathscr{K} \tag{8.40}$$

where $\boldsymbol{K} > 0$ is a positive definite gain matrix and $\hat{\boldsymbol{\theta}}_i$, $\hat{\boldsymbol{\theta}}_{d,i}$ denote the estimates of the unknown parameters $\boldsymbol{\theta}_i$, $\boldsymbol{\theta}_{d,i}$ respectively, provided by the update laws:

$$\dot{\hat{\boldsymbol{\theta}}}_i = -\boldsymbol{\Gamma}_i\boldsymbol{\Omega}_i(\boldsymbol{q}_i,\boldsymbol{\zeta}_i,\boldsymbol{v}^r_{O_i},\dot{\boldsymbol{v}}^r_{O_i})\boldsymbol{z}_i,\ \boldsymbol{\Gamma}_i > 0 \tag{8.41}$$

$$\dot{\hat{\boldsymbol{\theta}}}_{d,i} = -\boldsymbol{\Gamma}_{d_i}\boldsymbol{\Delta}_i(\boldsymbol{\zeta}_i,\boldsymbol{q}_i,t)\boldsymbol{z}_i,\ \boldsymbol{\Gamma}_{d_i} > 0 \tag{8.42}$$

with $\boldsymbol{\Gamma}_i$, $\boldsymbol{\Gamma}_{d_i}$ positive diagonal gain matrices.

Theorem 8.1. *Consider $N+1$ UVMSs operating in a constrained workspace $\mathscr{W}$ with dynamics given as (8.10) that obey Properties 8.1–8.2, grasping rigidly a common object.*

The control scheme for each UVMS i, $i \in \mathcal{K}$ given in (8.40) with adaptive laws (8.41)-(8.42) guarantees $\lim_{t\to\infty} \boldsymbol{w}_i(t) = 0$ and the boundedness os all signals in the closed loop system.

Proof. Consider the following Lyapunov function candidate:

$$\boldsymbol{V} = \sum_{i\in\mathcal{K}} \frac{1}{2}\boldsymbol{z}_i^\top \boldsymbol{M}_i \boldsymbol{z}_i + \sum_{i\in\mathcal{K}} \frac{1}{2}\tilde{\boldsymbol{\theta}}_i^\top \boldsymbol{\Gamma}_i^{-1}\tilde{\boldsymbol{\theta}}_i + \sum_{i\in\mathcal{K}} \frac{1}{2}\tilde{\boldsymbol{\theta}}_{d_i}^\top \boldsymbol{\Gamma}_{d_i}^{-1}\tilde{\boldsymbol{\theta}}_{d_i}$$

where $\tilde{\boldsymbol{\theta}}_i = \hat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i$ and $\tilde{\boldsymbol{\theta}}_{d_i} = \hat{\boldsymbol{\theta}}_{d_i} - \boldsymbol{\theta}_{d_i}$ denote the parametric errors. Differentiating with respect to time yields:

$$\dot{\boldsymbol{V}} = \sum_{i\in\mathcal{K}} \frac{1}{2}\boldsymbol{z}_i^\top \dot{\boldsymbol{M}}_i \boldsymbol{z}_i + \sum_{i\in\mathcal{K}} \boldsymbol{z}_i^\top \boldsymbol{M}_i \boldsymbol{z}_i + \sum_{i\in\mathcal{K}} \tilde{\boldsymbol{\theta}}_i^\top \boldsymbol{\Gamma}_i^\top \dot{\hat{\boldsymbol{\theta}}}_i + \sum_{i\in\mathcal{K}} \tilde{\boldsymbol{\theta}}_{d_i}^\top \boldsymbol{\Gamma}_{d_i}^\top \dot{\hat{\boldsymbol{\theta}}}_{d_i}$$

Invoking Property 1 and substituting the adaptive laws (8.41)-(8.42), we get:

$$\dot{\boldsymbol{V}} = \sum_{i\in\mathcal{K}} -\boldsymbol{z}_i^\top \boldsymbol{K}\boldsymbol{z}_i - \boldsymbol{z}_i^\top \boldsymbol{D}_i \boldsymbol{z}_i \leq \boldsymbol{0} \tag{8.43}$$

Hence, we deduce $\boldsymbol{z}_i$, $\tilde{\boldsymbol{\theta}}_i$ and $\tilde{\boldsymbol{\theta}}_{d_i} \in L_\infty$. Moreover, from the definition of $\boldsymbol{z}_i$ in (8.38) , we conclude that $\boldsymbol{x}_O, \boldsymbol{v}_O \in L_\infty$, and consequently $\boldsymbol{v}_{O_i}^r, \dot{\boldsymbol{v}}_{O_i}^r \in L_\infty$. Furthermore, employing (8.39) we arrive at $\dot{\boldsymbol{z}} \in L_\infty$. Therefore, integrating both sides of (8.43) leads to:

$$\boldsymbol{V}(t) - \boldsymbol{V}(0) \leq \sum_{i\in\mathcal{K}} \int_0^t -\boldsymbol{z}_i^\top(\tau)\boldsymbol{K}\boldsymbol{z}_i - \boldsymbol{z}_i^\top \boldsymbol{D}_i \boldsymbol{z}_i(\tau) d\tau \tag{8.44}$$

Thus, $\sum_{i\in\mathcal{K}} \int_0^t -\boldsymbol{z}_i^\top(\tau)\boldsymbol{K}\boldsymbol{z}_i - \boldsymbol{z}_i^\top \boldsymbol{D}_i \boldsymbol{z}_i(\tau) d$ is bounded, which results in $\boldsymbol{z}_i \in L_2$. Finally, Barbalat's Lemma leads to $\boldsymbol{z}_i \to \boldsymbol{0}$, $\forall i \in \mathcal{K}$ as $t \to 0$, since $\boldsymbol{z}_i \in L_2$ and $\dot{\boldsymbol{z}}_i \in {}_\infty$, which completes the proof. □

8.3.2 Follower's Estimation Scheme

It should be noticed that the followers are not aware of either the object's desired configuration $\boldsymbol{x}_O^d$ or the obstacles' position in the workspace. However, even thought explicit communication among the leader and the followers is not permitted, the followers will estimate the object's desired trajectory profile by $\hat{\boldsymbol{x}}_O^{d_i}(t)$ $i \in \mathcal{N}$, via their own state measurements by adopting a novel prescribed performance estimator. Hence, let us define the error:

$$\boldsymbol{e}_i(t) = \boldsymbol{x}_O(t) - \hat{\boldsymbol{x}}_O^{d_i}(t) \in \mathbb{R}^6, \ i \in \mathcal{N}. \tag{8.45}$$

The expression of prescribed performance for each element of $\boldsymbol{e}_i(t) = [e_{i1}(t), \ldots, e_{i6}(t)]^\top$, $i \in \mathscr{N}$ is given by the following inequalities:

$$-\rho_{ij}(t) < e_{ij}(t) < \rho_{ij}(t),\ j = 1, \ldots, 6 \text{ and } i \in \mathscr{N} \tag{8.46}$$

for all $t \geq 0$, where $\rho_{ij}(t),\ j = 1, \ldots, 6$ and $i \in \mathscr{N}$ denote the corresponding performance functions. A candidate exponential performance function could be:

$$\rho_{ij}(t) = (\rho_{ij,0} - \rho_{ij,\infty})e^{-\lambda t} + \rho_{ij,\infty},\ i \in \mathscr{N} \tag{8.47}$$

where the constant λ dictates the exponential convergence rate, $\rho_{ij,\infty},\ i \in \mathscr{N}$ denotes the ultimate bound and $\rho_{ij,0}$ is chosen to satisfy $\rho_{ij,0} > |e_{ij}(0)|,\ i \in \mathscr{N}$. Hence, following the prescribed performance control technique [256], the estimation law is designed as follows:

$$\dot{\hat{x}}^{d_i}_{O_j} = k_{ij} \ln\left(\frac{1 + \frac{e_{ij}(t)}{\rho_{ij}(t)}}{1 - \frac{e_{ij}(t)}{\rho_{ij}(t)}}\right), k_{ij} > 0,\ j = 1, \ldots, 6 \tag{8.48}$$

for $i \in \mathscr{N}$, from which the followers' estimate $\hat{\boldsymbol{x}}^{d_i}_O(t) = [\hat{\boldsymbol{x}}^{d_i}_{O_1}(t), \ldots, \hat{\boldsymbol{x}}^{d_i}_{O_6}(t)]^\top,\ i \in \mathscr{N}$ is calculated via a simple integration. Moreover, differentiating (8.48) with respect to time, we acquire the desired acceleration signal:

$$\ddot{\hat{x}}^{d_i}_{O_j} = \frac{2k_{ij}}{1 - \left(\frac{e_{ij}(t)}{\rho_{ij}(t)}\right)^2} \frac{\dot{e}_{ij}(t)\rho_{ij}(t) - e_{ij}(t)\dot{\rho}_{ij}(t)}{\left(\rho_{ij}(t)\right)^2} \tag{8.49}$$

employing only the velocity $\dot{\boldsymbol{x}}_O(t)$ of the object, which can be easily calculated via (8.4), and not its acceleration which is unmeasurable.

Lemma 8.1. *Consider the error:*

$$\boldsymbol{e}_i(t) = \boldsymbol{x}_O(t) - \hat{\boldsymbol{x}}^{d_i}_O(t) = [e_{i1}(t), \ldots, e_{i6}(t)]^\top,\ i \in \mathscr{N} \tag{8.50}$$

where $\boldsymbol{x}_O(t)$ and $\hat{\boldsymbol{x}}^{d_i}_O(t),\ i \in \mathscr{N}$ denote the object's actual configuration and the estimation of the object's desired trajectory profile at the followers' side respectively. Given the appropriately selected performance functions $\rho_{ij}(t),\ j = 1, \ldots, 6$ and $i \in \mathscr{N}$ that satisfy $|e_{ij}(0)| < \rho_{ij}(0),\ j = 1, \ldots, 6$ and $i \in \mathscr{N}$ and incorporate the desired transient and steady state performance specifications, the estimation law (8.48) guarantees that $|e_{ij}(t)| < \rho_{ij}(t),\ j = 1 \ldots, 6$ and $i \in \mathscr{N}$ for all $t \geq 0$ as well as that $\hat{\boldsymbol{x}}^{d_i}_O$ and $\dot{\hat{\boldsymbol{x}}}^{d_i}_O$ remain bounded.

Proof: The proof follows identical arguments for each element of $\boldsymbol{e}_i(t),\ i \in \mathscr{N}$. Hence, let us define the normalized errors:

$$\xi_{ij} = \frac{e_{ij}(t)}{\rho_{ij}(t)},\ j = 1, \ldots, 6 \text{ and } i \in \mathscr{N}. \tag{8.51}$$

The estimation law (8.48) may be rewritten as a function of the normalized error ξ_{ij} as follows:

$$\dot{\hat{x}}^{d_i}_{O_j} = k_{ij} \ln\Big(\frac{1+\xi_{ij}}{1-\xi_{ij}}\Big),\ j = 1,\ldots,6 \text{ and } i \in \mathcal{N}. \tag{8.52}$$

Hence, differentiating ξ_{ij} with respect to time and substituting (8.52), we obtain:

$$\dot{\xi}_{ij} = h_{ij}(t,\xi_{ij}) \equiv \frac{\dot{x}_{O_j}(t) - k_{ij}\ln\Big(\frac{1+\xi_{ij}}{1-\xi_{ij}}\Big)}{\rho_{ij}(t)} - \xi_{ij}\frac{\dot{\rho}_{ij}(t)}{\rho_{ij}(t)} \tag{8.53}$$

We also define the non-empty and open set $\Delta_{\xi_{ij}} = (-1,1)$. In the sequel, we shall prove that $\xi_{ij}(t)$ never escapes a compat subset of $\Delta_{\xi_{ij}}$ and thus the performance bounds (8.18) are met. The following analysis is divided in two phases. First, we show that a maximal solution exists, such that $\xi_{ij}(t) \in \Delta_{\xi_{ij}} \forall t \in [0,\tau_{\max})$, and subsequently we prove by contradiction that $\tau_{\max}$ is extended to ∞.

Phase I: Since $|e_{ij}(0)| < \rho_{ij}(0)$, we conclude that $\xi_{ij}(0) \in \Delta_{\xi_{ij}}$. Moreover, owing to the smoothness of the object trajectory and the proposed estimation scheme (8.48) over $\Delta_{\xi_{ij}}$, the function $h_{ij}(t,\xi_{ij})$ is continuous for all $t \geq 0$ and $\xi_{ij} \in \Delta_{\xi_{ij}}$. Therefore, the hypotheses of the Theorem-11.2 hold and the existence of a maximal solution $\xi_{ij}(t)$ of (8.53) on a time interval $[0,\tau_{\max})$ such that $\xi_{ij}(t) \in \Delta_{\xi_{ij}}, \forall t \in [0,\tau_{\max})$ is ensured.

Phase II: Notice that the transformed error signal:

$$\epsilon_{ij}(t) = \ln\Big(\frac{1+\xi_{ij}(t)}{1-\xi_{ij}(t)}\Big),\ j = 1,\ldots,6 \tag{8.54}$$

for $i \in \mathcal{N}$ is well defined for all $t \in [0,\tau_{\max})$. Hence, consider the positive definite and radially unbounded function $V_{ij} = \frac{1}{2}(\epsilon_{ij})^2$. Differentiating with respect to time and substituting (8.53), we obtain:

$$\dot{V}_{ij} = \frac{2\epsilon_{ij}}{(1-\xi_{ij}^2)\rho_{ij}(t)}\Big(\dot{x}_{O_j}(t) - k_{ij}\epsilon_{ij} - \xi_{ij}\dot{\rho}_{ij}(t)\Big)$$

Since $\dot{x}_{O_j}(t),\ j = 1,\ldots,6$ was proven bounded in Theorem-8.1 for all $t \geq 0$, and $\dot{\rho}_{ij}(t)$ are bounded by construction, we conclude that:

$$|\dot{x}_{O_j}(t) - \xi_{ij}\dot{\rho}_{ij}(t)| \geq \bar{d}_{ij},\ j = 1,\ldots,6 \text{ and } i \in \mathcal{N}.$$

for an unknown positive constant $\bar{d}_{ij}$. Moreover, $\frac{1}{1-(\xi_{ij})^2} > 1, \forall \xi_{ij} \in \Delta_{\xi_{ij}}$ and $\rho_{ij}(t) > 0$ for all $t \geq 0$. Hence, we conclude that $\dot{V}_{ij} < 0$ when $|\epsilon_{ij}(t)| > \frac{\bar{d}_{ij}}{k_{ij}}$ and consequently that:

$$|\epsilon_{ij}(t)| \geq \bar{\epsilon}_{ij} = \max\Big\{|\epsilon_{ij}(0)|, \frac{\bar{d}_{ij}}{k_{ij}}\Big\},\ \forall t \in [0,\tau_{\max}) \tag{8.55}$$

Thus, invoking the inverse of (8.54), we get:

$$-1 < \frac{e^{-\bar{\epsilon}_{ij}} - 1}{e^{-\bar{\epsilon}_{ij}} + 1} = \underline{\xi}_{ij} \leq \xi_{ij}(t) \leq \bar{\xi}_{ij} = \frac{e^{\bar{\epsilon}_{ij}} - 1}{e^{\bar{\epsilon}_{ij}} + 1} < 1 \tag{8.56}$$

for $j = 1, \ldots, 6$ and $i \in \mathscr{N}$. Therefore, $\xi_{ij}(t) \in \Delta'_{\xi_{ij}} = \left[\underline{\xi}_{ij}, \bar{\xi}_{ij}\right]$, $\forall t \in [0, \tau_{\max})$, which is a nonempty and compact subset of $\Delta_{\xi_{ij}}$. Consequently, assuming $\tau_{\max} < \infty$, Proposition-11.1 dictates the existence of a time instant $t' \in [0, \tau_{\max})$ such that $\xi_{ij}(t') \notin \Delta'_{\xi_{ij}}$, which is a clear contradiction. Therefore, $\tau_{\max}$ is extended to ∞. As a result $\xi_{ij}(t) \in \Delta'_{\xi_{ij}} \in \Delta_{\xi_{ij}}$, $\forall t \geq 0$. Thus, from (8.51) and (8.54), we conclude that:

$$-\rho_{ij}(t) < \underline{\xi}_{ij}\rho_{ij}(t) \leq e_{ij}(t) \leq \bar{\xi}_{ij}\rho_{ij}(t) < \rho_{ij}(t), \forall t \geq 0$$

Finally, invoking (8.46) as well as the boundedness of $\boldsymbol{x}_O(t)$ and $\dot{\boldsymbol{x}}_O(t)$ from Theorem 8.1, we also deduce the boundedness of $\hat{\boldsymbol{x}}_O^{d_i}(t)$, and $\dot{\hat{\boldsymbol{x}}}_O^{d_i}(t)$ for all $t \geq 0$, which completes the proof. Based on the aforementioned estimation of the object's desired trajectory profile $\hat{\boldsymbol{x}}_O^{d_i}(t)$, $\dot{\hat{\boldsymbol{x}}}_O^{d_i}(t)$ and $\ddot{\hat{\boldsymbol{x}}}_O^{d_i}(t)$, $i \in \mathscr{N}$, we can easily derive the corresponding desired trajectory profile for the follower's End-Effector via (8.11), as follows:

$$\begin{aligned} \boldsymbol{v}_{O_{F_i}}^{d_i}(t) &= \boldsymbol{J}_O^{-1}(\boldsymbol{\eta}_{2,O})\dot{\hat{\boldsymbol{x}}}_O^{d_i}(t) \\ \dot{\boldsymbol{v}}_{O_{F_i}}^{d_i}(t) &= \boldsymbol{J}_O^{-1}(\boldsymbol{\eta}_{2,O})\ddot{\hat{\boldsymbol{x}}}_O^{d_i} + \dot{\boldsymbol{J}}_O^{-1}(\boldsymbol{\eta}_{2,O})\dot{\hat{\boldsymbol{x}}}_O^{d_i} \end{aligned} \tag{8.57}$$

Remark 8.3. The proposed estimation scheme is more robust against trajectory profiles with non-zero acceleration than previous results presented in [250, 251]. In particular, our method guarantees bounded closed loop signals and practical asymptotic stabilization of the estimation errors. Moreover, the aforementioned ultimate bounds depend directly on the design parameters $\rho_{ij,\infty}, j = 1, \ldots, 6$ and $i \in \mathscr{N}$ of the performance functions $\rho_{ij}(t), j = 1, \ldots, 6$ and $i \in \mathscr{N}$, which can be set arbitrarily small to a value reflecting the resolution of the measurement devices, thus achieving practical convergence of the estimation errors to zero. Additionally, the transient response depends on the convergence rate of the performance functions $\rho_{ij}(t), j = 1, \ldots, 6$ and $i \in \mathscr{N}$ that is directly affected by the parameter λ.

Remark 8.4. The appropriate selection of the performance function $\rho_{ij}(t)$ imposes transient and steady state performance characteristics on the estimation errors $e_{ij}(t)$ irrespectively of the design parameters $k_{ij}, i \in \mathscr{N}$, $j = 1 \ldots, 6$. In particular, for an initial estimation of the object's desired trajectory profile $\hat{\boldsymbol{x}}_O^{d_i}(0)$ and given $\boldsymbol{x}_O(0)$, the performance functions $\rho_{ij}(t)$, $i \in \mathscr{N}$ and, $j = 1, \ldots, 6$ are designed such that: i) $-\rho_{ij}(0) < e_{ij}(0) < \rho_{ij}(0)$ and ii) the desired transient and steady state performance specifications are met. On the other hand, extensive simulation studies have revealed that the selection of the control gains $\boldsymbol{K}$, $\boldsymbol{\Gamma}_i$, $\boldsymbol{\Gamma}_{d_i}$, $i \in \mathscr{K}$ can have positive influence on

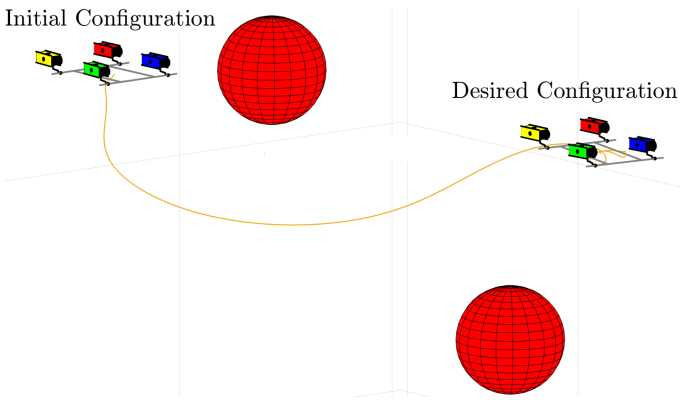


FIGURE 8.3: Four UVMSs transport a rigidly grasped object in a constrained workspace with static obstacles. Only the leading UVMS (indicated with blue color) is aware of the object's desired configuration and the obstacles' position in the workspace. A safe object trajectory in 3D space is indicated by orange color.

the closed loop system response in both the control input characteristics (magnitude and slew rate) as well as the evolution of the tracking errors. In particular, decreasing the gain values leads to slow convergence which is improved when adopting higher values, enlarging however the control effort both in magnitude and rate. Thus, an additional fine tuning is needed in real scenarios to retain the required control input signals within the feasible range that can be implemented by the actuators.

8.4 Simulation Study

In this section, the theoretical findings of this work are verified in a dynamic simulation environment built in MATLAB® presented in Section 2.4. We consider a scenario involving a 3D motion in a constrained workspace with static obstacles(see Fig.8.3). The object of interest was a pipe grid whose design parameters are given in Table-8.1. The initial and desired configuration of the object are $\boldsymbol{x}_O^{init} = [-4, 0.5, 0.6, 0, 0, 0]^\top$ and $\boldsymbol{x}_O^d = [10, 0, 0, 0, 0, 0]^\top$ respectively. Moreover, the calculation of the interaction force/-torque vector $\boldsymbol{\lambda_i}$, $i \in \mathscr{K}$ were performed following [156]. The cooperative transportation

TABLE 8.1: Object characteristics

Parameter	Value	Unit
Length	1.8	m
Pipe Diameter	5	cm
Mass in Air	1.5	kg

is performed by 4 UVMSs grasping the object at its corners. The blue UVMS acts as the leader. Thus, we assume that the desired object's configuration as well as the obstacles' position in the workspace are transferred to the leading UVMS beforehand. The obstacles are modeled as spheres (1 m radius) and are located in the workspace in order

to complicate the transportation task of the object. In this respect, a Navigation Function is constructed following (8.17) in order to handle the aforementioned constrained workspace. Since, only the leading UVMS (blue) is aware of the obstacles' position in the workspace and the object's desired configuration, the followers will estimate it via the proposed algorithm (8.48), by simply observing the motion of the object and without communicating explicitly with the leader. In order to test the robustness of the proposed scheme, in all subsequent simulations studies, the dynamics of the UVMS were affected by external disturbances in the form of slowly time varying sea currents modeled by the corresponding velocities $v_x^c = 0.3\sin(\frac{\pi}{15}t)\frac{m}{s}$ and $v_y^c = 0.3\cos(\frac{\pi}{15}t)\frac{m}{s}$. Finally, in all simulations the control gains and the parameters of the proposed estimator were chosen as shown in Table-8.2 and Table-8.3.

TABLE 8.2: Control gains of the proposed control scheme

Parameter	Value
$\boldsymbol{M}_{d_o},\ i \in \mathscr{K}$	$1 \cdot \boldsymbol{I}_{6\times 6}$
$\boldsymbol{D}_{d_o},\ i \in \mathscr{K}$	$1 \cdot \boldsymbol{I}_{6\times 6}$
$\boldsymbol{K}_{d_o},\ i \in \mathscr{K}$	$1 \cdot \boldsymbol{I}_{6\times 6}$
$\boldsymbol{M}_{d},\ i \in \mathscr{K}$	$1 \cdot \boldsymbol{I}_{6\times 6}$
$\boldsymbol{D}_{d},\ i \in \mathscr{K}$	$1 \cdot \boldsymbol{I}_{6\times 6}$
$\boldsymbol{K}_{d},\ i \in \mathscr{K}$	$1 \cdot \boldsymbol{I}_{6\times 6}$
k (See (8.17))	12
k_{NF} (See (8.18))	0.5
$\boldsymbol{\Gamma}_i,\ i \in \mathscr{K}$	$10 \cdot \boldsymbol{I}_{10\times 10}$
$\boldsymbol{\Gamma}_{d_i},\ i \in \mathscr{K}$	$10 \cdot \boldsymbol{I}_{6\times 6}$

TABLE 8.3: Parameters of the proposed Estimator

Parameter	Value
$k_j^i,\ i \in \mathscr{N},\ j = 1,\ldots,6$	1.2
$\rho_{1,0}^i,\ i \in \mathscr{N}$	5
$\rho_{2,0}^i,\ i \in \mathscr{N}$	4
$\rho_{3,0}^i,\ i \in \mathscr{N}$	4
$\rho_{4,0}^i,\ i \in \mathscr{N}$	1
$\rho_{5,0}^i,\ i \in \mathscr{N}$	1
$\rho_{6,0}^i,\ i \in \mathscr{N}$	1
$\rho_{j,\infty}^i,\ i \in \mathscr{N},\ j = 1,\ldots,6$	0.03
λ	1

8.4.1 Simulation Study A

The results are illustrated in Figs.8.4-8.8. The evolution of the system under the proposed methodology is given in Fig.8.4. It should be noticed that the UVMSs have transported cooperatively the grasped object from the initial configuration to the desired one without colliding with obstacles. By observing the object's tracking error (Fig.8.5) it can be con-

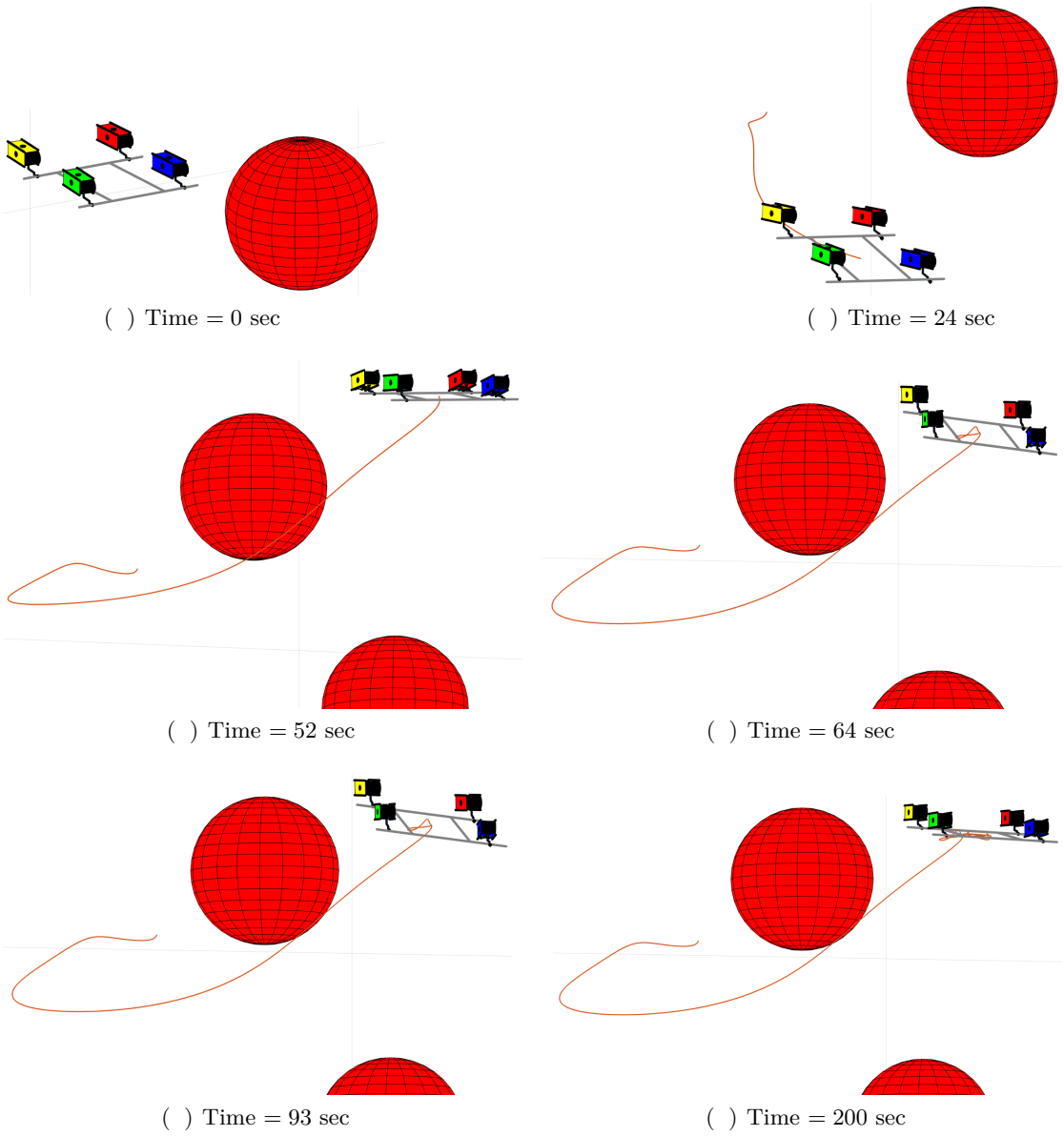


FIGURE 8.4: Simulation study A: The evolution of the proposed methodology in 6 consecutive time instants.

cluded that even under the influence of external disturbances, the errors in all directions converge very close to zero. The estimation errors of the proposed estimation scheme are presented in Fig.8.6. It can be easily seen that the estimation errors smoothly converge to zero and remain always within the performance envelope defined by the corresponding performance functions as it was expected from the aforementioned theoretical analysis. The evolution of the Navigation Function potential is presented in Fig.8.7. The value of Navigation Function remains strictly less than 1 during the simulation study which consequently means that obstacles have been successfully avoided. Moreover, the task space control commands in Fig-8.8.

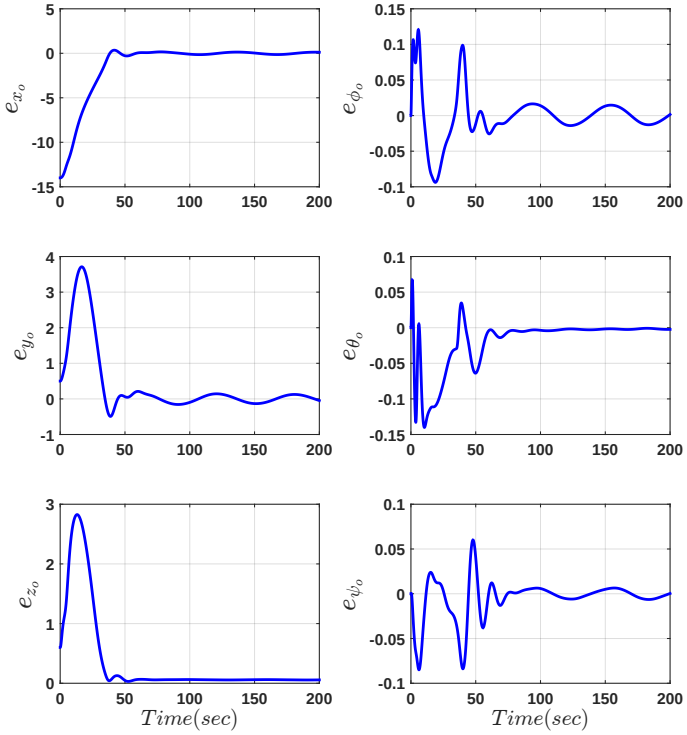


FIGURE 8.5: Simulation study A: The object tracking errors in all directions.

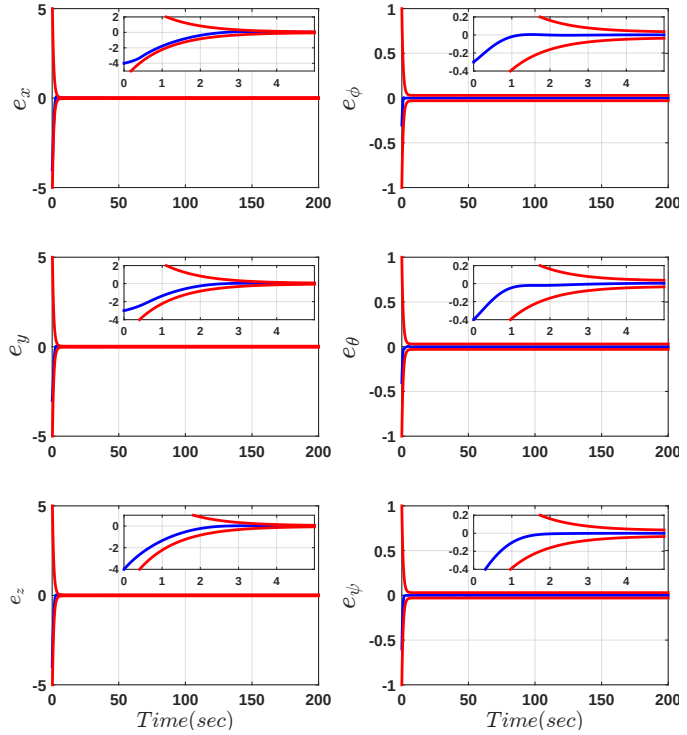


FIGURE 8.6: Simulation study A: The estimation errors along with the performance bounds imposed by the proposed method. The estimation errors and performance bounds are indicated by blue and red color respectively. Inside the box for each

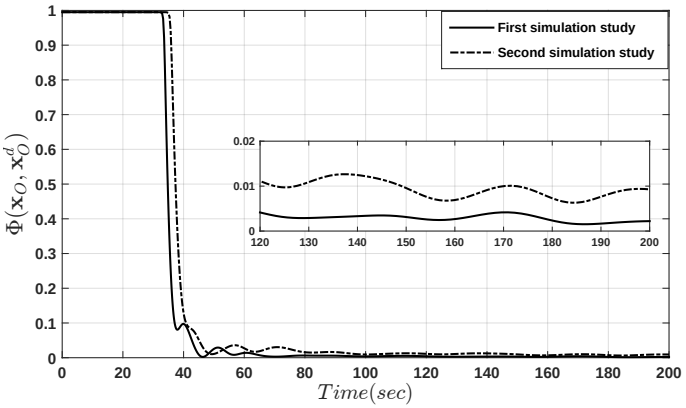


FIGURE 8.7: Simulation studies A-B: The evolution of the Navigation Function potential.

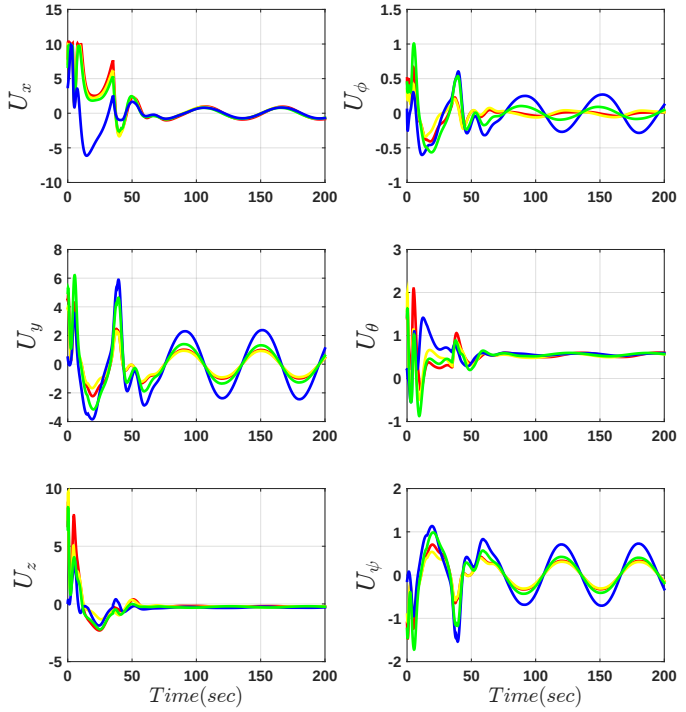


FIGURE 8.8: Simulation study A: The evolution of control inputs. The corresponding control inputs of the leading UVMS is indicated with blue color while the following UVMSs are indicated with red, yellow and green color respectively.

8.4.2 Simulation study B: Comparison with a Centralized Scheme

We considered exactly the same scenario, but instead a centralized control scheme was implemented to compare its response with the proposed scheme. More specifically, we incorporated the Navigation Function concept within the centralized control scheme presented in [2] (i.e., we modified the proportional term of the control scheme) in order to achieve obstacle avoidance. Observing the error trajectory tracking of Fig.8.9, it can be concluded that the system under the centralized control scheme reached to the desired configuration while avoiding the obstacles within the workspace. However, employing

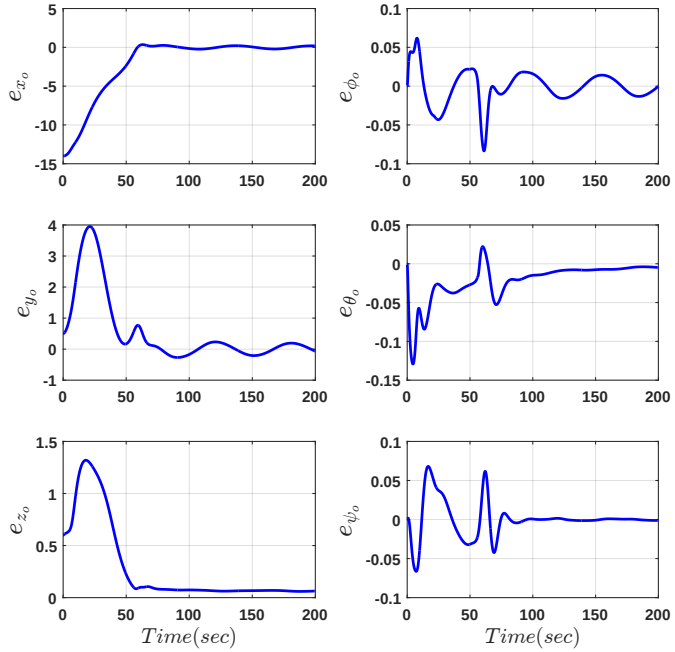


FIGURE 8.9: Simulation study B: The object tracking errors in all directions employing the centralized control scheme presented in [2].

the aforementioned centralized scheme requires 92 variables (i.e., 20 state and velocity variables $\boldsymbol{q}_i$ and $\boldsymbol{\zeta}_i$, $i \in \mathscr{K}$ for each UVMS as well as 12 state and velocity variables for the object) to be exchanged online among the robots. Considering floating point variables, this implementation requires the transfer of 2944 bytes in every control loop and it is well known that the achievable exchange rate depends directly on the available bandwidth. Therefore, considering a full-duplex communication, in order to achieve a 10 Hz exchanging rate, which is an ordinary rate in underwater robotics, a modem with at least 30 $kbytes/s$ bandwidth capability should be available which is unreasonably demanding based on the current technology in underwater acoustic modems and the small number of cooperating robots. For instance, high speed acoustic modems provided by *Evologics* company (e.g., S2C 48/78, S2C 40/80, S2C 42/65) supply data transfer rates up to 31 $kbits/s$. Although, recent advances in underwater acoustic modem technology accomplish continuously higher rate of communication, however, the number of participating robots remains still small owing to the limited bandwidth. For instance, in the considered scenario, if the number of cooperating robots increases by one (i.e., 5 cooperative UVMSs), the bandwidth requirements will increases respectively up to 35 kb/s which would also raise significantly the mission costs. On the contrary, it should be notice that the proposed control strategy imposes no restriction regarding the underwater communication bandwidth.

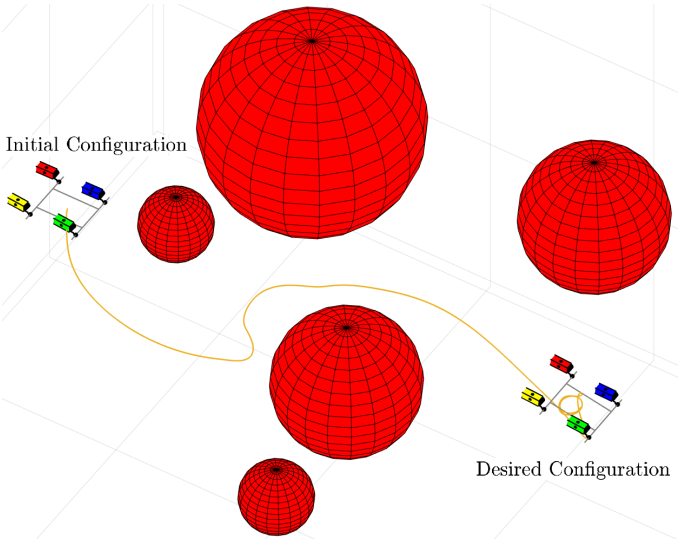


FIGURE 8.10: Simulation study C: The evolution of the object trajectory in 3D space is indicated by orange color.

8.4.3 Simulation study C

In order to verify the capabilities of the proposed scheme at full extend, a more complex scenario will be considered where more obstacles were involved within workspace. We kept unaltered the control gains, the estimator parameters and the external disturbances. The object initial and desired configuration are $\boldsymbol{x}_O^{init} = [-8, 0.5, 0.6, 0, 0, 0]^\top$ and $\boldsymbol{x}_O^d = [7, 0, 0, 0, 0, 0]^\top$ respectively. The workspace, initial and desired configuration, as well as the trajectory of the object are illustrated in Fig.8.10. By observing the object tracking

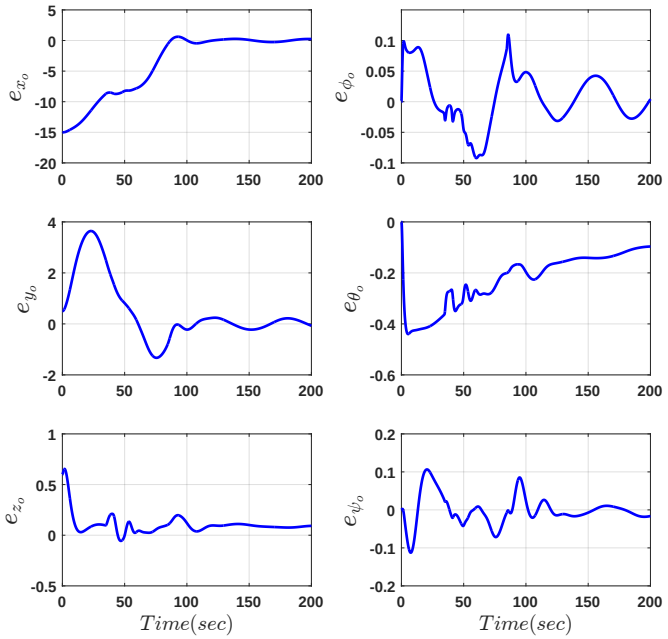


FIGURE 8.11: Simulation study C: The object tracking errors in all directions.

errors (Fig.8.11), it can be concluded that even in a more complicated case with more obstacles with increased uncertainty, the team of UVMSs successfully transported the object from the initial to the desired configuration while avoiding collisions with the obstacles of the workspace (see Fig.8.12). Finally, the evolution of task space control

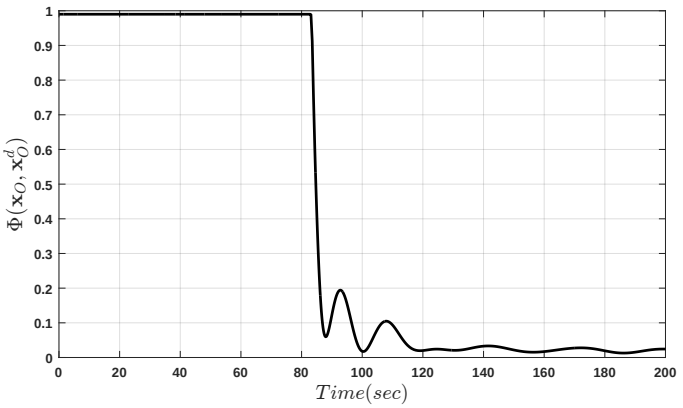


FIGURE 8.12: Simulation study C: The evolution of the Navigation Function.

signals are presented in Fig 8.13.

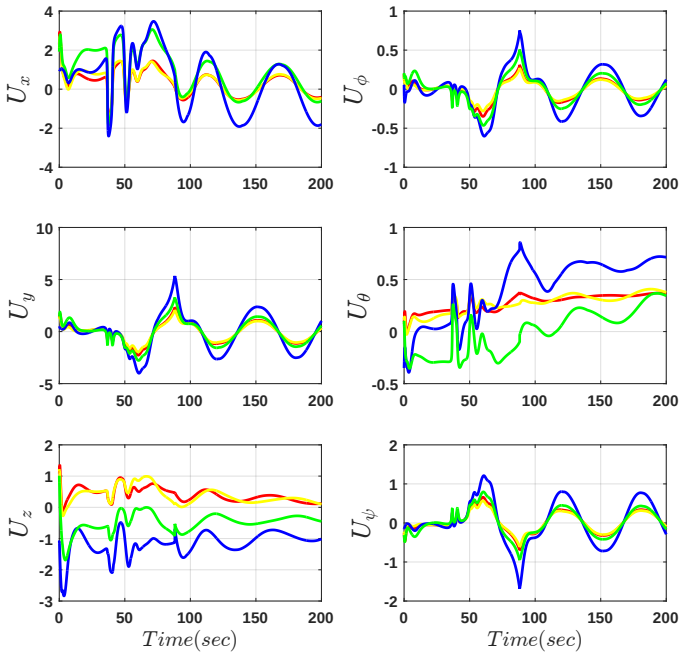


FIGURE 8.13: Simulation study C: Evolution of control inputs. The corresponding control inputs of the leading UVMS is indicated with blue color while the following UVMSs are indicated with red, yellow and green color respectively.

8.5 Conclusions

In this chapter, we presented a cooperative object transportation scheme for Underwater Vehicle Manipulator Systems under implicit communication, avoiding thus completely tedious explicit data transmission. In the proposed scheme, only the leading UVMS aware of the desired configuration of the object and the obstacles' position in the workspace, and aims at navigating the overall formation towards the goal configuration by trying to achieve the desired tracking behavior via an impedance control law, while avoiding collisions with the static obstacles. On the contrary, the followers adopt a prescribed performance estimation technique in order to estimate the object's desired trajectory and implement a similar impedance control law achieving in this way tracking of the desired trajectory despite the uncertainty and external disturbance in the object and the UVMS dynamics respectively. Each following UVMS employs the proposed estimator based on its own local measurements, thus the estimated desired trajectory of the object for each following UVMS is relative to its own inertial frame. Moreover, contrary to the existing work in the related literature, the proposed scheme imposes no restrictions on the underwater communication bandwidth. Furthermore, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities. Additionally, the feedback relies on each UVMS's force/torque measurements and no explicit data is exchanged online among the robots, thus reducing the required communication bandwidth and increasing robustness. Future research efforts will be devoted towards extending the proposed methodology for multiple UVMSs with underactuated vehicle dynamics.

Chapter 9

A Decentralized Predictive Control Approach for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems

9.1 Introduction

Most of the underwater manipulation tasks can be carried out more efficiently, if multiple UVMSs are cooperatively involved [2] (see Fig.9.1). On the other hand, Underwater multi-robot tasks are very demanding, with the most significant challenge being imposed by the strict communication constraints [15]. Therefor, employing communication based control structure in underwater environment may result in severe performance problems owing to the limited bandwidth and update rate of underwater acoustic devices. Moreover, the number of operating underwater robots in this case, is strictly limited owing to the narrow bandwidth of acoustic communication devices[161]. Therefore, to overcome such limitations, recent studies on underwater cooperative manipulation are dealing with designing control schemes under lean communication requirements.

Cooperative manipulation has been well-studied in the literature, especially the centralized schemes. Despite its efficiency, centralized control is less robust, and its complexity increases rapidly as the number of participating robots becomes large. On the other hand,

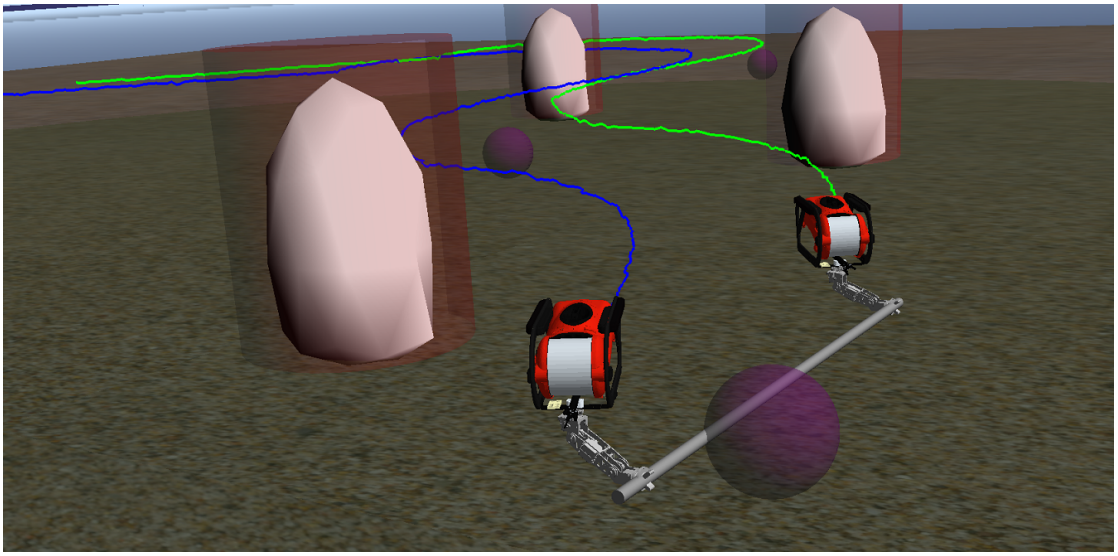


FIGURE 9.1: UVMSs under cooperative transportation.

decentralized cooperative manipulation schemes usually depend on explicit communication interchange among the robots. For instance, in recent studies [171, 172], potential fields methods were employed and a multi layer control structure was developed to manage the guidance of UVMSs and the manipulation tasks. Moreover, compelling results towards the same direction have been given in [173, 175]. In particular, a three-fold decentralized cooperative control strategy is proposed where initially, each robot individually finds out an optimal task space control velocity, which is transfered afterwards among the robots in order to obtain a commonly agreed velocity via a fusion policy. However, employing the aforementioned strategies, requires each robot to communicate with the whole robot team, which consequently restricts the number of robots involved in the cooperative manipulation task owing to bandwidth limitations. Moreover, regarding cooperative manipulation, various studies can be found on the literature employing decentralized control schemes where each agent involved to the cooperative tasks uses only its local information or observes [165, 257, 258]. Most of the aforementioned studies assume that each robots is equipped with a force/torque sensor on its end effector in order to acquire knowledge of the interaction contact forces/torques between the end effector and the common object, which may lead to a performance reduction due to sensor noise [259–261]. In addition, in most of the studies dealing with cooperative manipulation in literature, very important properties concerning the robotic manipulator systems such as: singular kinematic configurations of Jacobian matrix and joint limits have not been considered at all.

In this work, the problem of decentralized cooperative object transportation considering multiple UVMSs in a constrained workspace with static obstacles is addressed. Specifically, given N UVMSs rigidly grasp a common object, we design decentralized

controllers for each UVMS in order to navigate the object from an initial position to the final one, while avoiding significant constraints and limitations such as: kinematic and representation singularities, obstacles within the workspace, joint limits and control input saturations. More precisely, by exploiting the coupled dynamics between the robots and the object and by using certain load sharing coefficients we design a decentralized Nonlinear Model Predictive Control (NMPC) [58] for each UVMS in order to transport cooperatively the object and steer it along of a computed feasible path within the workspace. The design of that feasible path is based on the Navigation Function concept [185] which is adopted here in order to achieve distributed consensus on the object's desired trajectory as well to avoid collisions with the obstacles and the workspace boundary. In proposed control strategy we also take into account constraints that emanate from control input saturation as well kinematic and representation singularities. Moreover, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities. Finally, it should be noticed that the proposed methodology is decentralized, since each UVMS calculates its control signals without communicating with each other and exploits information acquired solely by onboard sensors, i.e., position and velocity measurements (e.g., sensor fusion based on measurement of various onboard sensors such as IMU, USBL and DVL), avoiding thus any tedious inter-robot explicit communication. This, consequently, increases significantly the robustness of the cooperative scheme and furthermore avoids any restrictions imposed by the acoustic communication bandwidth (e.g., the number of participating UVMSs).

9.2 Problem Formulation

Consider N UVMSs rigidly grasping an object[1] within a constrained workspace with static obstacles (see Fig 9.1). We also assume that each UVMS is fully-actuated at its end-effector frame. This assumption implies that all UVMSs are able to exert arbitrary forces and torques on the object along and around any directions. Moreover, we assume that the UVMSs are equipped with appropriate sensors, that allow them to measure their position and velocity (e.g., employing a sensor fusion technique based on measurements by various onboard sensors such as USBL, IMU, DVL and depth-sensor). Additionally, the geometric parameters of the both UVMSs and the commonly grasped object are considered known.

[1]The end-effector frame of each UVMS is always constant relative to the object's body fixed frame.

9.2.1 UVMS Kinematics

Consider N UVMSs operating in a bounded workspace $\mathscr{W} \subseteq \mathbb{R}^3$. First, we denote the coordinates of each UVMS's end effector by $\boldsymbol{p}_i = [\boldsymbol{\eta}_{1,p_i}^\top, \boldsymbol{\eta}_{2,p_i}^\top]^\top$ where $\boldsymbol{\eta}_{1,p_i}^\top = [x_{p_i}, y_{p_i}, z_{p_i}]^\top$ and $\boldsymbol{\eta}_{2,p_i}^\top = [\phi_{p_i}, \theta_{p_i}, \psi_{p_i}]^\top$ denote the position and the orientation expressed in Euler angles representation with respect to the inertial frame. Let $\boldsymbol{q}_i = [\boldsymbol{q}_{B,i}^\top,\ \boldsymbol{q}_{m,i}^\top]^\top \in \mathbb{R}^{n_i}$, with $n_i \in \mathbb{N}$, $i \in \mathscr{N}$ be the joint state variables of each UVMS, where $\boldsymbol{q}_{B,i} = [\boldsymbol{\eta}_{1,B_i}^\top, \boldsymbol{\eta}_{2,B_i}^\top]^\top$ is the vector that involves the position $\boldsymbol{\eta}_{1,B_i}^\top$ and the orientation $\boldsymbol{\eta}_{2,B_i}^\top$ of the base and $\boldsymbol{q}_{m,i}$ is the vector of the angular positions of the manipulator's joints. More specifically, $\boldsymbol{\eta}_{1,B_i}^\top = [x_{B_i}, y_{B_i}, z_{B_i}]^\top$ and $\boldsymbol{\eta}_{2,B_i}^\top = [\phi_{B_i}, \theta_{B_i}, \psi_{B_i}]^\top,\ i \in \{O, 1, \ldots, N\}$ denote the position and the orientation expressed in Euler angles representation with respect to the inertial frame. Thus, we have [10, 154]:

$$\dot{\boldsymbol{q}}_{B,i} = \boldsymbol{J}_{B,i}(\boldsymbol{q}_{B,i})\boldsymbol{\upsilon}_i,\ i \in \mathscr{N} \tag{9.1}$$

where $\boldsymbol{\upsilon}_i$ is the velocity of the vehicle expressed in the body-fixed frame and $\boldsymbol{J}_{B,i}(\boldsymbol{q}_{v,i})$ is the Jacobian matrix transforming the velocities from the body-fixed to the inertial frame. Let also define the UVMS' end effector generalized velocities by $\boldsymbol{\upsilon}_i = [\dot{\boldsymbol{\eta}}_{1,i}^\top, \boldsymbol{\omega}_i^\top]^\top,\ i \in \mathscr{N}$, where $\dot{\boldsymbol{\eta}}_{1,i}$ and $\boldsymbol{\omega}_i$ denote the linear and angular velocity respectively. In addition, the position and orientation of the UVMS end-effector with respect to inertial frame, is given by the forward kinematics of the complete system (arm and vehicle base) as follows:

$$\boldsymbol{p}_i = \mathscr{F}(\boldsymbol{q}_i)\ , i \in \mathscr{N} \tag{9.2}$$

Moreover, without any loss of generality, for the augmented UVMS system we get [10]:

$$\boldsymbol{\upsilon}_i = \boldsymbol{J}_i(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i,\ i \in \mathscr{N} \tag{9.3}$$

where $\dot{\boldsymbol{q}}_i = [\dot{\boldsymbol{q}}_{B,i}^\top, \dot{\boldsymbol{q}}_{m,i}^\top]^\top \in \mathbb{R}^{n_i}$ is the velocity vector involving the velocities of the vehicle with respect to the inertial frame as well as the joint velocities of the manipulator and $\boldsymbol{J}_i(\boldsymbol{q}_i)$ is the geometric Jacobian matrix [10]. Note that the $\boldsymbol{J}_{B,i}$ becomes singular at representation singularities, when $\theta_{B_i} = \pm\frac{\pi}{2}$ and $\boldsymbol{J}_i(\boldsymbol{q}_i)$ becomes singular at kinematic singularities defined by the set

$$Q_{s_i} = \{\boldsymbol{q}_i \in \mathbb{R}^{n_i} : \det(\boldsymbol{J}_i(\boldsymbol{q}_i)[\boldsymbol{J}_i(\boldsymbol{q}_i)]^\top) \geq \epsilon\},\ i \in \mathscr{N}. \tag{9.4}$$

with ϵ to be a small positive number.

9.2.2 UVMS Dynamics

Without any loss of generality, the dynamics of a UVMS after straightforward algebraic manipulations can be written as [10]:

$$\boldsymbol{M}_{q_i}(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_i + \boldsymbol{C}_{q_i}(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \boldsymbol{D}_{q_i}(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \boldsymbol{g}_{q_i}(\boldsymbol{q}_i) = \boldsymbol{\tau}_i - \boldsymbol{J}_i{}^{\top}\boldsymbol{\lambda}_i \tag{9.5}$$

for $i \in \mathcal{N}$, where $\boldsymbol{\lambda}_i$ is the vector of generalized interaction forces and torques that UVMS exerts on the object, $\boldsymbol{\tau}_i$ denotes the vector of control inputs (forces and torques), $\boldsymbol{M}_{q_i}(\boldsymbol{q}_i)$ is the inertial matrix, $\boldsymbol{C}_{q_i}(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)$ represents coriolis and centrifugal terms, $\boldsymbol{D}_{q_i}(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)$ models dissipative effects and $\boldsymbol{g}_i(\boldsymbol{q}_i)$ encapsulates the gravity and buoyancy effects. In view of (9.3) we have:

$$\dot{\boldsymbol{v}}_i = \boldsymbol{J}_i(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_i + \dot{\boldsymbol{J}}_i(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i, \;\; i \in \mathcal{N} \tag{9.6}$$

where $\dot{\boldsymbol{J}}_i(\boldsymbol{q}_i) \in \mathbb{R}^{6\times n_i}$ represents the Jacobian derivative function. Then, by employing the differential kinematics (9.3) as well as (9.6), we obtain from (9.5) the transformed task space dynamics [252]:

$$\boldsymbol{M}_i(\boldsymbol{q}_i)\dot{\boldsymbol{v}}_i + \boldsymbol{C}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\boldsymbol{v}_i + \boldsymbol{D}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\boldsymbol{v}_i + \boldsymbol{g}_i(\boldsymbol{q}_i) = \boldsymbol{u}_i - \boldsymbol{\lambda}_i \tag{9.7}$$

for all $i \in \mathcal{N}$ with corresponding task space terms $\boldsymbol{M}_i \in \mathbb{R}^{6\times 6}$, $\boldsymbol{C}_i \in \mathbb{R}^{6\times 6}$, $\boldsymbol{D}_i \in \mathbb{R}^{6\times 6}$, $\boldsymbol{g}_i \in \mathbb{R}^6$:

$$\begin{aligned}
\boldsymbol{M}_i(\boldsymbol{q}_i) &= [\boldsymbol{J}_i(\boldsymbol{q}_i)\boldsymbol{M}_{q_i}^{-1}\boldsymbol{J}_i(\boldsymbol{q}_i)^{\top}]^{-1} \\
\boldsymbol{C}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\boldsymbol{J}_i(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i &= \boldsymbol{M}_i(\boldsymbol{q}_i)\big[\boldsymbol{J}_i(\boldsymbol{q}_i)\boldsymbol{M}_{q_i}^{-1}\boldsymbol{C}_{q_i} - \dot{\boldsymbol{J}}_i(\boldsymbol{q}_i)\big]\dot{\boldsymbol{q}}_i \\
\boldsymbol{D}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\boldsymbol{J}_i(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i &= \boldsymbol{M}_i(\boldsymbol{q}_i)\boldsymbol{J}_i(\boldsymbol{q}_i)\boldsymbol{M}_{q_i}^{-1}\boldsymbol{D}_{q_i}\dot{\boldsymbol{q}}_i \\
\boldsymbol{g}_i(\boldsymbol{q}_i) &= \boldsymbol{M}_i(\boldsymbol{q}_i)\boldsymbol{J}_i(\boldsymbol{q}_i)\boldsymbol{M}_{q_i}^{-1}\boldsymbol{g}_{q_i}
\end{aligned}$$

Moreover $\boldsymbol{u}_i \in \mathbb{R}^6$ is the vector of task space generalized forces/torques. It is worth noting that the vector of control inputs $\boldsymbol{\tau}_i, \;\; i \in \mathcal{K}$ can be related to the task space wrench $\boldsymbol{u}_i \in \mathbb{R}^6, \;\; i \in \mathcal{K}$ via:

$$\boldsymbol{\tau}_i = \boldsymbol{J}_i^{\top}(\boldsymbol{q}_i)\boldsymbol{u}_i + \boldsymbol{\tau}_{i0}(\boldsymbol{q}_i) \tag{9.8}$$

where the vector $\boldsymbol{\tau}_{i0}(\boldsymbol{q}_i)$ does not contribute to the end effector's wrench $\boldsymbol{u}_i$ (i.e., it belongs to the null space of the Jacobian $\boldsymbol{J}_i^{\top}$) and can be regulated independently to achieve secondary tasks (e.g., maintaining manipulator's joint limits, increasing the

manipulability)[252][2]. The UVMS task space dynamics (9.7) can be written in vector form as:

$$\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{v}} + \boldsymbol{C}(\dot{\boldsymbol{q}}, \boldsymbol{q})\boldsymbol{v} + \boldsymbol{D}(\dot{\boldsymbol{q}}, \boldsymbol{q})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{u} - \boldsymbol{\lambda} \tag{9.9}$$

where $\boldsymbol{v} = [\boldsymbol{v}_1^\top, \ldots, \boldsymbol{v}_N^\top]^\top \in \mathbb{R}^{6N}$, $\boldsymbol{M} = \text{diag}\{[\boldsymbol{M}_i]\} \in \mathbb{R}^{6N \times 6N}$, $\boldsymbol{C} = \text{diag}\{[\boldsymbol{C}_i]\} \in \mathbb{R}^{6N \times 6N}$, $\boldsymbol{D} = \text{diag}\{[\boldsymbol{D}_i]\} \in \mathbb{R}^{6N \times 6N}$, $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_1^\top, \ldots, \boldsymbol{\lambda}_N^\top]^\top$, $\boldsymbol{u} = [\boldsymbol{u}_1^\top, \ldots, \boldsymbol{u}_N^\top]^\top$, $\boldsymbol{g} = [\boldsymbol{g}_1^\top, \ldots, \boldsymbol{g}_N^\top]^\top \in \mathbb{R}^{6N}$.

9.2.3 Object Dynamic

We denote the coordinates of the object coordinates by $\boldsymbol{x}_O = [\boldsymbol{\eta}_{1,O}^\top, \boldsymbol{\eta}_{2,O}^\top]^\top$ where $\boldsymbol{\eta}_{1,O}^\top = [x_O, y_O, z_O]^\top$ and $\boldsymbol{\eta}_{2,O}^\top = [\phi_O, \theta_O, \psi_O]^\top$ denote the position and the orientation expressed in Euler angles representation with respect to the inertial frame. Let also define the object generalized velocities by $\boldsymbol{v}_O = [\dot{\boldsymbol{\eta}}_{1,O}^\top, \boldsymbol{\omega}_O^\top]^\top$. Without any loss of generality, we consider the following second order dynamic for the object, which can be derived based on the Newton-Euler formulations:

$$\dot{\boldsymbol{x}}_O = \boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O})^{-1}\boldsymbol{v}_O \tag{9.10}$$

$$\boldsymbol{M}_O(\boldsymbol{x}_O)\dot{\boldsymbol{v}}_O + \boldsymbol{C}_O(\boldsymbol{v}_O, \boldsymbol{x}_O)\boldsymbol{v}_O + \boldsymbol{D}_O(\boldsymbol{v}_O, \boldsymbol{x}_O)\boldsymbol{v}_O + \boldsymbol{g}_O = \boldsymbol{\lambda}_O \tag{9.11}$$

where $\boldsymbol{M}_O(\boldsymbol{x}_O)$ is the positive definite inertia matrix, $\boldsymbol{C}_O(\boldsymbol{v}_O, \boldsymbol{x}_O)$ is the Coriolis matrix, $\boldsymbol{g}_O$ is the vector of gravity and buoyancy effects, $\boldsymbol{D}_O(\boldsymbol{v}_O, \boldsymbol{x}_O)$ models dissipative effects and $\boldsymbol{\lambda}_O$ is the vector of generalized forces acting on the object's center of mass. Moreover, $\boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O})$ is the object representation Jacobian that transforms the Euler angle rates into velocity $\boldsymbol{\omega}_O$ and can be given as:

$$\boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O}) = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{J}''_O(\boldsymbol{\eta}_{2,O}) \end{bmatrix}, \tag{9.12}$$

with:

$$\boldsymbol{J}''_O(\boldsymbol{\eta}_{2,O}) = \begin{bmatrix} 1 & 0 & -\sin(\theta_O) \\ 0 & \cos(\phi_O) & \cos(\theta_O)\sin(\phi_O) \\ 0 & -\sin(\phi_O) & \cos(\theta_O)\cos(\phi_O) \end{bmatrix}.$$

[2] For more details on task priority based control and redundancy resolution for UVMSs the reader is referred to [145] and [151].

Note that the $\boldsymbol{J}''_O(\boldsymbol{\eta}_{2,O})$ is singular when when $\theta_O = \pm\frac{\pi}{2}$ and $\boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O})$ becomes singular at kinematic singularities defined by the set

$$O_s = \{\boldsymbol{x}_O \in \mathbb{R}^6 : \det(\boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O})[\boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O})]^\top) \geq \epsilon\}. \tag{9.13}$$

with ϵ to be a small positive number.

Hence, the problem that we aim to solve in this chapter is stated as follows:

Problem 9.1. *Given N UVMSs operating in a constrained workspace $\mathscr{W}$ and rigidly grasping an object. Given a desired configuration for the object $\boldsymbol{x}_O^d$, design distributed control protocols $\boldsymbol{\tau}_i$, $\in \mathscr{N}$ that navigate safely the whole robotic team to the desired configuration while satisfying the following specifications:*

1. *Collision avoidance with the obstacles and the boundary of the workspace;*
2. *Impose no strict requirements regarding the underwater communication bandwidth;*
3. *Avoiding the singular properties of the system;*
4. *Achieve distributed consensus on a mutually agreed trajectory of the commonly grasped object;*

9.3 Control Methodology

In this section we propose a solution to the Problem-9.1. The proposed approach builds on designing a Nonlinear Model Predictive Control scheme for the system of the UVMSs and the object. The proposed approach is decentralized, since it does not based on a centralized system calculating all control inputs for the whole team of UVMS. Moreover, thank to the novel formulation of the problem, the proposed control strategy relieves the team of robots from intense inter-robot communication during the execution of the collaborative tasks. This, consequently, increases significantly the robustness of the cooperative scheme and furthermore avoids any restrictions imposed by the acoustic communication bandwidth (e.g., the number of participating UVMSs). In this way, we assume that the all of UVMS are aware of both the desired configuration of the object as well as of the obstacles position in the workspace[3]. Thus, the control objective is to navigate the overall formation towards the goal configuration while avoiding collisions with the static obstacles that lie within the workspace. Towards this direction, the concept of Navigation Functions [185] will be incorporated to deal with consensus on a

[3]The desired configuration of the object can be transmitted to each UVMS before executing the cooperation task.

mutually agreed trajectory of the commonly object. That overall dynamics are decoupled next among the object and the robots in N parts accounting individually for each UVMS $i \in \mathcal{N}$ by using certain load sharing coefficients. Each UVMS at each sampling time, solves a NMPC subject to its corresponding part of that overall dynamics and a number of inequality constraints that incorporate its internal limitations (e.g., joint limits, kinematic and representation singularities, collision between the arm and the base, manipulability).

9.3.1 Coupled Dynamics

Consider the N UVMS rigidly grasping a common object. Owing to the rigid grasp of the object, the following equations hold:

$$\boldsymbol{p}_i = \boldsymbol{x}_O + \begin{bmatrix} {}^I\boldsymbol{R}_O \boldsymbol{l}_i \\ \boldsymbol{\alpha}_i \end{bmatrix}, \; i \in \mathcal{N} \tag{9.14}$$

where the vectors $\boldsymbol{l}_i = [l_{ix}, l_{iy}, l_{iz}]^\top$ and $\boldsymbol{\alpha}_i = [\alpha_{ix}, \alpha_{iy}, \alpha_{iz}]^\top, \; i \in \mathcal{N}$ represent the *constant* relative position and orientation of the end-effector w.r.t the object, expressed in the object's frame and ${}^I\boldsymbol{R}_O$ denotes the rotation matrix which describes the orientation of the object expressed in the inertial frame $\{I\}$. Thus, using (9.14) each UVMS can compute the object's position w.r.t inertial frame $\{I\}$, since the object geometric parameters are considered known. Furthermore, due to the grasping rigidly, it holds that $\boldsymbol{\omega}_i = \boldsymbol{\omega}_O, \; i \in \mathcal{N}$, one obtains:

$$\boldsymbol{v}_O = \boldsymbol{J}_{i_O} \boldsymbol{v}_i, \; i \in \mathcal{N} \tag{9.15}$$

where $\boldsymbol{J}_{i_O}, \; i \in \mathcal{N}$ denotes the Jacobian from the end-effector of each UVMS to the object's center of mass, that is defined as:

$$\boldsymbol{J}_{i_O} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & -\boldsymbol{S}(\boldsymbol{l}_i) \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \in \mathbb{R}^{6\times6}, \quad i \in \mathcal{N}$$

where $\boldsymbol{S}(\boldsymbol{l}_i)$ is the skew-symmetric matrix of vector $\boldsymbol{l}_i = [l_{ix}, l_{iy}, l_{iz}]^\top$ defined as:

$$\boldsymbol{S}(\boldsymbol{l}_i) = \begin{bmatrix} 0 & -l_{iz} & l_{iy} \\ l_{iz} & 0 & -l_{ix} \\ -l_{iy} & l_{ix} & 0 \end{bmatrix} \in \mathbb{R}^{3\times3}, \; i \in \mathcal{N}$$

Notice that $\boldsymbol{J}_{i_O}, \; i \in \mathcal{N}$ are always full-rank owing to the grasp rigidity and hence obtain a well defined inverse. Thus, the object's velocity can be easily computed via

(9.15). Moreover, from (9.15), one obtains the acceleration relation:

$$\dot{\boldsymbol{v}}_O = \boldsymbol{J}_{i_O}\dot{\boldsymbol{v}}_i + \dot{\boldsymbol{J}}_{i_O}\boldsymbol{v}_i,\ i \in \mathscr{N} \tag{9.16}$$

which will be used in the subsequent analysis. In addition, the kineto-statics duality along with the grasp rigidity suggest that the force $\boldsymbol{\lambda}_O$ acting on the object's center of mass and the generalized forces $\boldsymbol{\lambda}_i,\ i \in \mathscr{N}$, exerted by the UVMSs at the grasping points, are related through:

$$\boldsymbol{\lambda}_O = \boldsymbol{G}^\top \boldsymbol{\lambda} \tag{9.17}$$

where:

$$\boldsymbol{G} = \left[[\boldsymbol{J}_{O_1}]^\top, \ldots, [\boldsymbol{J}_{O_N}]^\top\right]^\top \in \mathbb{R}^{6N\times 6} \tag{9.18}$$

is the full column-rank grasp matrix, $\boldsymbol{J}_{O_i} = [\boldsymbol{J}_{i_O}]^{-1},\ i \in \mathscr{N}$ and $\boldsymbol{\lambda} = [\boldsymbol{\lambda_1}^\top, \ldots, \boldsymbol{\lambda_N}^\top]^\top$ is the vector of overall interaction forces and torques.

Remark 9.1. Wrenches that lie on the null space of the grasp matrix $\boldsymbol{G}^\top$ do not contribute to the object dynamics. Therefore, we may incorporate in the control scheme an extra component $\boldsymbol{\lambda}_{int,i} = (\boldsymbol{I} - (\boldsymbol{G}^\top)^{\#}\boldsymbol{G}^\top)\boldsymbol{\lambda}^d_{int},\ i \in \mathscr{N}$, that belongs to the null space of $\boldsymbol{G}^\top$, in order to regulate the steady state internal forces, where $(\boldsymbol{G}^\top)^{\#}$ denotes the generalized inverse of $\boldsymbol{G}^\top$. Notice that owing to the rigid grasp, $l_i,\ i \in \mathscr{N}$ remain constant. Thus, since $l_i,\ i \in \mathscr{N}$ are considered known to the team of UVMSs[4], if $\boldsymbol{\lambda}^d_{int}$ is chosen constant, no communication is needed during task execution in order to compute $\boldsymbol{G}^\top$, $(\boldsymbol{G}^\top)^{\#}$ and $\boldsymbol{\lambda}_{int,i}$.

By substituting (9.9) into (9.17) one obtains:

$$\boldsymbol{\lambda} = \boldsymbol{G}^\top\left[\boldsymbol{u} - \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{v}} - \boldsymbol{C}(\dot{\boldsymbol{q}},\boldsymbol{q})\boldsymbol{v} - \boldsymbol{D}(\dot{\boldsymbol{q}},\boldsymbol{q})\boldsymbol{v} - \boldsymbol{g}(\boldsymbol{q})\right] \tag{9.19}$$

which, after substituting (9.15), (9.16), (9.10) and rearranging terms, yields the overall system coupled dynamics:

$$\widetilde{\boldsymbol{M}}(\tilde{\boldsymbol{q}}_{ov})\dot{\boldsymbol{v}}_O + \widetilde{\boldsymbol{C}}(\tilde{\boldsymbol{q}}_{ov})\boldsymbol{v}_O + \widetilde{\boldsymbol{D}}(\tilde{\boldsymbol{q}}_{ov})\boldsymbol{v}_O + \widetilde{\boldsymbol{g}}(\tilde{\boldsymbol{q}}_{ov}) = \boldsymbol{G}^\top\boldsymbol{u} \tag{9.20}$$

[4]This can be achieved by using the acoustic modems before beginning the task execution.

where $\tilde{\boldsymbol{q}}_{ov} = [\boldsymbol{q}^\top, \dot{\boldsymbol{q}}^\top, \boldsymbol{x}_O^\top, \boldsymbol{v}_O^\top]^\top$ and:

$$\begin{aligned}
\widetilde{\boldsymbol{M}}(\tilde{\boldsymbol{q}}_{ov}) &= \boldsymbol{M}_O(\boldsymbol{x}_O) + \boldsymbol{G}^\top \boldsymbol{M}(\boldsymbol{q})\boldsymbol{G} \\
\widetilde{\boldsymbol{C}}(\tilde{\boldsymbol{q}}_{ov}) &= \boldsymbol{C}_O(\boldsymbol{v}_O, \boldsymbol{x}_O) + \boldsymbol{G}^\top \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{G}}(\dot{\boldsymbol{q}}, \boldsymbol{q}) + \boldsymbol{G}^\top \boldsymbol{C}(\dot{\boldsymbol{q}}, \boldsymbol{q})\boldsymbol{G} \\
\widetilde{\boldsymbol{D}}(\tilde{\boldsymbol{q}}_{ov}) &= \boldsymbol{D}_O(\boldsymbol{v}_O, \boldsymbol{x}_O) + \boldsymbol{G}^\top \boldsymbol{D}(\dot{\boldsymbol{q}}, \boldsymbol{q})\boldsymbol{G} \\
\widetilde{\boldsymbol{g}}(\tilde{\boldsymbol{q}}_{ov}) &= \boldsymbol{g}_O(\boldsymbol{x}_O) + \boldsymbol{G}^\top \boldsymbol{g}(\boldsymbol{q})
\end{aligned}$$

Now, consider the design constants $c_i,\ i \in \mathscr{N}$ satisfying:

$$c_i \in (0,1), \forall i \in \mathscr{N} \qquad \text{and} \qquad \sum_{i \in \mathscr{N}} c_i = 1, \tag{9.21}$$

that we introduce in order to act as the load sharing coefficients for the team of UVMS. In view of (9.21), the object dynamics (9.11) can be rewritten as:

$$\sum_{i \in \mathscr{N}} c_i \Big\{ \boldsymbol{M}_O(\boldsymbol{x}_O)\dot{\boldsymbol{v}}_O + \boldsymbol{C}_O(\boldsymbol{x}_O, \boldsymbol{v}_O)\boldsymbol{v}_O + \boldsymbol{D}_O(\boldsymbol{x}_O, \boldsymbol{v}_O)\boldsymbol{v}_O + \boldsymbol{g}_O(\boldsymbol{x}_O) \Big\} = \sum_{i \in \mathscr{N}} \boldsymbol{J}_{O_i}^\top \boldsymbol{\lambda}_i \tag{9.22}$$

from which, by employing (9.3), (9.6), (9.15), (9.7) and (9.16), and after straightforward algebraic manipulations, we obtain the coupled dynamics:

$$\sum_{i \in \mathscr{N}} \Big\{ \widetilde{\boldsymbol{M}}_i(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_i + \widetilde{\boldsymbol{C}}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \widetilde{\boldsymbol{D}}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\dot{\boldsymbol{q}}_i + \widetilde{\boldsymbol{g}}_i(\boldsymbol{q}_i) \Big\} = \sum_{i \in \mathscr{N}} \boldsymbol{J}_{O_i}^\top \boldsymbol{u}_i \tag{9.23}$$

where:

$$\begin{aligned}
\widetilde{\boldsymbol{M}}_i(\boldsymbol{q}_i) &= c_i \boldsymbol{M}_O \boldsymbol{J}_{i_O} \boldsymbol{J}_i + \boldsymbol{J}_{O_i}^\top \boldsymbol{M}_i \boldsymbol{J}_i \\
\widetilde{\boldsymbol{C}}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i) &= c_i \Big[\boldsymbol{M}_O \boldsymbol{J}_{i_O} \dot{\boldsymbol{J}}_i + \boldsymbol{M}_O \dot{\boldsymbol{J}}_{i_O} \boldsymbol{J}_i + \boldsymbol{C}_O \boldsymbol{J}_{i_O} \boldsymbol{J}_i \Big] + \boldsymbol{J}_{O_i}^\top \Big[\boldsymbol{M}_i \dot{\boldsymbol{J}}_i + \boldsymbol{C}_i \boldsymbol{J}_i \Big] \\
\widetilde{\boldsymbol{D}}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i) &= c_i \boldsymbol{D}_O \boldsymbol{J}_{i_O} \boldsymbol{J}_i + \boldsymbol{J}_{O_i}^\top \boldsymbol{D}_i \boldsymbol{J}_i \\
\widetilde{\boldsymbol{g}}_i(\boldsymbol{q}_i) &= c_i \boldsymbol{g}_O + \boldsymbol{J}_{O_i}^\top \boldsymbol{g}_i
\end{aligned}$$

which is the distributed version of (9.20), since for each UVMS, it is based only individually on its locally measurements (i.e., $\boldsymbol{q}_i$ and $\dot{\boldsymbol{q}}_i$). Now, by using the notation $\boldsymbol{x}_i = [\boldsymbol{q}_i^\top, \dot{\boldsymbol{q}}_i^\top]^\top$, the decentralized dynamics of each UVMS based on (9.23), can be written as compact form:

$$\dot{\boldsymbol{x}}_i = f_i(\boldsymbol{x}_i, \boldsymbol{u}_i) = \begin{bmatrix} f_{i_1}(\boldsymbol{x}_i) \\ f_{i_2}(\boldsymbol{x}_i, \boldsymbol{u}_i) \end{bmatrix}, i \in \mathscr{N} \tag{9.24}$$

where:

$$f_{i_1}(\boldsymbol{x}_i) = \dot{\boldsymbol{q}}_i$$
$$f_{i_2}(\boldsymbol{x}_i, \boldsymbol{u}_i) = \widetilde{\boldsymbol{M}}_i^{\#}(\boldsymbol{q}_i)\Big(\boldsymbol{J}_{O_i}^{\top}(\boldsymbol{q}_i)\boldsymbol{u}_i - \widetilde{\boldsymbol{C}}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\dot{\boldsymbol{q}}_i - \widetilde{\boldsymbol{D}}_i(\dot{\boldsymbol{q}}_i, \boldsymbol{q}_i)\dot{\boldsymbol{q}}_i - \widetilde{\boldsymbol{g}}_i(\boldsymbol{q}_i)\Big)$$

with:

$$\widetilde{\boldsymbol{M}}_i^{\#}(\boldsymbol{q}_i) = \widetilde{\boldsymbol{M}}_i(\boldsymbol{q}_i)\Big[\widetilde{\boldsymbol{M}}_i(\boldsymbol{q}_i)\widetilde{\boldsymbol{M}}_i^{\top}(\boldsymbol{q}_i)\Big]^{-1}$$

9.3.2 Description of the Workspace

Consider the team of N UVMSs operating in a bounded workspace $\mathscr{W} \subseteq \mathbb{R}^3$ with boundary $\partial\mathscr{W}$. The object of interest is a rigid body which is required to be transported cooperatively by the robot team from an initial to a goal position. Without any loss of the generality, the obstacles, the robots as well as the workspace are all modeled by spheres (i.e., we adopt the spherical world representation [185]). However, the proposed control strategy could be extended to more general and complex geometries following the analysis in [185]. In this spirit, let $\mathscr{B}(\boldsymbol{x}_O, r_0)$ be a closed ball that covers the volume of the object and has radius r_0. We also define the closed balls $\mathscr{B}(\boldsymbol{p}_i, \bar{r}), i \in \mathscr{K}$, centered at the end-effector of each UVMS that cover the robot volume for all possible configurations. Notice that the value of $\bar{r}$ can be calculated easily for each UVMS based solely on its own design parameters. We also assume that the distance among the grasping points on the given object is at least $2\bar{r}$. In particular, the distance $2\bar{r}$ denotes the minimum allowed distance at which two bounding spheres $\mathscr{B}(\boldsymbol{p}_i, \bar{r})$ and $\mathscr{B}(\boldsymbol{p}_j, \bar{r})$ $i, j \in \mathscr{K}, i \neq j$ do not collide. Furthermore, we define a ball area $\mathscr{B}(\boldsymbol{x}_O, R)$ located at $\boldsymbol{x}_O$ with radius $R = \bar{r} + r_o$ that includes the whole volume of the robotic team and the object (see Fig. 9.2). Finally, the $\mathscr{M}$ static obstacles within the workspace are defined as closed spheres described by $\pi_m = \mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m})$, $m \in \{1, \ldots, \mathscr{M}\}$, where $\boldsymbol{p}_{\pi_m} \in \mathbb{R}^3$ is the center and the $r_{\pi_m} > 0$ the radius of the obstacle π_m. Obviously, the ultimate goal of the proposed cooperative control strategy is to transport the object from the initial configuration to the desired one, without colliding with the obstacles and the boundary of workspace. Additionally, based on the property of spherical world [185], for each pair of obstacles $m, m' \in \{1, \ldots, \mathscr{M}\}$ the following inequality holds:

$$||\boldsymbol{p}_{\pi_m} - \boldsymbol{p}_{\pi'_m}|| > 2R + r_{\pi_m} + r_{\pi'_m}$$

which intuitively means that the obstacles m and m' are disjoint in a such a way that the whole team of UVMSs including the object can pass through the free space between them. Therefore, there exists a feasible trajectory $\boldsymbol{x}_O(t)$ for the whole team that connects

the initial configuration $\boldsymbol{x}_O(t_0)$ with $\boldsymbol{x}_O^d$ such as:

$$\mathscr{B}(\boldsymbol{x}_O(t), R) \cap \{\mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m}) \cup \partial\mathscr{W}\} = \emptyset, \ \forall m \in \{1, \ldots, \mathscr{M}\}$$

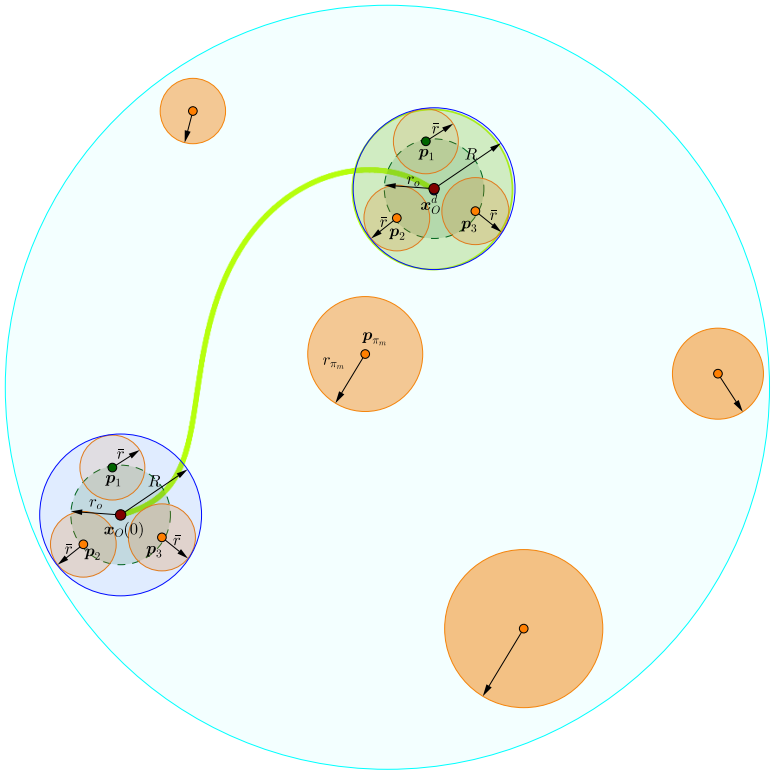


FIGURE 9.2: Graphical representation of a feasible trajectory of the team of UVMS carrying object from the initial position $\boldsymbol{x}_O(t_0)$ to the desired position $\boldsymbol{x}_O^d$. The boundary of workspace $\partial\mathscr{W}$ is illustrated in cyan. Red circles indicate the obstacles within the workspace $\mathscr{W}$. A feasible trajectory of the whole team is depicted in green.

9.3.3 Navigation Function

The desired object trajectory within the workspace $\mathscr{W}$ relies on the Navigation Function concept originally proposed by Rimon and Koditschek in [185] as follows:

$$\phi_O(\boldsymbol{x}_O; \boldsymbol{x}_O^d) = \frac{\gamma(\boldsymbol{x}_O - \boldsymbol{x}_O^d)}{[\gamma^k(\boldsymbol{x}_O - \boldsymbol{x}_O^d) + \beta(\boldsymbol{x}_O)]^{\frac{1}{k}}} \tag{9.25}$$

where $\phi_O : \mathscr{W} - \bigcap_{m=1}^{\mathscr{M}} \mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m}) \longrightarrow [0, 1)$ denotes the potential that derives a safe motion vector field within the free space $\mathscr{W} - \bigcap_{m=1}^{\mathscr{M}} \mathscr{B}(\boldsymbol{p}_{\pi_m}, r_{\pi_m})$. Moreover, $k > 1$ is a design constant, $\gamma(\boldsymbol{x}_O - \boldsymbol{x}_O^d) > 0$ with $\gamma(\boldsymbol{0}) = 0$ represents the attractive potential field to the

goal configuration $\boldsymbol{x}_O^d$ and $\beta(\boldsymbol{x}_O) > 0$ with:

$$\lim_{\boldsymbol{x}_O \to \left\{ \begin{smallmatrix} \text{Boundary} \\ \text{Obstacles} \end{smallmatrix} \right.} \beta(\boldsymbol{x}_O) = 0$$

represents the repulsive potential field by the workspace boundary and the obstacle regions. In that respect, it was proven in [185] that $\phi_O(\boldsymbol{x}_O, \boldsymbol{x}_O^d)$ has a global minimum at $\boldsymbol{x}_O^d$ and no other local minima for sufficiently large k. Thus, a feasible path that leads from any initial obstacle-free configuration[5] to the desired configuration might be generated by following the negated gradient of $\phi_O(\boldsymbol{x}_O, \boldsymbol{x}_O^d)$. Consequently, the object's desired motion profile is designed as follows:

$$\boldsymbol{v}_O^d(t) = -K_{NF} \boldsymbol{J}'_O(\boldsymbol{\eta}_{2,O}) \nabla_{\boldsymbol{x}_O} \phi_O(\boldsymbol{x}_O(t), \boldsymbol{x}_O^d) \tag{9.26}$$

where $K_{NF} > 0$ is a positive gain. Now let us define a sequence of sampling time $\{t_j\}_{j \geq 0}$ with a constant sampling time $h > 0$ with $h < T_p$ for the system such that:

$$t_{j+1} = t_j + h, \ \forall j \geq 0 \tag{9.27}$$

Therefore, since all UVMS $i \in \mathcal{N}$ are aware of both the desired configuration of the object as well as of the obstacles position in the workspace, given a current position of the object $\boldsymbol{x}_O(t_i)$ and $\boldsymbol{v}_O(t_j)$ at the time t_j they can propagate for time interval $s \in [t_j, t_j + T_P]$ where T_P is the prediction horizon, a map of desired trajectory and velocity of the object based on (9.25), (9.26) given as $\boldsymbol{x}_O^d(s)$ and $\boldsymbol{v}_O^d(s), \ s \in [t_j, t_j + T_P]$ which will be used in the subsequent analysis.

9.3.4 Constraints

State Constraints:

In this chapter we assume that the UVMS must be avoid various constraints that can be modified as state constraints of the system. More specifically, joint limits and singularity avoidance should be satisfied by each UVMS. These requirements are captured by the state constraint set X_i of the system, given by:

$$\boldsymbol{x}_i(t) \in X_i \subset \mathbb{R}^{2n_i} \tag{9.28}$$

[5] Except from a set of measure zero[185].

which is formed by the following constraints:

$$\theta_O(t) \in (-\frac{\pi}{2}, \frac{\pi}{2}) \tag{9.29}$$

$$\boldsymbol{q}_i \in \mathbb{R}^{n_i} \backslash \Big(Q_{s_i}(\boldsymbol{q}_i) \cup Q_{l_i}(\boldsymbol{q}_i) \Big),\ i \in \mathscr{N} \tag{9.30}$$

$$|\dot{q}_{k_i}| \leq \bar{\dot{q}}_{k_i}, \forall k \in \{1, \ldots, n\}, i \in \mathscr{N} \tag{9.31}$$

where $Q_{s_i}(\boldsymbol{q}_i)$ is the set of singular position of the system (9.4) and $Q_{l_i}(\boldsymbol{q}_i)$ is the set of manipulator's joint limits defined as:

$$Q_{l_i}(\boldsymbol{q}_i) = \{\boldsymbol{q}_i \in \mathbb{R}^{n_i} : |q_{k_i}| \leq \bar{q}_i\}, \forall k \in \{1, \ldots, n_i\}, i \in \mathscr{N} \tag{9.32}$$

where $\bar{q}_{k_i}$ is the limit bound for the corresponding joint $q_{k_i},\ k \in \{1, \ldots, n\}, i \in \mathscr{N}$. Moreover, $\bar{\dot{q}}_{k_i}$ is the upper value for the joint velocity $\dot{q}_{k_i},\ k \in \{1, \ldots, n\}, i \in \mathscr{N}$. Therefore, the set X_i capture all the state constraints of the systems (9.24), i.e., singularity avoidance as well as joint limits limitations.

Remark 9.2. Notice that collision avoidance between the whole system (UVMS and the object) and obstacles within the workspace (see Fig 9.2) are achieved based on the desired trajectory and velocity of the object calculated from (9.25) and (9.26) as it is explained previously.

Input Constraints:
The actuation of the vehicle body and the manipulator are generated by the thrusters and servo respectively. Hence, the input constraints for τ_{k_i}, $k \in \{1, \ldots, \tau_n\}, i \in \mathscr{N}$, with τ_n to be the number of actuated joints, can be given as:

$$||\boldsymbol{\tau}_i|| \leq \bar{\boldsymbol{\tau}}_i \Leftrightarrow ||\boldsymbol{J}_i(\boldsymbol{q}_i)^\top \boldsymbol{u}_i|| \leq \bar{\boldsymbol{\tau}}_i$$

where $\bar{\boldsymbol{\tau}}_i$ is a vector including corresponding limit bound for each actuated joint τ_{k_i}, $k \in \{1, \ldots, \tau_{n_i}\}, i \in \mathscr{N}$. Therefore, we can define the control input set T_i:

$$\boldsymbol{\tau}_i(t) \in T_i \subset \mathbb{R}^{\tau_{n_i}} \tag{9.33}$$

with:

$$T_i = \{\boldsymbol{\tau}_i \in \mathbb{R}^{\tau_{n_i}} : ||\boldsymbol{J}_i(\boldsymbol{q}_i)^\top \boldsymbol{u}_i|| \leq \bar{\boldsymbol{\tau}}_i,\ \forall \boldsymbol{x}_i \in X_i\}$$

9.3.5 Control design

As it is already mentioned, given the current position and velocity of the object at sampling time j denoted by $\boldsymbol{x}_O(t_j)$ and $\boldsymbol{v}_O(t_j)$ respectively, each UVMS $i \in \mathcal{N}$ for a time interval $s \in [t_j, t_j + T_P]$ where T_P is a prediction horizon and based on (9.25), (9.26) and (9.27), can propagates/calculates a map of desired trajectory and velocity for the object denoted by $\boldsymbol{x}_O^d(s)$ and $\boldsymbol{v}_O^d(s)$ respectively. As will be explained in the sequel, at each sampling time, UVMS $i \in \mathcal{N}$ solves its corresponding part of the dynamics (9.23) via an NMPC scheme subject to its dynamics (9.24) and a number of inequality constraints. More specifically, the control objective for each UVMS $i \in \mathcal{N}$ is to follow these desired trajectory and velocity, while respecting the state constraints (9.29)-(9.31) as well as the input constraints (9.33). In particular, in sampled data NMPC, a Finite Horizon Optimal Control Problem (FHOCP) is solved at discrete sampling time instants t_j based on the current state measurements $\boldsymbol{x}_i(t_j),\ i \in \mathcal{N}$. For UVMS $i,\ i \in \mathcal{N}$, the open-loop input signal applied in between the sampling instants is given by the solution of the following FHOCP:

$$\min_{\hat{\boldsymbol{\tau}}_i(\cdot)} J_i(\boldsymbol{x}(t_j), \hat{\boldsymbol{\tau}}_i(\cdot)) = \tag{9.34}$$

$$\min_{\hat{\boldsymbol{\tau}}_i(\cdot)} \left\{ \int_{t_i}^{t_i+T_p} \Big[F_i\big(\hat{\boldsymbol{x}}_O(s), \hat{\boldsymbol{v}}_O(s), \hat{\boldsymbol{\tau}}_i(s)\big) \Big] ds + E_i\Big(\hat{\boldsymbol{x}}_O(t_j + T_P), \hat{\boldsymbol{v}}_O(t_j + T_P)\Big) \right\}$$

subject to:

$$\dot{\hat{\boldsymbol{x}}}_i(s) = f_i(\hat{\boldsymbol{x}}_i(s), \hat{\boldsymbol{u}}_i(s)), \qquad \hat{\boldsymbol{x}}_i(t_j) = \boldsymbol{x}_i(t_j), \tag{9.35}$$

$$\hat{\boldsymbol{\tau}}_i(s) = \boldsymbol{J}_i^\top(\hat{\boldsymbol{q}}_i)\hat{\boldsymbol{u}}_i + \boldsymbol{\tau}_{i0}(\boldsymbol{q}_i),\ s \in [t_j, t_j + T_P] \tag{9.36}$$

$$\hat{\boldsymbol{x}}_O(s) = \mathscr{F}(\hat{\boldsymbol{q}}_i(s)) - \begin{bmatrix} {}^I\boldsymbol{R}_O \boldsymbol{l}_i \\ \boldsymbol{\alpha}_i \end{bmatrix},\ s \in [t_j, t_j + T_P], \tag{9.37}$$

$$\hat{\boldsymbol{v}}_O(s) = \boldsymbol{J}_{i_O}\boldsymbol{J}_i(\hat{\boldsymbol{q}}_i(s))\dot{\hat{\boldsymbol{q}}}_i(s), \qquad s \in [t_j, t_j + T_P], \tag{9.38}$$

$$\hat{\boldsymbol{x}}_i(s) \in X_i, \qquad s \in [t_j, t_j + T_P], \tag{9.39}$$

$$\hat{\boldsymbol{\tau}}_i(s) \in T_i, \qquad s \in [t_j, t_j + T_P], \tag{9.40}$$

$$\hat{\boldsymbol{x}}(t_j + T_P) \in \mathscr{E}_f \tag{9.41}$$

where $\mathscr{E}_f$ is a terminal region around the desired trajectory profile and F and E are the running and terminal cost function, respectively. In order to distinguish the predicted variables (i.e., internal to the controller) we use the double subscript notation $\hat{(\cdot)}$ corresponding to the system (9.35). This means that $\hat{\boldsymbol{x}}_i(s),\ s \in [t_j, t_j + T_P]$ is the solution of (9.24) based on the measurement of the state at time instance t_j (i.e., $\boldsymbol{x}_i(t_j)$, provided by the on–board navigation system) while applying a trajectory of inputs (i.e., $\hat{\boldsymbol{u}}_i(s),\ s \in [t_j, t_j + T_P]$). Notice that we use this notation in order to account for the

mismatch between the predicted values of the system and the actual closed-loop values. The cost function $F_i(\cdot)$, as well as the terminal cost $E(\cdot)$, are both of quadratic form given as:

$$F_i\big(\hat{\boldsymbol{x}}_O(s), \hat{\boldsymbol{v}}_O(s), \hat{\boldsymbol{\tau}}_f(s)\big) = [\hat{\boldsymbol{x}}_O^\top(s), \hat{\boldsymbol{v}}_O^\top(s)]\boldsymbol{Q}[\hat{\boldsymbol{x}}_O^\top(s), \hat{\boldsymbol{v}}_O^\top(s)]^\top + \hat{\boldsymbol{\tau}}_f^\top(s)\boldsymbol{R}\hat{\boldsymbol{\tau}}_f(s)$$
$$E_i\big(\hat{\boldsymbol{x}}_O, \hat{\boldsymbol{v}}_O\big) = [\hat{\boldsymbol{x}}_O^\top(s), \hat{\boldsymbol{v}}_O^\top(s)]\boldsymbol{P}[\hat{\boldsymbol{x}}_O^\top(s), \hat{\boldsymbol{v}}_O^\top(s)]^\top$$

with $\boldsymbol{P} \in \mathbb{R}^{12\times12}, \boldsymbol{Q} \in \mathbb{R}^{12\times12}$ and $\boldsymbol{R} \in \mathbb{R}^{\tau_{n_i}\times\tau_{n_i}}$ being symmetric and positive definite matrices to be appropriately tuned. The terminal set $\mathscr{E}_f \subset X_i$ is chosen as: $\mathscr{E}_f = \{\boldsymbol{x}_i \in X_i : E_i(\cdot) \leq \epsilon\}$, where ϵ is an arbitrarily and positive small constant to be appropriately tuned. The solution of FHOCP (9.34)-(9.41) at time t_j provides an optimal control input trajectory denoted by $\hat{\boldsymbol{\tau}}_i^*(s; \boldsymbol{x}(t_j)),\ s \in [t_j, t_j + T_P]$. This control input us then applied to the system until the next sampling time t_{j+1}:

$$\boldsymbol{\tau}_i(s; \boldsymbol{x}(t_j)) = \hat{\boldsymbol{\tau}}_i^*(s; \boldsymbol{x}(t_j)),\ s \in [t_j, t_j + h] \tag{9.42}$$

At time $t_{j+1} = t_j + h$ a new finite horizon optimal control problem is solved in the same manner, leading to a receding horizon approach. Notice that the control input $\boldsymbol{\tau}_i(\cdot)$ is of feedback form, since it is recalculated at each sampling instant based on the then-current state. The pseudo-code description of the proposed real-time control scheme for UVMS $i,\ i \in \mathscr{N}$ is given in given in *Algorithm2*:

Algorithm 2 Real time MPC algorithm:

1: **Triggering time** ▷ At time instance t_j UVMS i measures its state vector $\boldsymbol{x}_i$
2: $\boldsymbol{p}_i(t_j) \leftarrow$ eq.(9.2) ▷ calculates its EE pose
3: $\boldsymbol{v}_i(t_j) \leftarrow$ eq.(9.3) ▷ calculates its EE velocity
4: $\boldsymbol{v}_O(t_j), \boldsymbol{x}_O(t_j) \leftarrow$ eq.(9.14) − (9.15) ▷ calculates object pose and velocity
5: $\boldsymbol{x}_O^d(s), \boldsymbol{v}_O^d(s),\ s \in [t_j, t_j + T_P] \leftarrow$ eq.(9.25), (9.26) ▷ propagates for the time interval $s,\ s \in [t_j, t_j + T_P]$ a map of safe/desired trajectory and velocity of the object
6: $\hat{\boldsymbol{\tau}}_i^*(s; \boldsymbol{x}(t_j)),\ s \in [t_j, t_j + T_P] \leftarrow$ FHOCP($\boldsymbol{x}_i(t_j)$) ▷ Run FHOCP of (9.34)-(9.41). The solution is a optimal control input trajectory for the time interval $[t_j, t_j + T_P]$.
7: **for** $s \in [t_j, t_j + h]$ **do**
8: Apply the $\boldsymbol{\tau}_i(s; \boldsymbol{x}(t_j)) = \hat{\boldsymbol{\tau}}_i^*(s; \boldsymbol{x}(t_j))$ to the UVMS.
9: $t_{j+1} = t_j + h$ ▷ The next triggering time
10: **goto** *Triggering.*

9.4 Simulation study

Real-time simulation have been performed to validate the proposed approach. The simulation environment is designed based on UwSim dynamic simulator [262] running on the Robot Operating System (ROS) [215]. We consider a scenario involving 3D motion with

two UVMSs with the same structure, transporting a bar in a constrained workspace with static obstacles (see Fig.9.1). The UVMS model is an AUV equipped with a small 4 DoF manipulator attached at the bow of the vehicle (see Fig.9.1). The dynamic parameters of the vehicle have been identified via a proper identification scheme [263], while the manipulator's parameters as well as object's parameters have been extrapolated by the CAD data. The complete state vector of the vehicle (3D position, orientation, velocity) is available via the sensor fusion and state estimation module given in our previous results [263]. The Constrained NMPC employed in this work is implemented using the NLopt Optimization library [193].

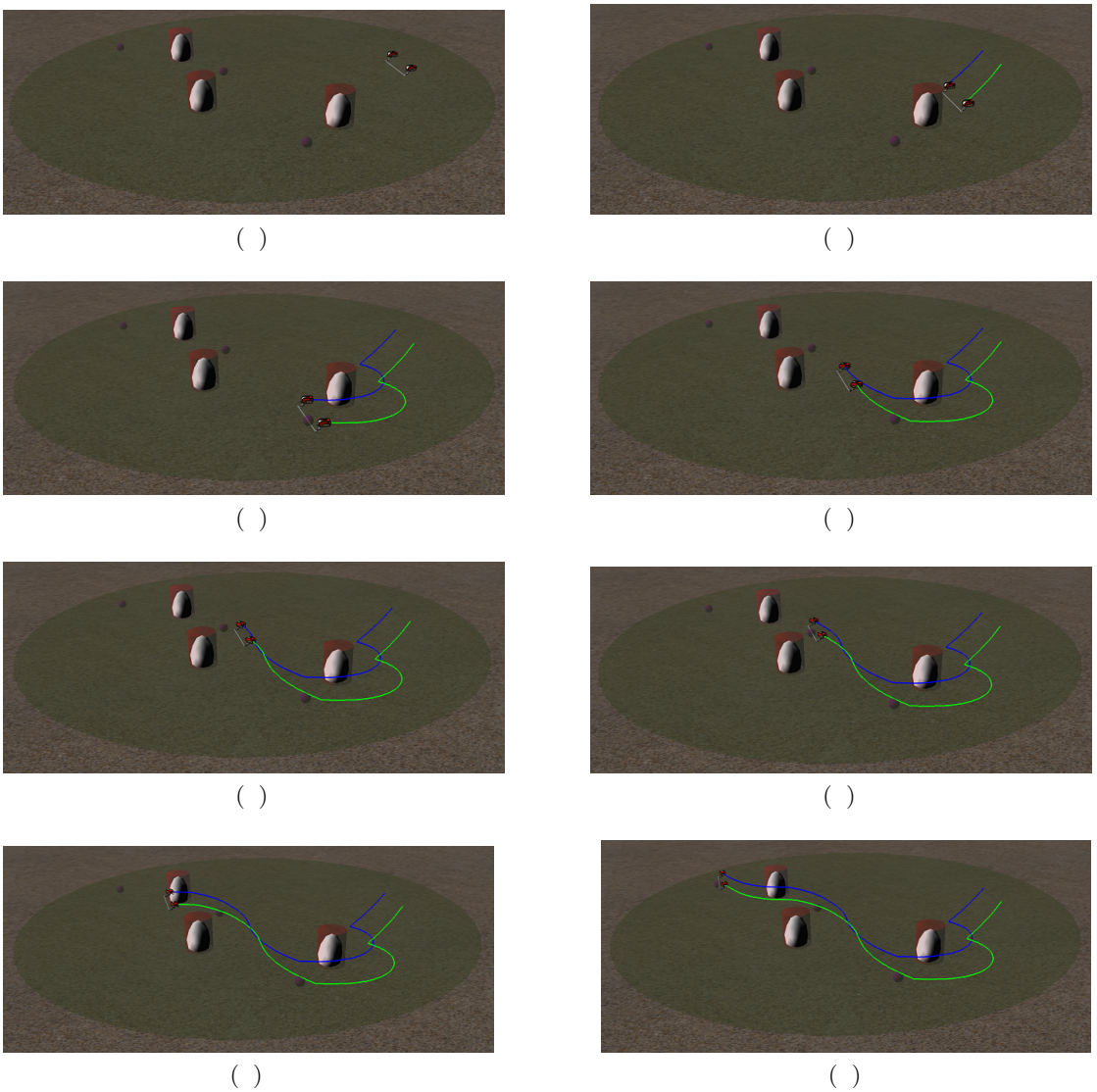


FIGURE 9.3: The evolution of the proposed methodology in 8 consecutive time instants.

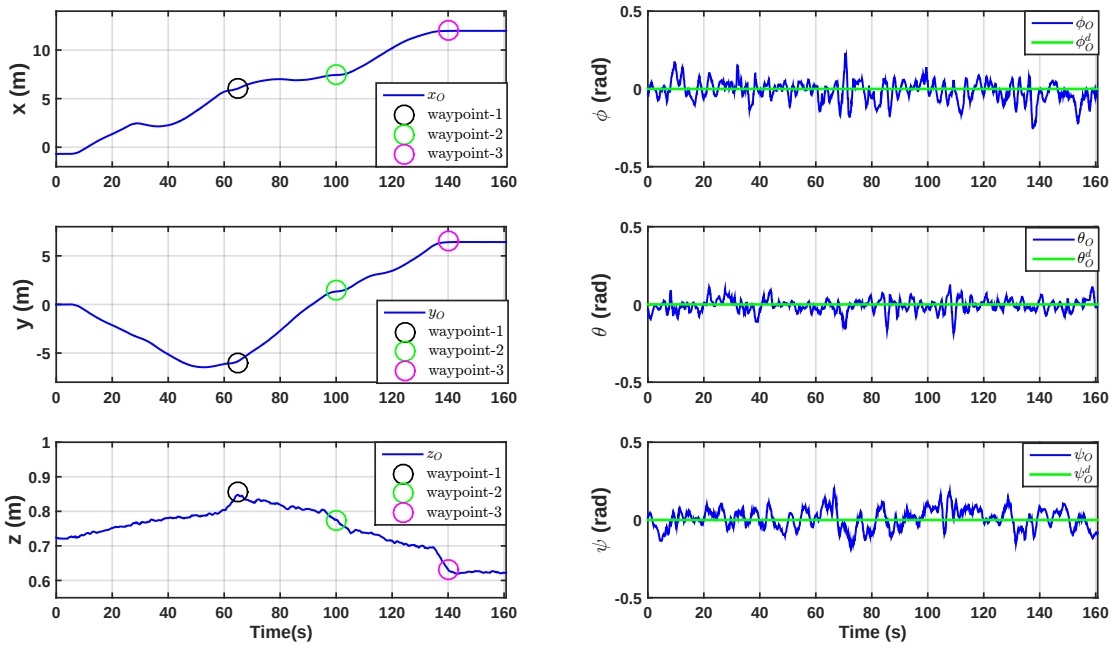


FIGURE 9.4: Object coordinates during the control operation

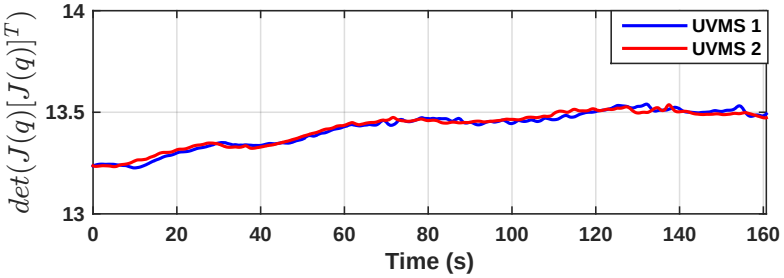


FIGURE 9.5: $det(\boldsymbol{J}(\boldsymbol{q})[\boldsymbol{J}(\boldsymbol{q})]^{\top})$ during the control operation

9.4.1 Simulation results

In the following simulation, the objective for the team of UVMSs is to follow a set of predefined way points, while simultaneously avoid obstacles within the workspace. The position of the obstacles w.r.t the inertial frame $\mathscr{I}$ in $x-y$ plane is given by: $\boldsymbol{x}_{obs_1} = [4,\ -4.5]$, $\boldsymbol{x}_{obs_2} = [9,\ -1.5]$ and $\boldsymbol{x}_{obs_3} = [9,\ 5]$ respectively. These obstacles are cylinders with radius $r_{\pi_i} = 0.6m,\ i = \{1, 2, 3\}$ and are modeled together with the workspace boundaries according to the spherical world representations as consecutive spheres. The radius of the sphere $\mathscr{B}(\boldsymbol{p}_i, \bar{r}),\ i \in \{1, 2\}$ which covers all the UVMS volume (for all possible configurations) is defined as $\bar{r} = 1m$. In this way, the Navigation function (9.25)-(9.26) was designed with gain $K_{NF} = 0.5$. Regarding to constraints (9.31), we consider that the vehicle's velocity most not exceed $0.5m/s$for translation and $0.1rad/s$ for rotational. In the same vein the manipulator joint velocities must be retained between $(-0.1, 0.1)rad/s$. Moreover, the manipulator joint positions (9.32) must be retained between $(-2, 2)rad$. Furthermore, input saturations (9.33) for the vehicle and manipulator are considered

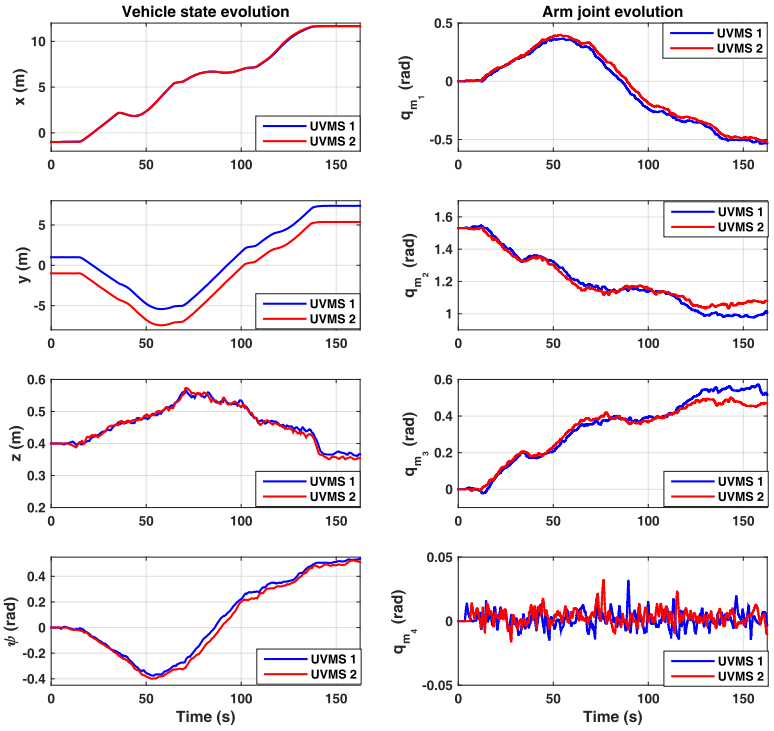


FIGURE 9.6: The evolution of the system states at joint level

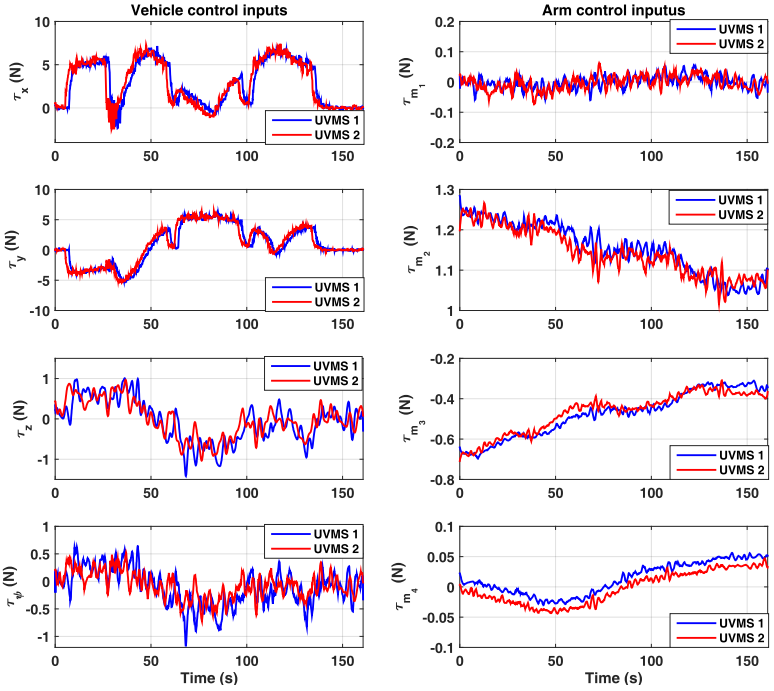


FIGURE 9.7: The control input signals during the control operation

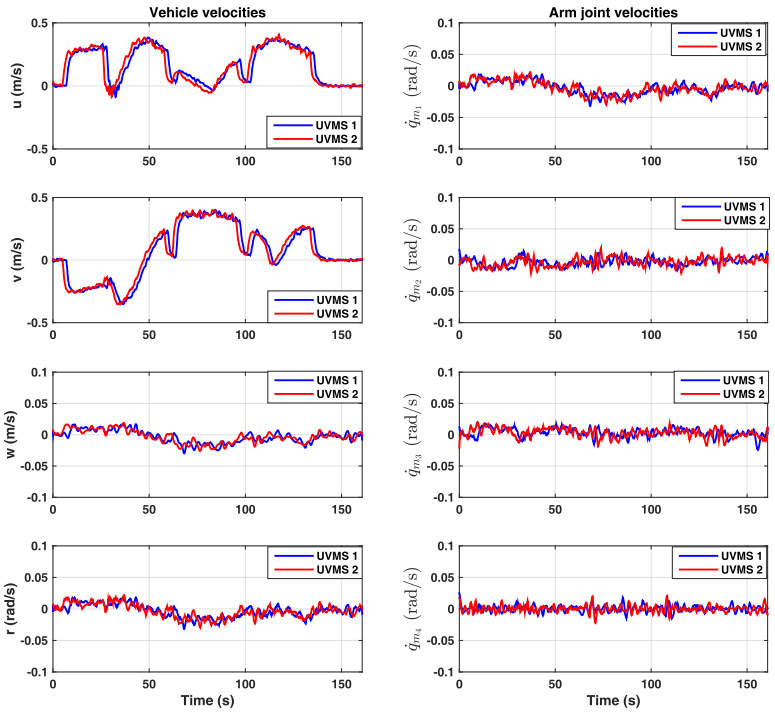


FIGURE 9.8: The evolution of the system velocities at joint level

as: $\bar{\tau}_v = 10N$ and $\bar{\tau}_m = 2N$, respectively. The sampling time (9.27) and the prediction horizon are $h = 0.12sec$ and $T_p = 5 \times h = 0.6sec$ respectively. The matrices $\boldsymbol{P}_x$, $\boldsymbol{Q}_x$, $\boldsymbol{Q}_v$ and $\boldsymbol{R}$ as well as he load sharing coefficients c_1 and c_2 for both UVMSs are equal and set to: $\boldsymbol{P}_x = \boldsymbol{Q}_x = 0.8 \cdot \boldsymbol{I}_{6\times6}$, $\boldsymbol{R} = 0.3 \cdot \boldsymbol{I}_{8\times8}$, $\boldsymbol{Q}_v = 0.4 \cdot \boldsymbol{I}_{6\times6}$, and $c_1 = c_2 = 0.5$. The initial position of the object is $\boldsymbol{x}_O = [-0.7,\ 0,\ 0.72,\ 0.04,\ -0.07,\ 0]$. We set 3 waypoints as $\boldsymbol{x}^d_{O_1} = [6,\ -6,\ 0.85,\ 0,\ 0,\ 0]$, $\boldsymbol{x}^d_{O_2} = [7.5,\ 1.5,\ 0.78,\ 0,\ 0,\ 0]$ and $\boldsymbol{x}^d_{O_3} = [12,\ 6.5,\ 0.65,\ 0,\ 0,\ 0]$ which make the mission more challenging considering the obstacles' positions within the workspace (See Fig.9.3 and Fig.9.1). The results are presented in Fig.9.3-Fig.9.5. The trajectory of the system within the workspace as well as object coordinates evolution are depicted in Fig. 9.3 and Fig.9.4 respectively. It can be seen that the UVMSs have successfully transported cooperatively the object and have followed the set of predefined way points while safely avoids the obstacles. The evolution of $det(\boldsymbol{J}(\boldsymbol{q})[\boldsymbol{J}(\boldsymbol{q})]^\top)$ (see (9.4) and (9.30)) during the operation is given in Fig.9.5. It can be easily seen that value remained positive during the cooperative manipulation task. Moreover, the evolution of the system velocity and its states at joints level as well as the corresponding control inputs are indicated in Fig.9.8, Fig. 9.6 and Fig.9.7 respectively. As it was expected from the theoretical findings, these values were retained in the corresponding feasible regions defined by the corresponding upper bounds and consequently all of the system constraints were satisfied.

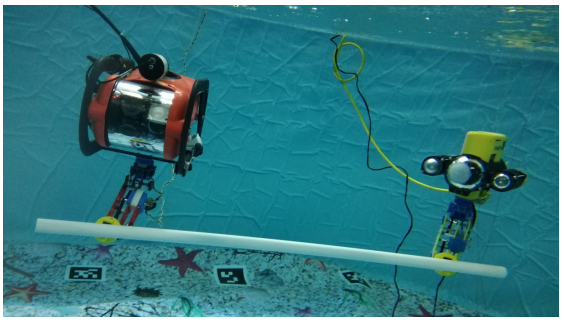


FIGURE 9.9: Experimental study: custom made UVMSs under cooperative transportation.

9.5 Experimental results

In order to verify the theoretical results of this chapter, we performed a set of real-time experiments. The team of UVMS consists of two small ROV equipped with our custom made water proof manipulators (see Fig.9.9). Notice that the Videoray ROV is under non-holonomic constraint in Sway direction. The objective for the team of UVMS is to stabilize and follow a set of predefined way point in our small test tank. More specifically, for the case of stabilization scenario, the objective is to transfer and to stabilize the object into desired position $\boldsymbol{x}_O^d = [0,\ 0,\ 0.4,\ 0,\ 0,\ 1.57]$. The results are presented in Fig.9.10-Fig.9.14. The trajectory of the system within the workspace as well as object coordinates evolution are depicted in Fig. 9.10 and Fig.9.4 respectively. It can be seen that the UVMSs have successfully transported cooperatively and stabilized the object. The evolution of the system velocity and its states at joints level as well as the corresponding control inputs are indicated in Fig.9.14, Fig. 9.12 and Fig.9.13 respectively. In case of way point tracking scenario, the objective is to follow four predefined way points inside the test tank as it can be observed in Fig. 9.15. Moreover, the evolution of the system states at joints level as well as the corresponding control inputs are indicated in Fig. 9.16 and Fig.9.13 respectively. As it was expected from the theoretical findings, in both of the considered scenarios, the all of the system values were retained in the corresponding feasible regions defined by the corresponding upper bounds and consequently all of the system constraints were satisfied.

9.5.1 Video

A video demonstrating the aforementioned real time simulation and the experimental results of the proposed methodology can be found in a HD video at the following urls:
simulation: https://youtu.be/O5WHov7EGMI
experiment: https://youtu.be/noRZNsjX6N0.
Alternatively, they can be found in the attached dvd as file: video_ch9a.mpg and video_ch9b.mpg respectively.

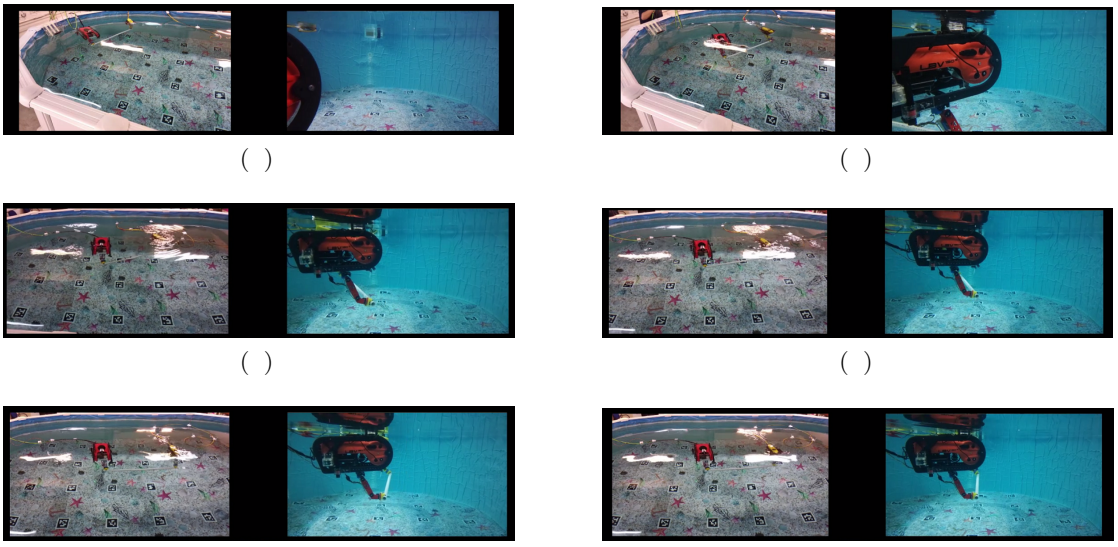


FIGURE 9.10: Experimental study - stabilization scenario: The evolution of the proposed methodology in 6 consecutive time instants.

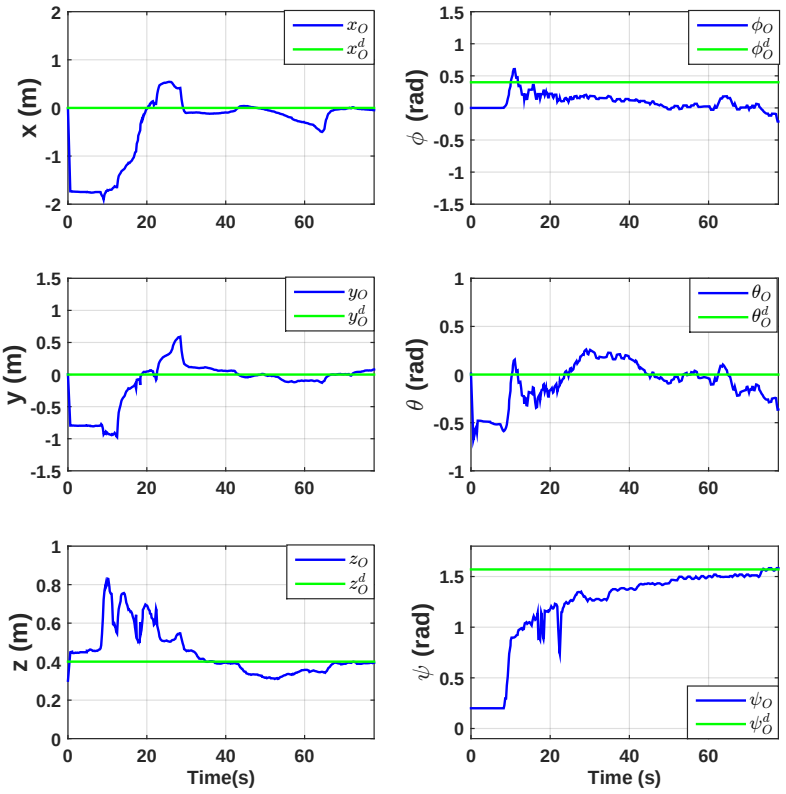


FIGURE 9.11: Experimental study - stabilization scenario: Object coordinates during the control operation

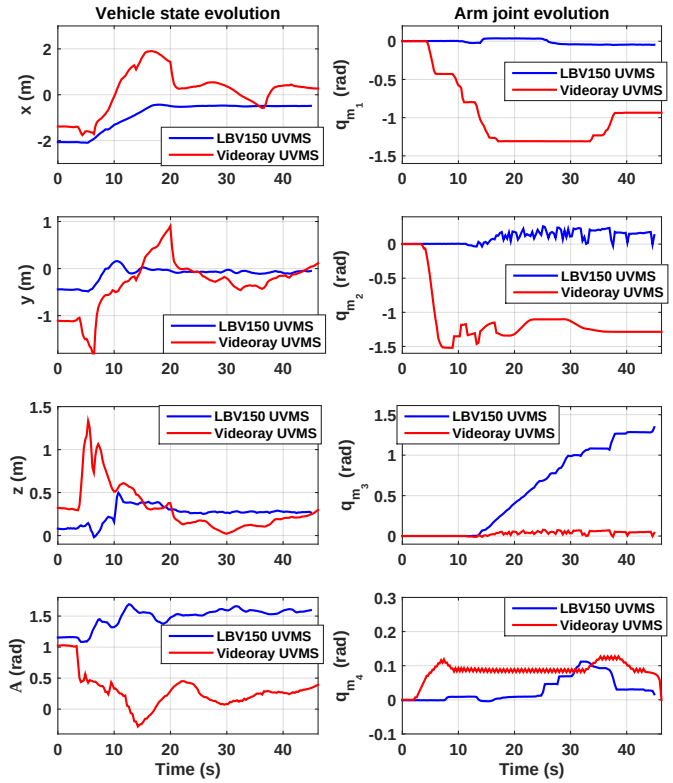


FIGURE 9.12: Experimental study - stabilization scenario: The evolution of the system states at joint level

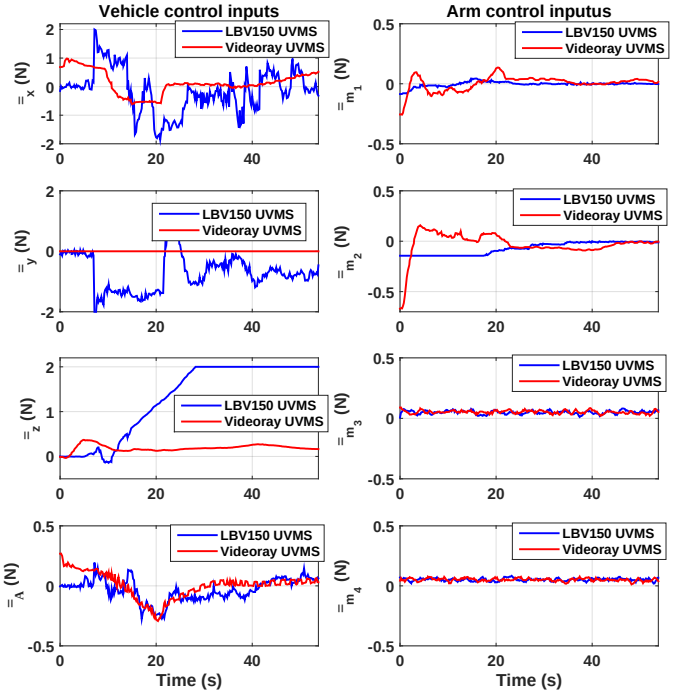


FIGURE 9.13: Experimental study - stabilization scenario: The control input signals during the control operation

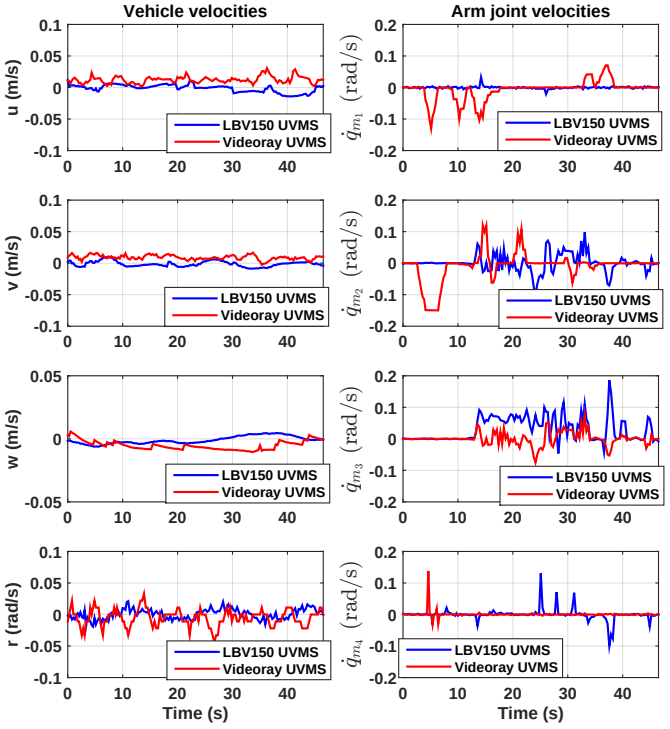


FIGURE 9.14: Experimental study- stabilization scenario: The evolution of the system velocities at joint level

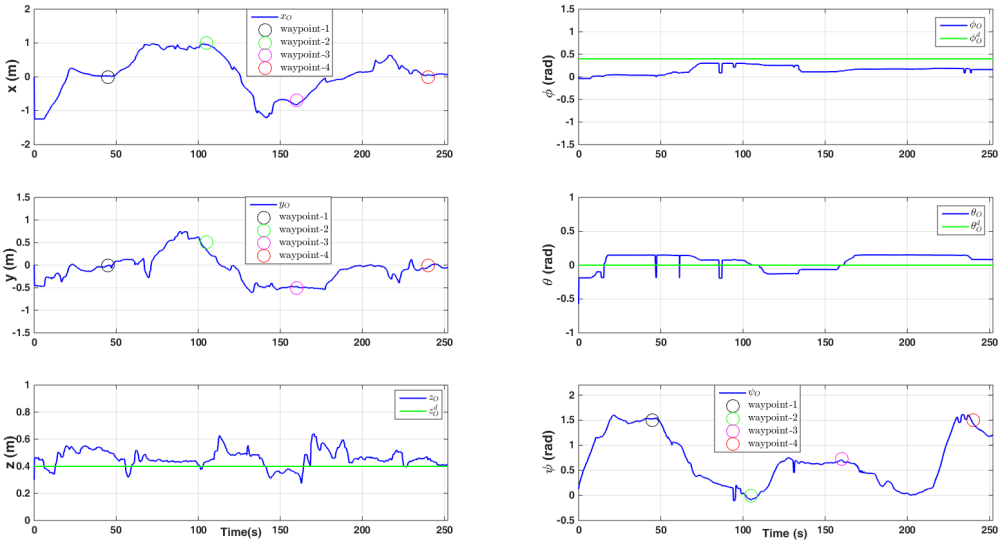


FIGURE 9.15: Experimental study - Waypoint tracking scenario: Object coordinates during the control operation

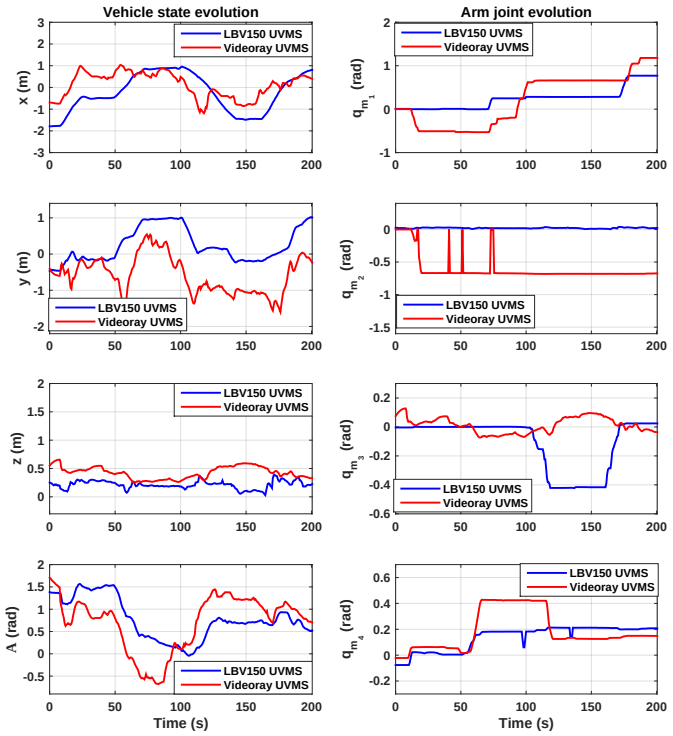


FIGURE 9.16: Experimental study - Waypoint tracking scenario: The evolution of the system states at joint level

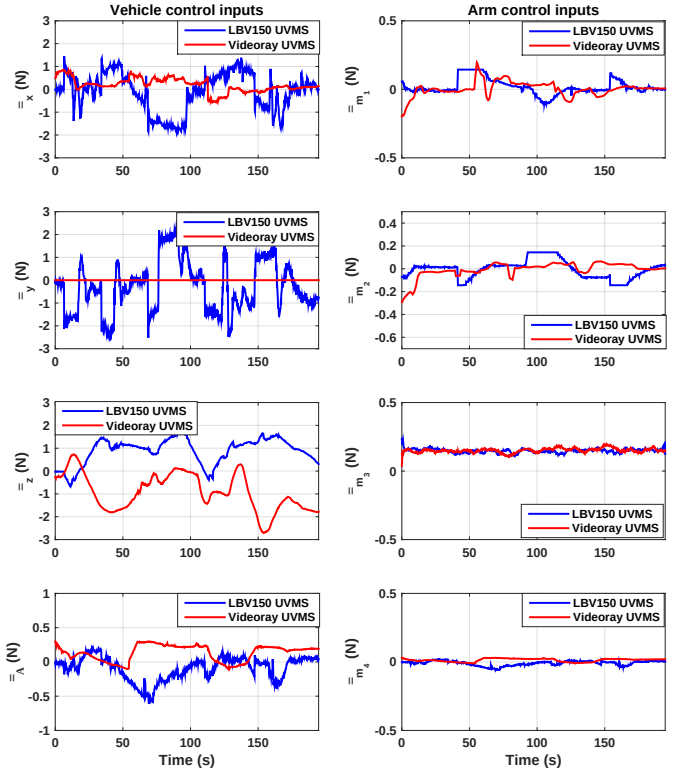


FIGURE 9.17: Experimental study - Waypoint tracking scenario: The control input signals during the control operation

9.6 Summary and Future Work

In this chapter we presented a novel decentralized object transportation control scheme for a team UVMSs in a constrained workspace with static obstacles. The purpose of this decentralized control scheme is to navigate cooperatively the object from an initial position to a desired one, while satisfying various limitation such as: obstacles, joint limits, control input saturation as well as kinematic and representation singularities. Thank to the novel formulation of the problem, the proposed control strategy relieves the team of robots from inter-robot communication during the execution of the collaborative tasks. This, consequently, increases significantly the robustness of the cooperative scheme and furthermore avoids any restrictions imposed by the acoustic communication bandwidth (e.g., the number of participating UVMSs). Moreover, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities.

Part VI - Summary and Future Research Directions

Chapter 10

Summary and Future Research Directions

10.1 Summary

In this Dissertation, we focused on motion, visual servoing, cooperative&interaction control methodologies for single and multiple Underwater Vehicle Manipulator Systems (UVMSs) considering significant issues such as: external disturbances, limited power resources, strict communication constraints along with underwater sensing and localization issues. More precisely, this thesis was divided into three main parts: i) Motion Control consists of Chapter 3, ii) Visual servoing consists of Chapters 4, 5 and 6 and iii) Interaction&Cooperative Transportation consists of Chapters 7, 8 and 9. In each part, we formulate and present control methodologies in order to solve the corresponding problems and aim to achieve and satisfy the aforementioned limitation introduced from the nature of underwater environment.

More precisely, regarding the first part, we assumed that the considered underwater robotic vehicles must be able to travel from an initial position to a desired one (i.e., relatively close to the point on interest) satisfying aforementioned limitations of the underwater environment. In this way, we presented a Model predictive Control strategy for underwater robotic vehicles operating in a constrained workspace including obstacles. The purpose of this control scheme is to guide the vehicle towards specific way points. Various constraints such as: energy consumption issues, obstacles, workspace boundaries, thruster saturation and predefined upper bound of the vehicle velocity (requirements for various underwater tasks such as seabed inspection, mosaicking etc.) are considered during the control design. We are currently working towards extending the proposed methodology for multiple underwater vehicles operating in a dynamic environment including not only static but also moving obstacles. A disadvantage of the proposed control scheme is the bigger CPU effort requirement relative to he case where the dynamic model of the vehicle in which the ocean currents are not incorporated (See Section 3.5.2.3). Thus, future research directions include computational burden relaxation of the proposed scheme. A possible extension could be a framework towards Event-Triggered NMPC such that the control inputs

are updated only when it is necessary. In other word, a feedback controller that computes the control input and a triggering mechanism that determines when the new control update should occur. This will be results in the reduction of the computational effort.

When the robots have already arrived in a region close to the point of interest (e.g., the object), they should be able to reach very close in order to grasp it or to do some interaction task. Thus, we considered visual feedback (i.e., imaging sonar or usual camera) as an appropriate feedback for designing of efficient controllers. This is motivated by the delay and inaccuracy of the measurements provided by common underwater localization systems e.g., Doppler Velocity Log (DVL) or Ultra Short Baseline (USBL), which make them inappropriate for achieving this task. In this way, we formulated a number of novel visual servoing control strategies (i.e., Image based and Position based) in order to reach the robot's end effector close to or stabilize to the point of interest.

More precisely, in Chapter 4, a Self-triggered PBVS scheme based on NMPC was presented for underwater robotic vehicles for case when the relative position between the robot and the point of interest is available online (e.g., by detection of a known marker on the object). The purpose of the controller is to navigate and stabilize the vehicle towards a visual target and assuring that the target will always remain inside the camera's field of view. In Chapter 5 we extended that work to an IBVS scheme for case that the relative position is not available online. Both of the aforementioned schemes are combined with a self-triggering mechanism that decides when the visual tracking algorithm needs to be triggered and new control signals must be calculated. Thus, the proposed schemes result in reduction of the CPU computational effort, energy consumption and increases the autonomy of the system. Furthermore, for case when neither the camera calibration matrix of the supposed vision system is available in priory, a model free IBVS control strategy based is proposed which satisfies the visibility constraints. Studying the effect of camera calibration and depth distribution errors to obtain the magnitude of the robust initialization domain is left open for future research. In the same spirit, considering robot kinematic constraints (e.g., joint limits in case of manipulator, non-holonomic constraints, workspace limitations) on the camera motion model is a promising direction that would increase the applicability of the proposed IBVS scheme.

Regarding the third part, we considered two separate scenarios, namely i) Interaction and ii) Cooperative Transportation. Regarding the interaction scenario, we presented in Chapter 7 a robust force/position control scheme for a UVMS in compliant contact with the environment, with great applications in underwater robotics (e.g. sampling of the sea organisms, underwater welding, object handling). The proposed control scheme does not required any a priori knowledge of the UVMS dynamical parameters or the stiffness model. It guarantees a predefined behavior in terms of desired overshoot, transient and steady state response and it is robust with respect to external disturbances and measurement noises. Moreover, the proposed controller exhibits the following important characteristics: i) it is of low complexity and thus can be easily used in most UVMSs ii) the performance of the proposed scheme (e.g., desired overshoot, steady state response) is a priori and explicitly imposed by certain designer-specified performance functions, and is fully decoupled by the control gains selection, thus simplifying the control design.

Regarding to cooperative transportation scenario, first, for case when the robots are equipped with appropriate force/torque sensors at its end effector we proposed a decentralized impedance control scheme with the coordination relying solely on implicit communication arising from the physical interaction of the robots with the commonly grasped object. More precisely, in Chapter 8, we presented a cooperative impedance control for multiple underwater vehicle manipulator systems under implicit communication, avoiding thus completely tedious explicit data transmission. In the proposed scheme, only the leading UVMS is aware of the desired configuration of the object and the obstacles' position in the workspace, and aims at navigating the overall formation towards the goal configuration by trying to achieve the desired tracking behavior via an impedance control law, while avoiding collisions with the static obstacles. On the contrary, the followers adopt a prescribed performance estimation technique in order to estimate the object's desired trajectory and implement a similar impedance control law achieving in this way tracking of the desired trajectory despite the uncertainty and external disturbance in the object and the UVMS dynamics respectively. Each following UVMS employs the proposed estimator based on its own local measurements, thus the estimated desired trajectory of the object for each following UVMS is relative to its own inertial frame. In this way, the whole team mutually agrees on a desired trajectory of the commonly grasped object. Moreover, contrary to the existing work in the related literature, the proposed scheme imposes no restrictions on the underwater communication bandwidth. Furthermore, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities. Additionally, the feedback relies on each UVMS's force/-torque measurements and no explicit data is exchanged online among the robots, thus reducing the required communication bandwidth and increasing robustness. Future research efforts will be devoted towards extending the proposed methodology for multiple UVMSs with underactuated vehicle dynamics.

Moreover, for case when the robots are not equipped with force/torque sensor at it end effector, we have proposed a decentralized predictive control approach which takes into account constraints that emanate from control input saturation as well kinematic and representation singularities. More precisely, in Chapter 9 we presented a novel decentralized object transportation control scheme for a team UVMSs in a constrained workspace with static obstacles. The purpose of this decentralized control scheme is to navigate cooperatively the object from an initial position to a desired one, while satisfying various limitation such as: obstacles, joint limits, control input saturation as well as kinematic and representation singularities. The proposed control strategy relieves the team of robots from intense inter-robot communication during the execution of the collaborative tasks. This, consequently, increases significantly the robustness of the cooperative scheme and furthermore avoids any restrictions imposed by the acoustic communication bandwidth (e.g., the number of participating UVMSs). Moreover, the control scheme adopts load sharing among the UVMSs according to their specific payload capabilities. Future research efforts will be devoted towards extension of the proposed methodology to a Leader-follower architecture under implicit communication. In the same spirit, considering non-holonomic constraints on the UVMS model is a promising direction that would increase the applicability of the proposed control scheme.

10.2 Major Contributions

Considering the aforementioned summaries (see Section 10.1), the major contributions of this Ph.D. thesis can be categorized in the following areas:

1. **AUV and UVMS Modelling and Simulation**
   - Synthesis and modification of the existing related literature is performed, leading to the kinematic and dynamic equations of a robotic underwater vehicle and the UVMS that despite their simple form are accurate and can be easily integrated into the embedded system framework.
   - A dynamic simulation environment built in MATLAB®, following the derived kinematic and dynamic equations.
2. **Motion Control**
   - Model predictive control strategy for underwater robotic vehicles operating in a constrained workspace including obstacles:
     - The purpose of the controller is to guide the vehicle towards specific waypoints while avoid the obstacle within the workspace.
     - Satisfying various constraints such as: obstacles, thruster saturation and predefined upper bound of the vehicle velocity are considered during control design.
     - Exploitation of the ocean currents, when these are in favor of the way-point tracking mission, resulting in reduced energy consumption by the thrusters.
3. **Visual Servoing Control**
   - Self-triggered position based visual servoing:
     - Designed for case when the relative position between the robot and the point of interest is available real-time (e.g., by detection of a known marker on the object)
     - The purpose of the controller is to navigate and stabilize the vehicle towards a visual target and assuring that the target will always remain inside the camera's field of view.
     - The visual servoing controller is combined with a self triggered mechanism that decides when the Visual Tracking Algorithm (VTA) needs to be triggered and when new control inputs must be calculated. This results to a significant smaller number of requested measurements from the vision system, as well as less frequent computations of the control law. This way, the processing effort are reduced and the accuracy and autonomy of the robotic system are increased.
   - Self-triggered image based visual servoing:
     - Designed for cases where the relative position between the robot system and the object is not available real time. Instead, we assume that a desired image view of the object is available and thus the control objective is to drive and stabilize the current image view to the desired image view.

  - Satisfying field of view constraints and control input saturations.
  - Combination of the self triggered mechanism that decides when the visual tracking algorithm needs to be triggered and new control inputs must be calculated. This results in the reduction of the computational effort, energy consumption and increases in this way the autonomy rate of the system.

- Model-free image based visual servoing:
  - Designed for case when the camera parameters of the vision system are not available ahead of time.
  - The proposed scheme imposes prescribed transient and steady state response on the image feature coordinate errors and satisfies the visibility constraints that inherently arise owing to the camera's limited field of view, despite the inevitable calibration and depth measurement errors.
  - Prescribed performance control was included to image based visual servoing scheme, guaranteeing that the image errors remain bounded by preselected performance functions, despite the inevitable calibration and depth measurement errors.
  - The priori guaranteed performance that is imposed by certain designer-specified performance functions, simplifies significantly the selection of the controller parameters.
  - It is of low complexity and thus it can be easily integrated on embedded control systems.

4. **Interaction & Cooperative Control**

- Robust force/position control scheme for an UVMS in compliant contact with the environment:
  - A force/position tracking control protocol for an Underwater Vehicle Manipulator System (UVMS) in compliant contact with the environment, with great applications in underwater robotics (e.g. sampling of the sea organisms, underwater welding, object handling).
  - The proposed control scheme does not required any a priori knowledge of the UVMS dynamical parameters or the stiffness model. In other words, the proposed scheme is is model-free, meaning that it can be implemented in various UVMS with unknown model parameters or model uncertainties.
  - It guarantees a predefined behavior in terms of desired overshoot, transient and steady state response and it is robust with respect to external disturbances and measurement noises.
  - The performance of the proposed scheme (e.g., desired overshoot, steady state response) is a priori and explicitly imposed by certain designer-specified performance functions, and is fully decoupled from the control gain selection and the initial conditions.
  - The derived algorithms were less-complex (no hard calculations and minimal informations were needed), compared to other force/position techniques, thus

its implementation to real underwater vehicle manipulator systems is straightforward.

- Cooperative impedance control for multiple underwater vehicle manipulator systems under lean communication:
  - Cooperative object transportation for a team of UVMSs in a constrained workspace involving static obstacles.
  - Implicit communication: The feedback relies on each UVMS's force/torque measurements, thus no explicit data is exchanged online among the robots, thus reducing the required communication bandwidth and increasing robustness.
  - Distributed leader-follower architecture: i) the leading UVMS: aware of the object's desired configuration ii) following UVMSs: estimate the desired trajectory in a distributed fashion.
  - Parametric uncertainty of the UVMS dynamics as well as external disturbances were considered during control design. More specifically, we design adaptive control laws in order to compensate for the parametric uncertainty of the UVMSs dynamics as well as the external disturbances.
  - The following UVMS's estimation is based on their own local measurements, thus the estimated desired trajectory of the object for each following UVMS is relative to its own inertial frame avoiding thus the need for an accurate common localization system that is extremely challenging and prone to errors in underwater environments.
  - Impedance control law for both the leader and the followers.
  - The proposed control scheme adopts load sharing among the UVMSs according to their specific payload capabilities.
- A Decentralized predictive control approach for cooperative transportation by multiple underwater vehicle manipulator systems:
  - A Nonlinear Model Predictive Control (NMPC) approach for a team of UVMS in order to transport an object in a constrained workspace, while avoiding significant constraints and limitations such as: kinematic and representation singularities, obstacles within the workspace, joint limits and control input saturations.
  - Thanks to the novel formulation of the problem, the proposed scheme avoids the need for existence of force/torque sensors on the robots.
  - Very important properties concerning the robotic manipulator systems such as: singular kinematic configurations of Jacobian matrix and joint limits have been considered during the control design.
  - The proposed control scheme adopts load sharing among the UVMSs according to their specific payload capabilities.
  - The feedback relies on each UVMS's locally measurements and no explicit data is exchanged online among the robots, thus reducing the required communication bandwidth and increasing robustness.

  - To the best of my knowledge, the experimental studies presented in this chapter are the first experimental results on underwater cooperative object transportation all over the world.

10.3 Future Research Directions

Considering the aforementioned summaries (see Section 10.1), a wide range of future work and extensions can be given as:

- *Motion Control*:
  - A disadvantage of the proposed control scheme is the bigger CPU effort requirement relative to he case where the dynamic model of the vehicle in which the ocean currents are not incorporated (See Section 3.5.2.3). Thus, future research directions include computational burden relaxation of the proposed scheme.
  - Extending the proposed methodology for multiple underwater vehicles operating in a dynamic environment including not only static but also moving obstacles (e.e.g, other agents are considered as moving obstacles) is another direction that should be considered.
  - The extending of the proposed model based (i.e., the NMPC uses the model of AUV as well as the model of sea currents) to a learning based NMPC can be a very interesting research direction.
- *Visual servoing*:
  - Considering robot kinematic constraints (e.g., joint limits in case of manipulator, non-holonomic constraints, workspace limitations, singularities) on the camera motion model is a promising direction that would increase the applicability of the proposed image based visual servoing scheme.
  - Studying the effect of camera calibration and depth distribution errors to obtain the magnitude of the robust initialization domain is left open for future research.
  - Formation control at the image plane (i.e., based on image features position on the image plane) while satisfying a synchronized motion on the $3D$ for all of the agents involved on the formation task, can be an interesting direction for the future research.
- *Interaction& Cooperative Manipulation*:
  - Regarding the proposed interaction control scheme, a systematic methodology of selecting the control gains and parameters of the prescribed performance force/position control scheme should be provided. Furthermore, a further analysis on the effect of the limited capabilities of the actuators, could help during the selection of the corresponding performance specifications.

– Regarding the proposed cooperative impedance control scheme for cooperative manipulation by multiple UVMS, future research efforts will be devoted towards extending the proposed methodology for multiple UVMSs with underactuated vehicle dynamics. Moreover, UVMS locally system's constraints (e.g., joint limits, singularities representation, input constraints) should be considered as future direction. Furthermore, in order to describe the workspace, we considered the whole team as a (potentially huge) sphere. This is fine for large scale object transportation. But this might not be adequate near structures because although there might not be a solution for the (huge) ball, in reality, we might have a solution. Something that could be considered as a possible future research direction.

– Regarding the proposed predictive control approach for cooperative manipulation by multiple UVMS, the future research efforts will be devoted towards extension of the proposed methodology to a Leader-follower architecture under implicit communication. In the same spirit, considering robustness analysis regarding external disturbances as well as model uncertainties is a promising direction that would increase the applicability of the proposed control scheme. Moreover, as in the previous case, workspace description's renewal, could be a possible future research direction.

Part VII - Appendices

Chapter 11

Appendices

11.1 Chapter 3–Appendix I

In view of the Property-3.1 and using triangle inequality we get:

$$
\begin{aligned}
&||\boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}(k+1|k)|| = \\
&||f(\boldsymbol{x}_k, \boldsymbol{\tau}_k) + \boldsymbol{w}_k - f(\hat{\boldsymbol{x}}(k|k), \boldsymbol{\tau}_k)|| = ||\boldsymbol{w}_k|| \leq \bar{w} \\
&||\boldsymbol{x}_{k+2} - \hat{\boldsymbol{x}}(k+2|k)|| = \\
&||f(\boldsymbol{x}_{k+1}, \boldsymbol{\tau}_{k+1}) + \boldsymbol{w}_{k+1} - f(\hat{\boldsymbol{x}}(k+1|k), \boldsymbol{\tau}_{k+1})|| \\
&\leq ||f(\boldsymbol{x}_{k+1}, \boldsymbol{\tau}_{k+1}) - f(\hat{\boldsymbol{x}}(k+1|k), \boldsymbol{\tau}_{k+1})|| + ||\boldsymbol{w}_{k+1}|| \\
&\leq L_f ||\boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}(k+1|k)|| + ||\boldsymbol{w}_{k+1}|| \leq (1 + L_f)\bar{w} \\
&\qquad\qquad \vdots \\
&||\boldsymbol{x}_{k+j} - \hat{\boldsymbol{x}}(k+j|k)|| \leq \sum_{i=0}^{j-1} (L_f)^i \bar{w}
\end{aligned}
$$

11.2 Chapter 3–Appendix II

In order to prove this, first it will be shown that:

$$
\begin{aligned}
&||\hat{\boldsymbol{x}}(k|k) - \hat{\boldsymbol{x}}(k|k-1)|| = \boldsymbol{w}_k \leq \bar{w} \\
&||\hat{\boldsymbol{x}}(k+1|k) - \hat{\boldsymbol{x}}(k+1|k-1)|| = ||f(\hat{\boldsymbol{x}}(k|k)) - f(\hat{\boldsymbol{x}}(k|k-1))|| \\
&\leq L_f(||\hat{\boldsymbol{x}}(k|k) - \hat{\boldsymbol{x}}(k|k-1)||) \leq L_f \cdot \bar{w} \\
&\qquad\qquad \vdots \\
&||\hat{\boldsymbol{x}}(k+N-1|k) - \hat{\boldsymbol{x}}(k+N-1|k-1)|| \leq L_f^{N-1} \cdot \bar{w}
\end{aligned}
$$

From *Assumption* 3.3 we have:

$$E(\hat{\boldsymbol{x}}(k+N-1|k)) - E(\hat{\boldsymbol{x}}(k+N-1|k-1)) \leq$$
$$\leq L_E||\hat{\boldsymbol{x}}(k+N-1|k) - \hat{\boldsymbol{x}}(k+N-1|k-1)|| \leq L_E \cdot L_f^{N-1} \cdot \bar{w}$$

For the nominal system and based on the optimal solution $\boldsymbol{\tau}^*(k+j|k-1)$ we have: $E(\hat{\boldsymbol{x}}(k+N-1|k-1)) \in \mathscr{E}_f$. Therefore, taking into account *Assumption* 3.4:

$$E(\hat{\boldsymbol{x}}(k+N-1|k)) \leq \alpha_{\varepsilon_f} + L_E \cdot L_f^{N-1} \cdot \bar{w}$$

we want $E(\hat{\boldsymbol{x}}(k+N-1|k))$ to belong to the set $\mathscr{E}$, thus from *Assumption* 3.4 it must satisfy $E(\hat{\boldsymbol{x}}(k+N-1|k)) \leq \alpha_\varepsilon$, which leads to:

$$E(\hat{\boldsymbol{x}}(k+N-1|k)) \leq \alpha_{\varepsilon_f} + L_E \cdot C_f^{N-1} \cdot \bar{w} \leq \alpha_\varepsilon$$

Therefore, if the uncertainties of the system are bounded by $\bar{w} \leq \frac{\alpha_\varepsilon - \alpha_{\varepsilon_f}}{L_E \cdot L_f^{N-1}}$ then $E(\hat{\boldsymbol{x}}(k+N-1|k))$ belongs to the set $\mathscr{E}$, and from *Assumption* 3.4 we get $E(\hat{\boldsymbol{x}}(k+N|k)) \in \mathscr{E}_f$, which consequently means that $\hat{\boldsymbol{x}}(k_i+N|k_i) \in \mathscr{E}_f$.

11.3 Chapter 3–Appendix III

The difference between the optimal cost and the feasible cost is:

$$\begin{aligned}
&\Delta J = \tilde{J}_N(k) - J_N^*(k-1) = \\
&= \sum_{i=0}^{i=N-1} F(\tilde{\boldsymbol{x}}(k+i|k), \tilde{\boldsymbol{\tau}}(k+i|k)) + E(\tilde{\boldsymbol{x}}(k+N|k)) \\
&- \sum_{i=0}^{i=N-1} F(\hat{\boldsymbol{x}}(k+i-1|k-1), \boldsymbol{\tau}^*(k+i-1|k-1)) \\
&- E(\hat{\boldsymbol{x}}(k+N-1|k-1)) = \sum_{i=0}^{i=N-2} F(\tilde{\boldsymbol{x}}(k+i|k), \tilde{\boldsymbol{\tau}}(k+i|k)) \\
&- F(\hat{\boldsymbol{x}}(k+i|k-1), \boldsymbol{\tau}^*(k+i|k-1)) + F(\tilde{\boldsymbol{x}}(k+N-1|k), \tilde{\boldsymbol{\tau}}(k+N-1|k)) \\
&- F(\hat{\boldsymbol{x}}(k-1|k-1), \boldsymbol{\tau}^*(k-1|k-1)) + E(\tilde{\boldsymbol{x}}(k+N|k)) - E(\hat{\boldsymbol{x}}(k+N-1|k-1))
\end{aligned}$$

where $\tilde{\boldsymbol{\tau}}(k+N-1|k) = h(\hat{\boldsymbol{x}}(k+N-1|k))$ taken from (3.21) and $\tilde{\boldsymbol{\tau}}(k+i|k) = \boldsymbol{\tau}^*(k+i|k-1)$ for $i = 0, \ldots, N-2$. Also from *Lemma* 3.2 combined with *Lemma* 3.1 we get:

$$\sum_{i=0}^{i=N-2} F(\tilde{\boldsymbol{x}}(k+i|k), \tilde{\boldsymbol{\tau}}(k+i|k)) - F(\hat{\boldsymbol{x}}(k+i|k-1), \boldsymbol{\tau}^*(k+i|k-1)) \leq L_F \cdot \sum_{i=0}^{i=N-2} L_f^i \cdot \bar{w}$$

From *Assumption* 3.3 it yields:

$$\begin{aligned}
&E(\tilde{\boldsymbol{x}}(k+N|k)) - E(\hat{\boldsymbol{x}}(k+N-1|k-1)) \\
&= E(\tilde{\boldsymbol{x}}(k+N|k)) - E(\tilde{\boldsymbol{x}}(k+N-1|k)) + E(\tilde{\boldsymbol{x}}(k+N-1|k)) - E(\hat{\boldsymbol{x}}(k+N-1|k-1)) \\
&\leq E(\tilde{\boldsymbol{x}}(k+N|k)) - E(\tilde{\boldsymbol{x}}(k+N-1|k)) + L_E L_f^{N-1}\bar{w}
\end{aligned}$$

We used the instantaneous difference of the predictive state $\hat{\boldsymbol{x}}(k+N-1|k)$ and the feasible state $\tilde{\boldsymbol{x}}(k+N-1|k-1)$ at the time-step $k+N-1$ such that:

$$||\tilde{\boldsymbol{x}}(k+N-1|k-1) - \hat{\boldsymbol{x}}(k+N-1|k)|| \leq L_f^{N-1}\bar{w}$$

Therefore, we obtain:

$$\begin{aligned}
\Delta J \leq &\left(L_F \sum_{i=0}^{i=N-2} L_f^i + L_E L_f^{N-1}\right)\bar{w} + \Big[F(\tilde{\boldsymbol{x}}(k+N-1|k), h(\hat{\boldsymbol{x}}(k+N-1|k))) \\
&+ E(\tilde{\boldsymbol{x}}(k+N|k)) - E(\tilde{\boldsymbol{x}}(k+N-1|k))\Big] - F(\hat{\boldsymbol{x}}(k-1|k-1), \boldsymbol{\tau}^*(k-1|k-1))
\end{aligned}$$

Finally, taking into account the *Assumption* 3.2 and *Lemma* 3.2:

$$\begin{aligned}
\Delta J &\leq \left(L_F \sum_{i=0}^{i=N-2} L_f^i + L_E L_f^{N-1}\right)\bar{w} - F(\hat{\boldsymbol{x}}(k-1|k-1), \boldsymbol{\tau}^*(k-1|k-1)) \\
&\leq \left(L_F \sum_{i=0}^{i=N-2} L_f^i + L_E L_f^{N-1}\right)\bar{w} - \underline{\mathrm{F}}(||\boldsymbol{x}_{k-1}||)
\end{aligned}$$

11.4 Chapter 4–Appendix IV

The Euclidean norm is used for the sake of simplicity. We get:

$$||f(\boldsymbol{x}_1, \boldsymbol{v}) - f(\boldsymbol{x}_2, \boldsymbol{v})||^2 = || \begin{pmatrix} \chi_1 + \cos\psi_1 udt - \chi_2 - \cos\psi_2 udt \\ y_1 + \sin\psi_1 udt - y_2 - \sin\psi_2 udt \\ z_1 - z_2 \\ \psi_1 - \psi_2 \end{pmatrix} ||^2 =$$

$$\begin{aligned}
&|\chi_1 - \chi_2 + udt(\cos\psi_1 - \cos\psi_2)|^2 + |z_1 - z_2|^2 + \\
&|\psi_1 - \psi_2|^2 + |y_1 - y_2 + udt(\sin\psi_1 - \sin\psi_2)|^2
\end{aligned}$$

From the mean value theorem, we can obtain:

$$||\cos\psi_1 - \cos\psi_2|| = ||\sin\psi^*(\psi_1 - \psi_2)|| \leq ||\psi_1 - \psi_2||$$

where $\psi^* \in (\psi_1, \psi_2)$. This yields the following:

$$
\begin{aligned}
&|\chi_1 - \chi_2 + udt(\cos\psi_1 - \cos\psi_2)|^2 \leq [2\max\{|\chi_1 - \chi_2|, udt|\cos\psi_1 - \cos\psi_2|\}]^2 \\
&\leq 4\max\{(\chi_1 - \chi_2)^2, (udt)^2(\psi_1 - \psi_2)^2\} \leq \max\{4, 4(udt)^2\}\max\{(\chi_1 - \chi_2)^2, (\psi_1 - \psi_2)^2\} \\
&\leq \max\{4, 4(\bar{u}dt)^2\}[(\chi_1 - \chi_2)^2 + (\psi_1 - \psi_2)^2]
\end{aligned}
$$

Applying similar derivations to the other elements, it can be concluded that for all $\boldsymbol{x}_1, \boldsymbol{x}_2 \in X$ it can be obtained:

$$
\begin{aligned}
&||f(\boldsymbol{x}_1, \boldsymbol{v}) - f(\boldsymbol{x}_2, \boldsymbol{v})||^2 \leq \\
&\max\{4, 4(\bar{u}dt)^2\}[(\chi_1 - \chi_2)^2 + (\psi_1 - \psi_2)^2] + (z_1 - z_2)^2 + \\
&\max\{4, 4(\bar{u}dt)^2\}[(y_1 - y_2)^2 + (\psi_1 - \psi_2)^2] + (\psi_1 - \psi_2)^2 = \\
&\max\{4, 4(\bar{u}dt)^2\}[(\chi_1 - \chi_2)^2 + (y_1 - y_2)^2 + 2(\psi_1 - \psi_2)^2] + (z_1 - z_2)^2 + (\psi_1 - \psi_2)^2 \\
&\leq (\max\{8, 8(\bar{u}dt)^2\} + 1)\Big[(\chi_1 - \chi_2)^2 + (y_1 - y_2)^2 + (\psi_1 - \psi_2)^2 + (z_1 - z_2)^2\Big] \\
&\leq (\max\{8, 8(\bar{u}dt)^2\} + 1)||x_1 - x_2||^2
\end{aligned}
$$

thus the Lipschitz constant is $L_f \triangleq (\max\{8, 8(\bar{u}dt)^2\} + 1)^{1/2}$, with $0 < L_f < \infty$ and that concludes the proof.

11.5 Chapter 4–Appendix V

$$
\begin{aligned}
&||F(\boldsymbol{x}_1, \boldsymbol{v}) - F(\boldsymbol{x}_2, \boldsymbol{v})|| = ||\boldsymbol{x}_1^\top \boldsymbol{Q}\boldsymbol{x}_1 - \boldsymbol{x}_2^\top \boldsymbol{Q}\boldsymbol{x}_2|| = ||\boldsymbol{x}_1^\top \boldsymbol{Q}\boldsymbol{x}_1 \\
&- \boldsymbol{x}_1^\top \boldsymbol{Q}\boldsymbol{x}_2 + \boldsymbol{x}_1^\top \boldsymbol{Q}\boldsymbol{x}_2 - \boldsymbol{x}_2^\top \boldsymbol{Q}\boldsymbol{x}_2|| = ||\boldsymbol{x}_1^\top \boldsymbol{Q}(\boldsymbol{x}_1 - \boldsymbol{x}_2) + \\
&+ (\boldsymbol{x}_1^\top - \boldsymbol{x}_2^\top)\boldsymbol{Q}\boldsymbol{x}_2|| = ||\boldsymbol{x}_1^\top \boldsymbol{Q}(\boldsymbol{x}_1 - \boldsymbol{x}_2) + \boldsymbol{x}_2^\top \boldsymbol{Q}(\boldsymbol{x}_1 - \boldsymbol{x}_2)|| = \\
&||(\boldsymbol{x}_1^\top + \boldsymbol{x}_2^\top)\boldsymbol{Q}(\boldsymbol{x}_1 - \boldsymbol{x}_2)|| \leq (||\boldsymbol{x}_1|| + ||\boldsymbol{x}_2||)\sigma_{max}(\boldsymbol{Q})||\boldsymbol{x}_1 - \boldsymbol{x}_2||
\end{aligned}
$$

Notice that $\forall \boldsymbol{x} \in X$ we have:$||\boldsymbol{x}|| = (|\chi|^2 + |y|^2 + |z|^2 + |\psi|^2)^{\frac{1}{2}} \leq (R_{max}^2 + z_{max}^2 + (\frac{\pi}{2})^2)^{\frac{1}{2}}$, which concludes the proof. Notice that the maximum value z_{max} along the z-axis is calculated by substituting the maximum feasible distance R_{max} into the visibility constraints of (4.10).

11.6 Chapter 4–Appendix VI

From Lemma 2 we can derive that:

$$
\begin{aligned}
&||\hat{\boldsymbol{x}}(k+N|k+1) - \hat{\boldsymbol{x}}(k+N|k)|| \leq L_f^{N-1}\bar{w} \\
&||\hat{\boldsymbol{x}}(k+N|k+2) - \hat{\boldsymbol{x}}(k+N|k)|| \leq L_f^{N-2}((1+L_f)\bar{w}) \\
&\qquad\qquad \vdots \\
&||\hat{\boldsymbol{x}}(k+N|k+m) - \hat{\boldsymbol{x}}(k+N|k)|| \leq L_f^{(N-m)} \sum_{i=0}^{m-1} (L_f)^i \bar{w}
\end{aligned}
$$

from the Lipschitz property of $E(\cdot)$ we get:

$$
\begin{aligned}
&E(\hat{\boldsymbol{x}}(k+N|k+m)) - E(\hat{\boldsymbol{x}}(k+N|k)) \\
&\leq L_E||\hat{\boldsymbol{x}}(k+N|k+m) - \hat{\boldsymbol{x}}(k+N|k)|| \\
&\leq L_E L_f^{(N-m)} \cdot \sum_{i=0}^{m-1} (L_f)^i \bar{w}
\end{aligned}
$$

Noticing that $\hat{\boldsymbol{x}}(k+N|k) \in \mathscr{O}_f$ and from Assumption-4.4, we get the following:

$$
E(\hat{\boldsymbol{x}}(k+N|k+m) \leq \alpha_{\varepsilon_f} + L_E G(m)\bar{w}
$$

with $G(m) \triangleq L_f^{(N-m)} \cdot \sum_{i=0}^{m-1}(L_f)^i$. It should hold that $E(\hat{\boldsymbol{x}}(k+N|k+m)) \leq \alpha_\varepsilon$, i.e., $\hat{\boldsymbol{x}}(k+N|k+m) \in \mathscr{O}$, thus:

$$
\alpha_{\varepsilon_f} + L_E G(m)\bar{w} \leq \alpha_\varepsilon \Rightarrow \bar{w} \leq \frac{(\alpha_\varepsilon - \alpha_{\varepsilon_f})}{L_E L_f^{(N-m)} \cdot \sum_{i=0}^{m-1}(L_f)^i} \tag{11.1}
$$

Now, applying a local control law, we get $\hat{\boldsymbol{x}}(k+N+1|k+m) \in \mathscr{O}_f$ for all $m = 1, \ldots, N-1$. From these results it can be concluded that the set $\mathcal{X}^{MPC}$ is robustly positively invariant if the uncertainties are bounded by (11.1) for all $m = 1, \ldots, N-1$. Notice, that (11.1) should still hold for $m = 1$, for the problem to be meaningful, in the sense that it should be feasible at least in the time-triggered case.

11.7 Chapter 4–Appendix VII

$$
\begin{aligned}
&\Delta J_m = \tilde{J}_N(k+m) - J_N^*(k) = \\
&\sum_{i=0}^{N-1} F(\tilde{\boldsymbol{x}}(k+i+m|k+m), \tilde{\boldsymbol{v}}(k+i+m|k+m)) \\
&- \sum_{i=0}^{N-1} F(\hat{\boldsymbol{x}}(k+i|k), \boldsymbol{v}^*(k+i|k)) + E(\tilde{\boldsymbol{x}}(k+N+m|k+m)) - E(\hat{\boldsymbol{x}}(k+N|k)) \\
&= \sum_{i=0}^{N-(m+1)} \Big\{ F(\tilde{\boldsymbol{x}}(k+i+m|k+m), \tilde{\boldsymbol{v}}(k+i+m|k+m)) - F(\hat{\boldsymbol{x}}(k+i+m|k), \boldsymbol{v}^*(k+i+m|k)) \Big\} \\
&- \sum_{i=0}^{m-1} F(\hat{\boldsymbol{x}}(k+i|k), \boldsymbol{v}^*(k+i|k)) + \sum_{i=1}^{m} F(\tilde{\boldsymbol{x}}(k+N-1+i|k+m), h(\tilde{\boldsymbol{x}}(k+N-1+i|k+m))) \\
&+ E(\tilde{\boldsymbol{x}}(k+N+m|k+m)) - E(\hat{\boldsymbol{x}}(k+N|k))
\end{aligned}
$$

with $\tilde{\boldsymbol{x}}(k+i|k+m)$ denoting the "feasible" state of the system which accounts for the predicted state at time-step $k+i$, based on the measurement of the real state at time-step $k+m$, when the feasible control sequence from (4.22) is used. Also from Lemma 4.2 and with the help of Lemma 4.4, it yields:

$$
\begin{aligned}
&\sum_{i=0}^{N-(m+1)} \Big\{ F(\tilde{\boldsymbol{x}}(k+i+m|k+m), \tilde{\boldsymbol{v}}(k+i+m|k+m)) - \\
&\qquad - F(\hat{\boldsymbol{x}}(k+i+m|k), \boldsymbol{v}^*(k+i+m|k)) \Big\} \\
&\leq L_F \sum_{i=0}^{N-(m+1)} (L_f)^i \bar{w}
\end{aligned}
$$

Using $\sum_{i=0}^{m-1} \Big\{ E(\tilde{\boldsymbol{x}}(k+N+i|k+m)) - E(\tilde{\boldsymbol{x}}(k+N+i|k+m)) \Big\}$ which adds up to zero, while taking into account Assumption 4.2, it can be obtained:

$$
\begin{aligned}
&F(\tilde{\boldsymbol{x}}(k+N-1+m|k+m), h(\tilde{\boldsymbol{x}}(k+N-1+m|k+m))) + \\
&E(\tilde{\boldsymbol{x}}(k+N+m|k+m)) - E(\tilde{\boldsymbol{x}}(k+N-1+m|k+m)) \leq 0
\end{aligned}
$$

Moreover:

$$
E(\tilde{\boldsymbol{x}}(k+N|k+m)) - E(\hat{\boldsymbol{x}}(k+N|k)) \leq L_E (L_f)^{(N-m)} \bar{w}
$$

Also, using Lemma (4.3) we get:

$$\sum_{i=0}^{m-1} F(\hat{\boldsymbol{x}}(k+i|k), \boldsymbol{v}^*(k+i|k)) \geq$$
$$\geq \sum_{i=0}^{m-1} min(q_1, q_2, q_3, q_4, r_1, r_2, r_3)||\hat{\boldsymbol{x}}(k+i|k)||$$

Substituting all these inequalities to the difference ΔJ_m, yields:

$$\Delta J_m = \tilde{J}_N(k+m) - J_N^*(k) \leq$$
$$\leq \left(L_E (L_f)^{(N-m)} + L_F \sum_{i=0}^{N-(m+1)} (L_f)^i \right) \bar{w}$$
$$- \sum_{i=0}^{m-1} min(q_1, q_2, q_3, q_4, r_1, r_2, r_3)||\hat{\boldsymbol{x}}(k+i|k)||$$

11.8 Chapter 5–Appendix VIII

For the proof, first we derive a bound for the difference of $\tilde{J}_N(k+m) - J_N^*(k-1)$, then an upper bound will be found for $(\frac{1}{B} - 1)\tilde{J}_N(k+m)$, and finally the overall difference ΔJ_m will be obtained.

$$\tilde{J}_N(k+m) - J_N^*(k-1) =$$
$$= \sum_{i=0}^{N-1} F(\tilde{s}(k+i+m|k+m), \tilde{V}(k+i+m|k+m))$$
$$- \sum_{i=0}^{N-1} F(\hat{\boldsymbol{s}}(k+i-1|k-1), V^*(k+i-1|k-1))$$
$$+ E(\tilde{s}(k+N+m|k+m)) - E(\hat{\boldsymbol{s}}(k+N-1|k-1)) =$$
$$= \sum_{i=0}^{N-(m+2)} \{F(\tilde{s}(k+i+m|k+m), \tilde{V}(k+i+m|k+m))$$
$$- F(\hat{\boldsymbol{s}}(k+i+m|k-1), V^*(k+i+m|k-1))\}$$
$$- \sum_{i=-1}^{m-1} F(\hat{\boldsymbol{s}}(k+i|k-1), V^*(k+i|k-1)) +$$
$$\sum_{i=0}^{m} F(\tilde{s}(k+N-1+i|k+m), h(\tilde{s}(k+N-1+i|k+m)))$$
$$+ \sum_{i=-1}^{m-1} \{E(\tilde{s}(k+N+i|k+m)) - E(\tilde{s}(k+N+i|k+m))\}$$
$$+ E(\tilde{s}(k+N+m|k+m)) - E(\hat{\boldsymbol{s}}(k+N-1|k-1))$$

with $\tilde{s}(k+i|k+m)$ to denote the "feasible" state of the system which accounts for the predicted state at the time-step $k+i$, based on the measurement of the actual state at time-step $k+m$,

when the feasible control from (5.18) is used. Also from Property-5.2 and Property-5.4 , it yields:

$$\begin{aligned}&\sum_{i=0}^{N-(m+2)} \{F(\tilde{s}(k+i+m|k+m), \tilde{V}(k+i+m|k+m)) \\ &- F(\hat{\boldsymbol{s}}(k+i+m|k-1), V^*(k+i+m|k-1))\} \\ &\leq L_S(m) \end{aligned} \tag{11.2}$$

Where $L_S(m) = C_F \cdot \sum_{i=0}^{N-(m+2)} (C_f)^i \bar{\boldsymbol{\xi}}$. Taking into account the Assumption-5.2, it is obtained:

$$\begin{aligned}&F(\tilde{s}(k+N-1+m|k+m), h(\tilde{s}(k+N-1+m|k+m)))+ \\ &E(\tilde{s}(k+N+m|k+m)) - E(\tilde{s}(k+N-1+m|k+m)) \leq 0 \end{aligned} \tag{11.3}$$

Moreover:

$$\begin{aligned}&E(\tilde{s}(k+N-1|k+m)) - E(\hat{\boldsymbol{s}}(k+N-1|k-1)) \\ &\leq C_E \cdot (C_f)^{(N-(m+1))} \cdot \bar{\boldsymbol{\xi}}\end{aligned}$$

Also, employing Property-5.3, it can be obtained:

$$\begin{aligned}&\sum_{i=-1}^{m-1} F(\hat{\boldsymbol{s}}(k+i|k+i), V^*(k+i|k+i)) \\ &\geq \sum_{i=-1}^{m-1} min(q_1, \ldots, q_{2n}, r_1, \ldots, r_6)||\hat{\boldsymbol{s}}(k+i|k-1)||\end{aligned}$$

After substituting it yields:

$$\begin{aligned}&\tilde{J}_N(k+m) - {J^*}_N(k-1) \\ &\leq \left(C_E \cdot (C_f)^{(N-(m+1))} + C_F \cdot \sum_{i=0}^{N-(m+2)} (L_f)^i \right) \bar{\boldsymbol{\xi}} \\ &- \sum_{i=-1}^{m-1} min(q_1, \ldots, q_{2n}, r_1, \ldots, r_6)||\hat{\boldsymbol{s}}(k+i|k-1)|| \end{aligned} \tag{11.4}$$

Now, we can obtain a bound of $(\frac{1}{B}-1)\tilde{J}_N(k+m)$:

$$
\begin{aligned}
&(\frac{1}{B}-1)\tilde{J}_N(k+m) = \\
&(\frac{1}{B}-1)\Big(\sum_{i=0}^{N-1} F(\tilde{s}(k+i+m|k+m),\tilde{V}(k+i+m|k+m)) \\
&+ E(\tilde{s}(k+N+m|k+m))\Big) \\
&= (\frac{1}{B}-1)\Bigg(\sum_{i=0}^{N-(m+2)} F(\tilde{s}(k+i+m|k+m),\tilde{V}(k+i+m|k+m))+ \\
&+ \sum_{i=0}^{m} F(\tilde{s}(k+N-1+i|k+m),h(\tilde{s}(k+N-1+i|k+m)))+ \\
&+ \sum_{i=-1}^{m-1}\{E(\tilde{s}(k+N+i|k+m)) - E(\tilde{s}(k+N+i|k+m))\} \\
&+ E(\tilde{s}(k+N+m|k+m))\Bigg)
\end{aligned}
$$

Using the (11.2) and the expression of $F(\cdot)$, results into the following:

$$
\begin{aligned}
&\sum_{i=0}^{N-(m+2)} F(\tilde{s}(k+i+m|k+m),\tilde{V}(k+i+m|k+m)) \leq \\
&\leq L_S(m) + \sum_{i=0}^{N-(m+2)} F(\hat{\boldsymbol{s}}(k+i+m|k-1),V^*(k+i+m|k-1)) \leq \\
&\leq L_S(m) + L_P(m)
\end{aligned}
$$

Where $L_P(m) = \max(p_1,\ldots,p_{2n},r_1,\ldots,r_6)\sum_{i=0}^{N-(m+2)}|||[\hat{\boldsymbol{s}}(\cdot),V^*(\cdot)]|||$. Now using (11.3), we get:

$$
(\frac{1}{B}-1)\tilde{J}_N(k+m) \leq (\frac{1}{B}-1)(L_S + L_P + E(\tilde{s}(k+N-1|k+m)))
$$

Taking into account the Assumption-5.4, Finally we get:

$$
(\frac{1}{B}-1)\tilde{J}_N(k+m) \leq (\frac{1}{B}-1)(L_S + L_P + \max(p_1,\ldots,p_{2n})\varepsilon_0^2) \tag{11.5}
$$

where ε_0 is a positive parameter, see Assumption-5.1.

11.9 Dynamical Systems

Consider the initial value problem:

$$
\dot{\xi} = H(t,\xi), \xi(0) = \xi^0 \in \Omega_\xi, \tag{11.6}
$$

with $H : \mathbb{R}_{\geq 0} \times \Omega_\xi \to \mathbb{R}^n$, where $\Omega_\xi \subseteq \mathbb{R}^n$ is a non-empty open set.

Definition 11.1. [264] A solution $\xi(t)$ of the initial value problem (11.6) is maximal if it has no proper right extension that is also a solution of (11.6).

Theorem 11.2. *[264] Consider the initial value problem (11.6). Assume that $H(t,\xi)$ is: a) locally Lipschitz in ξ for almost all $t \in \mathbb{R}_{\geq 0}$, b) piecewise continuous in t for each fixed $\xi \in \Omega_\xi$ and c) locally integrable in t for each fixed $\xi \in \Omega_\xi$. Then, there exists a maximal solution $\xi(t)$ of (11.6) on the time interval $[0, \tau_{\max})$, with $\tau_{\max} \in \mathbb{R}_{>0}$ such that $\xi(t) \in \Omega_\xi, \forall t \in [0, \tau_{\max})$.*

Proposition 11.1. *[264] Assume that the hypotheses of Theorem 11.2 hold. For a maximal solution $\xi(t)$ on the time interval $[0, \tau_{\max})$ with $\tau_{\max} < \infty$ and for any compact set $\Omega'_\xi \subseteq \Omega_\xi$, there exists a time instant $t' \in [0, \tau_{\max})$ such that $\xi(t') \notin \Omega'_\xi$.*

Chapter 12

List of Publications

My research has resulted to 14 peer-reviewed papers for international conferences worldwide and 6 peer-reviewed journal papers [57, 65–67, 108, 116–119, 152, 153, 176–179, 265–269]. The list of publications at this time (12/2018), is as follows:

Refereed Journal Papers

[J1] Charalampos P Bechlioulis, George C Karras, **Shahab Heshmati-alamdari**, Kostas J Kyriakopoulos, "Trajectory Tracking With Prescribed Performance for Underactuated Underwater Vehicles Under Model Uncertainties and External Disturbances", IEEE Transactions on Control Systems Technology 25 (2), 429-440, 2016.

[J2] **Shahab Heshmati-alamdari**, Charalampos P. Bechlioulis, George C. Karras, Kostas J. Kyriakopoulos, "Cooperative Impedance Control for Multiple Underwater Vehicle Manipulator Systems under Lean Communication", IEEE Journal of Oceanic Engineering, submitted

[J3] **Shahab Heshmati-alamdari**, George C. Karras, Panos Marantos, and Kostas J. Kyriakopoulos, "A Robust Predictive Control Approach for Underwater Robotic Vehicles", IEEE Transactions on Control Systems Technology (Under revision)

[J4] **Shahab Heshmati-alamdari**, Charalampos P. Bechlioulis , George C. Karras, Alexandros Nikou, Dimos V Dimarogonas, Kostas J. Kyriakopoulos"A Robust Interaction Control Approach for Underwater Vehicle Manipulator Systems", IFAC Annual Reviews in Control" Journal, 2018, Published.

[J5] Charalampos P Bechlioulis, **Shahab Heshmati-alamdari**, George C. Karras, Kostas J. Kyriakopoulos"Robust Image Based Visual Servoing with Prescribed Performance under Field of View Constraints", IEEE Transactions on Robotics, (Under revision).

[J6] **Shahab Heshmati-alamdari**, George C. Karras, Kostas J. Kyriakopoulos"A Decentralized Predictive Control Approach for Cooperative Transportation by Multiple Underwater Vehicle Manipulator Systems", (Under preparation).

Refereed Conference Papers

[C1] Alina Eqtami,**Shahab Heshmati-alamdari**, Dimos V Dimarogonas, Kostas J Kyriakopoulos, "Self-triggered model predictive control for nonholonomic systems", European Control Conference (ECC), 2013.

[C2] Alina Eqtami, **Shahab Heshmati-alamdari**, Dimos V Dimarogonas, Kostas J Kyriakopoulos, "A self-triggered model predictive control framework for the cooperation of distributed nonholonomic agents", IEEE 52nd Annual Conference on Decision and Control (CDC), 2013.

[C3] **Shahab Heshmati-alamdari**, Alina Eqtami, George C Karras, Dimos V Dimarogonas, Kostas J Kyriakopoulos, "A self-triggered visual servoing model predictive control scheme for under-actuated underwater robotic vehicles", IEEE International Conference on Robotics and Automation (ICRA), 2014.

[C4] **Shahab Heshmati-alamdari**, George K Karavas, Alina Eqtami, Michael Drossakis, Kostas J Kyriakopoulos, "Robustness analysis of model predictive control for constrained image-based visual servoing", IEEE International Conference on Robotics and Automation (ICRA), 2014.

[C5] Natalia Hurtos, Narcis Palomeras, Arnau Carrera, Marc Carreras, Charalampos P. Bechlioulis, George C. Karras, **Shahab Heshmati-alamdari**, Kostas Kyriakopoulos, "Sonar-Based Chain Following Using an Autonomous Underwater Vehicle", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), 2014.

[C6] **Shahab Heshmati-alamdari**, Charalampos P Bechlioulis, Minas V Liarokapis, Kostas J Kyriakopoulos, "Prescribed performance image based visual servoing under field of view constraints", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014), 2014.

[C7] **Shahab Heshmati-alamdari**, George C. Karras, Alina Eqtami and Kostas J. Kyriakopoulos,"A Robust Self Triggered Image Based Visual Servoing Model Predictive Control Scheme for Small Autonomous Robots", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015.

[C8] Alexandros Nikou, Christos Verginis, **Shahab Heshmati-alamdari**, Dimos V Dimarogonas, "A Nonlinear Model Predictive Control Scheme for Cooperative Manipulation with Singularity and Collision Avoidance", IEEE 25th Mediterranean Conference on Control and Automation, 2017.

[C9] Alexandros Nikou, **Shahab Heshmati-alamdari**, Christos Verginis, Dimos V Dimarogonas, "Decentralized Abstractions and Timed Constrained Planning of a General Class of Coupled Multi-Agent Systems", IEEE 56nd Annual Conference on Decision and Control (CDC), 2017.

[C10] **Shahab Heshmati-alamdari**, Alexandros Nikou, Kostas J. Kyriakopoulos, Dimos V. Dimarogonas, "A Robust Force Control Approach for Underwater Vehicle Manipulator Systems", The 20th World Congress of the International Federation of Automatic Control (IFAC 2017), 2017.

[C11] **Shahab Heshmati-alamdari**, George C. Karras, Panos Marantos, and Kostas J. Kyriakopoulos, "A Robust Model Predictive Control Approach for Underwater Robotic Vehicles Operating in a Constrained workspace", IEEE International Conference on Robotics and Automation (ICRA), 2018.

[C12] Michael Logothetis, George Karras, **Shahab Heshmati-alamdari**, Panagiotis Vlantis, Kostas Kyriakopoulos, "A Model Predictive Control Approach for Vision-based Object Grasping via Mobile Manipulator", IEEE IROS 2018 - International Conference on Intelligent Robots, Accepted.

[C13] **Shahab Heshmati-alamdari**, Charalampos P. Bechlioulis, George C. Karras, Kostas J. Kyriakopoulos, "Decentralized Impedance Control for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems under Lean Communication", IEEE OES Autonomous Underwater Vehicle Symposium, 2018, Accepted.

[C14] **Shahab Heshmati-alamdari**, George C. Karras, Kostas J. Kyriakopoulos, 'A Decentralized Predictive Control Approach for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems", IEEE International Conference on Robotics and Automation (ICRA), 2019, Submitted.

Bibliography

[1] V. Asouti, X. Trompoukis, I. Kampolis, and K. Giannakoglou, "Unsteady cfd computations using vertex-centered finite volumes for unstructured grids on graphics processing units," *International Journal for Numerical Methods in Fluids*, vol. 67, no. 2, pp. 232–246, 2011.

[2] Y. Sun and C. Cheah, "Coordinated control of multiple cooperative underwater vehicle-manipulator systems holding a common load," *Ocean '04 - MTS/IEEE Techno-Ocean '04: Bridges across the Oceans - Conference Proceedings*, vol. 3, pp. 1542–1547, 2004.

[3] J. Yuh, "Design and control of autonomous underwater robots: A survey," *Autonomous Robots*, vol. 8, no. 1, pp. 7–24, 2000.

[4] G. Griffiths, *Technology and Applications of Autonomous Underwater Vehicles*, ser. Ocean science and technology. CRC Press, 2002. [Online]. Available: https://books.google.gr/books?id=8DNKu9fDHuwC

[5] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, 2011.

[6] Z. Zeng, L. Lian, K. Sammut, F. He, Y. Tang, and A. Lammas, "A survey on path planning for persistent autonomy of autonomous underwater vehicles," *Ocean Engineering*, vol. 110, pp. 303–313, 2015.

[7] B. Hobson, J. Bellingham, B. Kieft, R. McEwen, M. Godin, and Y. Zhang, "Tethys-class long range auvs-extending the endurance of propeller-driven cruising auvs from days to weeks," *2012 IEEE/OES Autonomous Underwater Vehicles, AUV 2012*, 2012.

[8] P. Ridao, M. Carreras, D. Ribas, P. Sanz, and G. Oliver, "Intervention auvs: The next challenge," *Annual Reviews in Control*, vol. 40, pp. 227–241, 2015.

[9] F. Bruno, M. Muzzupappa, L. Barbieri, A. Gallo, G. Ritacco, A. Lagudi, M. F. La Russa, S. A. Ruffolo, G. M. Crisci, M. Ricca *et al.*, "The comas project: New materials and tools for improving the in situ documentation, restoration, and conservation of underwater archaeological remains," *Marine Technology Society Journal*, vol. 50, no. 4, pp. 108–118, 2016.

[10] G. Antonelli, *"Underwater Robots"*, ser. Springer Tracts in Advanced Robotics. Springer International Publishing, 2013.

[11] H. Farivarnejad and S. Moosavian, "Multiple impedance control for object manipulation by a dual arm underwater vehicle-manipulator system," *Ocean Engineering*, vol. 89, pp. 82–98, 2014.

[12] H. Shim, B.-H. Jun, P.-M. Lee, H. Baek, and J. Lee, "Workspace control system of underwater tele-operated manipulators on an rov," *Ocean Engineering*, vol. 37, no. 11-12, pp. 1036–1047, 2010.

[13] P. Londhe, S. Mohan, B. Patre, and L. Waghmare, "Robust task-space control of an autonomous underwater vehicle-manipulator system by pid-like fuzzy control scheme with disturbance estimator," *Ocean Engineering*, vol. 139, pp. 1–13, 2017.

[14] S. Mohan and J. Kim, "Coordinated motion control in task space of an autonomous underwater vehicle-manipulator system," *Ocean Engineering*, vol. 104, pp. 155–167, 2015.

[15] G. Marani, S. Choi, and J. Yuh, "Underwater autonomous manipulation for intervention missions auvs," *Ocean Engineering*, vol. 36, no. 1, pp. 15–23, 2009.

[16] E. Simetti and G. Casalino, "Whole body control of a dual arm underwater vehicle manipulator system," *Annual Reviews in Control*, vol. 40, pp. 191–200, 2015.

[17] D. Lane, D. O'Brien, M. Pickett, J. Davies, G. Robinson, D. Jones, E. Scott, G. Casalino, G. Bartolini, G. Cannata, A. Ferrara, D. Angelleti, M. Coccoli, G. Veruggio, R. Bono, P. Virgili, M. Canals, R. Pallas, E. Gracia, and C. Smith, "Amadeus: Advanced manipulation for deep underwater sampling," *IEEE Robotics and Automation Magazine*, vol. 4, no. 4, pp. 34–45, 1997.

[18] V. Rigaud, . Coste-Manière, M. Aldon, P. Probert, M. Perrier, P. Rives, D. Simon, D. Lane, J. Kiener, A. Casals, J. Amat, P. Dauchez, and M. Chantler, "Union: Underwater intelligent operation and navigation," *IEEE Robotics and Automation Magazine*, vol. 5, no. 1, pp. 25–34, 1998.

[19] P. Sanz, P. Ridao, G. Oliver, G. Casalino, C. Insaurralde, C. Silvestre, C. Melchiorri, and A. Turetta, "Trident: Recent improvements about autonomous underwater intervention missions," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 3, no. PART 1, pp. 355–360, 2012.

[20] M. Prats, J. Garcia, S. Wirth, D. Ribas, P. Sanz, P. Ridao, N. Gracias, and G. Oliver, "Multipurpose autonomous underwater intervention: A systems integration perspective," *2012 20th Mediterranean Conference on Control and Automation, MED 2012 - Conference Proceedings*, pp. 1379–1384, 2012.

[21] J. Fernández, M. Prats, P. Sanz, J. García, R. Marín, M. Robinson, D. Ribas, and P. Ridao, "Grasping for the seabed: Developing a new underwater robot arm for shallow-water intervention," *IEEE Robotics and Automation Magazine*, vol. 20, no. 4, pp. 121–130, 2013.

[22] E. Simetti, G. Casalino, S. Torelli, A. Sperindé, and A. Turetta, "Floating underwater manipulation: Developed control methodology and experimental validation within the trident project," *Journal of Field Robotics*, vol. 31, no. 3, pp. 364–385, 2014.

[23] M. Prats, D. Ribas, N. Palomeras, J. García, V. Nannen, S. Wirth, J. Fernández, J. Beltrán, R. Campos, P. Ridao, P. Sanz, G. Oliver, M. Carreras, N. Gracias, R. Marín, and A. Ortiz, "Reconfigurable auv for intervention missions: A case study on underwater object recovery," *Intelligent Service Robotics*, vol. 5, pp. 19–31, 2012.

[24] D. Ribas, P. Ridao, A. Turetta, C. Melchiorri, G. Palli, J. Fernandez, and P. Sanz, "I-auv mechatronics integration for the trident fp7 project," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2583–2592, 2015.

[25] D. Lane, F. Maurelli, P. Kormushev, M. Carreras, M. Fox, and K. Kyriakopoulos, "Persistent autonomy: The challenges of the pandora project," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 9, no. PART 1, pp. 268–273, 2012.

[26] A. Carrera, N. Palomeras, D. Ribas, P. Kormushev, and M. Carreras, "An intervention-auv learns how to perform an underwater valve turning," *OCEANS 2014*, 2014.

[27] A. Carrera, N. Palomeras, N. Hurtos, P. Kormushev, and M. Carreras, "Learning multiple strategies to perform a valve turning with underwater currents using an i-auv," *MTS/IEEE OCEANS 2015 - Genova: Discovering Sustainable Ocean Energy for a New World*, 2015.

[28] J. Gancet, D. Urbina, P. Letier, M. Ilzokvitz, P. Weiss, F. Gauch, G. Antonelli, G. Indiveri, G. Casalino, A. Birk, M. Pfingsthorn, S. Calinon, A. Tanwani, A. Turetta, C. Walen, and L. Guilpain, "Dexrov: Dexterous undersea inspection and maintenance in presence of communication latencies," *IFAC-PapersOnLine*, vol. 28, no. 2, pp. 218–223, 2015.

[29] S. Abu Sharkh and G. Griffiths, "Energy storage systems for unmanned underwater vehicles," *Underwater Technology*, vol. 25, no. 3, pp. 143–148, 2003.

[30] . Hasvold, N. Størkersen, S. Forseth, and T. Lian, "Power sources for autonomous underwater vehicles," *Journal of Power Sources*, vol. 162, no. 2 SPEC. ISS., pp. 935–942, 2006.

[31] J. Bellingham and K. Rajan, "Robotics in remote and hostile environments," *Science*, vol. 318, no. 5853, pp. 1098–1102, 2007.

[32] T. Joung, K. Sammut, F. He, and S.-K. Lee, "A study on the design optimization of an auv by using computational fluid dynamic analysis," *Proceedings of the International Offshore and Polar Engineering Conference*, pp. 696–702, 2009.

[33] D. N. Subramani and P. F. Lermusiaux, "Energy-optimal path planning by stochastic dynamically orthogonal level-set optimization," *Ocean Modelling*, vol. 100, pp. 57–77, 2016.

[34] A. Alvarez, A. Caiti, and R. Onken, "Evolutionary path planning for autonomous underwater vehicles in a variable ocean," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 2, pp. 418–429, 2004.

[35] B. Garau, A. Alvarez, and G. Oliver, "Path planning of autonomous underwater vehicles in current fields with complex spatial variability: an a* approach," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on.* IEEE, 2005, pp. 194–198.

[36] B. Garau, M. Bonet, A. Alvarez, S. Ruiz, and A. Pascual, "Path planning for autonomus underwater vehicles in realistic oceanic current fields: Application to gliders in the western mediterranean sea," *Journal of Maritime Research*, vol. 6, no. 2, pp. 5–21, 2009.

[37] T. Lee, H. Kim, H. Chung, Y. Bang, and H. Myung, "Energy efficient path planning for a marine surface vehicle considering heading angle," *Ocean Engineering*, vol. 107, pp. 118–131, 2015.

[38] C. Petres, Y. Pailhas, P. Patron, Y. Petillot, J. Evans, and D. Lane, "Path planning for autonomous underwater vehicles," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 331–341, 2007.

[39] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.

[40] J. J. Kuffner and S. M. LaValle, "Rrt-connect: An efficient approach to single-query path planning," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2. IEEE, 2000, pp. 995–1001.

[41] C. S. Tan, R. Sutton, and J. Chudley, "An incremental stochastic motion planning technique for autonomous underwater vehicles," *IFAC Proceedings Volumes*, vol. 37, no. 10, pp. 483–488, 2004.

[42] D. Rao and S. B. Williams, "Large-scale path planning for underwater gliders in ocean currents," in *Australasian Conference on Robotics and Automation (ACRA)*, 2009.

[43] P. Herman, "Decoupled pd set-point controller for underwater vehicles," *Ocean Engineering*, vol. 36, no. 6-7, pp. 529–534, 2009.

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

[45] B. Subudhi, K. Mukherjee, and S. Ghosh, "A static output feedback control design for path following of autonomous underwater vehicle in vertical plane," *Ocean Engineering*, vol. 63, pp. 72–76, 2013.

[46] R. Cristi, F. Papoulias, and A. Healey, "Adaptive sliding mode control of autonomous underwater vehicles in the dive plane," *IEEE Journal of Oceanic Engineering*, vol. 15, no. 3, pp. 152–160, 1990.

[47] A. Healey and D. Lienard, "Multivariable sliding-mode control for autonomous diving and steering of unmanned underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 18, no. 3, pp. 327–339, 1993.

[48] R. Cui, X. Zhang, and D. Cui, "Adaptive sliding-mode attitude control for autonomous underwater vehicles with input nonlinearities," *Ocean Engineering*, vol. 123, pp. 45–54, 2016.

[49] D. Maalouf, A. Chemori, and V. Creuze, "L1 adaptive depth and pitch control of an underwater vehicle with real-time experiments," *Ocean Engineering*, vol. 98, pp. 66–77, 2015.

[50] 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.

[51] G. Antonelli, F. Caccavale, S. Chiaverini, and G. Fusco, "A novel adaptive control law for underwater vehicles," *IEEE Transactions on Control Systems Technology*, vol. 11, no. 2, pp. 221–232, 2003.

[52] S. Mohan and J. Kim, "Indirect adaptive control of an autonomous underwater vehicle-manipulator system for underwater manipulation tasks," *Ocean Engineering*, vol. 54, pp. 233–243, 2012.

[53] M. Carreras, J. Yuh, J. Batlle, and P. Ridao, "A behavior-based scheme using reinforcement learning for autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 416–427, 2005.

[54] J. Yuh, "A neural net controller for underwater robotic vehicles," *IEEE Journal of Oceanic Engineering*, vol. 15, no. 3, pp. 161–166, 1990.

[55] K. Ishaque, S. S. Abdullah, S. Ayob, and Z. Salam, "A simplified approach to design fuzzy logic controller for an underwater vehicle," *Ocean Engineering*, vol. 38, no. 1, pp. 271–284, 2011.

[56] J. Han, J. Ok, and W. K. Chung, "An ethology-based hybrid control architecture for an autonomous underwater vehicle for performing multiple tasks," *IEEE Journal of Oceanic Engineering*, vol. 38, no. 3, pp. 514–521, 2013.

[57] C. Bechlioulis, G. Karras, S. Heshmati-Alamdari, and K. Kyriakopoulos, "Trajectory tracking with prescribed performance for underactuated underwater vehicles under model uncertainties and external disturbances," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 2, pp. 429–440, 2017.

[58] F. Allgöwer, R. Findeisen, and Z. Nagy, "Nonlinear model predictive control: From theory to application," *the Chinese Institute of Chemical Engineers*, vol. 35, no. 3, pp. 299–315, 2004.

[59] C. V. Caldwell, D. D. Dunlap, and E. G. Collins, "Motion planning for an autonomous underwater vehicle via sampling based model predictive control," *OCEANS 2010*, pp. 1–6, 2010.

[60] L. V. Steenson, S. R. Turnock, A. B. Phillips, C. Harris, M. E. Furlong, E. Rogers, L. Wang, K. Bodles, and D. W. Evans, "Model predictive control of a hybrid autonomous underwater vehicle with experimental verification," *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, vol. 228, no. 2, pp. 166–179, 2014.

[61] V. Huynh, M. Dunbabin, and R. Smith, "Predictive motion planning for auvs subject to strong time-varying currents and forecasting uncertainties," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 1144–1151, 2015.

[62] L. Medagoda and S. Williams, "Model predictive control of an autonomous underwater vehicle in an in situ estimated water current profile," *Program Book - OCEANS 2012 MTS/IEEE Yeosu: The Living Ocean and Coast - Diversity of Resources and Sustainable Activities*, 2012.

[63] D. Fernandez and G. Hollinger, "Model predictive control for underwater robots in ocean waves," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 88–95, 2017.

[64] P. Jagtap, P. Raut, P. Kumar, A. Gupta, N. Singh, and F. Kazi, "Control of autonomous underwater vehicle using reduced order model predictive control in three dimensional space," *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 772–777, 2016.

[65] S. Heshmati-Alamdari, A. Eqtami, G. Karras, D. Dimarogonas, and K. Kyriakopoulos, "A self-triggered visual servoing model predictive control scheme for under-actuated underwater robotic vehicles," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3826–3831, 2014.

[66] S. Heshmati-alamdari, G. C. Karras, P. Marantos, and K. J. Kyriakopoulos, "A robust model predictive control approach for autonomous underwater vehicles operating in a constrained workspace," *In Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1–5, 2018.

[67] ——, "A robust predictive control approach for underwater robotic vehicles," *IEEE Transactions on Control Systems Technology, (Under revision)*, 2018.

[68] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, 1996.

[69] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *IEEE Robotics and Automation Magazine*, vol. 13, no. 4, pp. 82–90, 2006.

[70] ——, "Visual servo control, part ii: Advanced approaches," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109–118, 2007.

[71] E. Malis, F. Chaumette, and S. Boudet, "2-1/2-d visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, 1999.

[72] P. Rives and J.-J. Borrelly, "Visual servoing techniques applied to underwater vehicles," *IEEE International Conference on Intelligent Robots and Systems*, vol. 3, 1997.

[73] S. Krupiski, G. Allibert, M.-D. Hua, and T. Hamel, "Pipeline tracking for fully-actuated autonomous underwater vehicle using visual servo control," *Proceedings of the American Control Conference*, pp. 6196–6202, 2012.

[74] G. Karras, S. Loizou, and K. Kyriakopoulos, "Towards semi-autonomous operation of under-actuated underwater vehicles: Sensor fusion, on-line identification and visual servo control," *Autonomous Robots*, vol. 31, no. 1, pp. 67–86, 2011.

[75] C. Silpa-Anan, T. Brinsmead, S. Abdallah, and A. Zelinsky, "Preliminary experiments in visual servo control for autonomous underwater vehicle," *IEEE International Conference on Intelligent Robots and Systems*, vol. 4, pp. 1824–1829, 2001.

[76] S. Negahdaripour and P. Firoozfam, "An rov stereovision system for ship-hull inspection," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 3, pp. 551–564, 2006.

[77] P.-M. Lee, B.-H. Jeon, and S.-M. Kim, "Visual servoing for underwater docking of an autonomous underwater vehicle with one camera," *Oceans Conference Record (IEEE)*, vol. 2, pp. 677–682, 2003.

[78] J.-Y. Park, B.-h. Jun, P.-m. Lee, and J. Oh, "Experiments on vision guided docking of an autonomous underwater vehicle using one camera," *Ocean Engineering*, vol. 36, no. 1, pp. 48–61, 2009.

[79] J.-F. Lots, D. Lane, and E. Trucco, "Application of 2 1/2 d visual servoing to underwater vehicle station-keeping," *Oceans Conference Record (IEEE)*, vol. 2, pp. 1257–1262, 2000.

[80] X. Cufí, R. Garcia, and P. Ridao, "An approach to vision-based station keeping for an unmanned underwater vehicle," *IEEE International Conference on Intelligent Robots and Systems*, vol. 1, pp. 799–804, 2002.

[81] S. Van der Zwaan, A. Bernardino, and J. Santos-Victor, "Visual station keeping for floating robots in unstructured environments," *Robotics and Autonomous Systems*, vol. 39, no. 3-4, pp. 145–155, 2002.

[82] F. Chaumette, *Potential problems of stability and convergence in image-based and position based visual servoing.* LNCIS Series, No 237, Springer-Verlag, 1998.

[83] P. Corke and S. Hutchinson, "A new partitioned approach to image-based visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 4, pp. 507–515, 2001.

[84] R. Mahony, T. Hamel, and F. Chaumette, "A decoupled image space approach to visual servo control of a robotic manipulator," in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, 2002, pp. 3781–3786.

[85] G. Morel, T. Liebezeit, J. Szewczyk, S. Boudet, and J. Pot, *Explicit incorporation of 2D constraints in vision based control of robot manipulators.* London: Springer London, 2000, pp. 99–108. [Online]. Available: http://dx.doi.org/10.1007/BFb0119389

[86] N. Gans, S. Hutchinson, and P. Corke, "Performance tests for visual servo control systems with application to partitioned approaches to visual servo control," *International Journal of Robotics Research*, vol. 22, no. 10-11, pp. 955–981, 2003.

[87] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping features in the field of view in eye-in-hand visual servoing: A switching approach," *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 908–913, 2004.

[88] L. Deng, F. Janabi-Sharifi, and W. Wilson, "Hybrid motion control and planning strategies for visual servoing," *IEEE Transactions on Industrial Electronics*, vol. 52, no. 4, pp. 1024–1040, 2005.

[89] N. Gans and S. Hutchinson, "Stable visual servoing through hybrid switched-system control," *IEEE Transactions on Robotics*, vol. 23, no. 3, pp. 530–540, 2007.

[90] O. Kermorgant and F. Chaumette, "Combining ibvs and pbvs to ensure the visibility constraint," *IEEE International Conference on Intelligent Robots and Systems*, pp. 2849–2854, 2011.

[91] E. Malis, "Visual servoing invariant to changes in camera-intrinsic parameters," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 72–81, 2004.

[92] F. Schramm, F. Geffard, G. Morel, and A. Micaelli, "Calibration free image point path planning simultaneously ensuring visibility and controlling camera path," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2074–2079, 2007.

[93] Y. Mezouar and F. Chaumette, "Optimal camera trajectory with image-based control," *Int. Journal of Robotics Research, IJRR*, vol. 22, no. 10, pp. 781–804, October 2003.

[94] ——, "Path planning for robust image-based control," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 4, pp. 534–549, 2002.

[95] ——, "Design and tracking of desirable trajectories in the image space by integrating mechanical and visibility constraints," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 1, pp. 731–736, 2001.

[96] N. Cowan, J. Weingarten, and D. Koditschek, "Visual servoing via navigation functions," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 4, pp. 521–533, 2002.

[97] M. Kazemi, K. Gupta, and M. Mehrandezh, "Path-planning for visual servoing: A review and issues," *Lecture Notes in Control and Information Sciences*, vol. 401, pp. 189–207, 2010.

[98] A. Hafez, A. Nelakanti, and C. Jawahar, "Path planning for visual servoing and navigation using convex optimization," *International Journal of Robotics and Automation*, vol. 30, no. 3, pp. 299–307, 2015.

[99] G. Chesi, "Visual servoing path planning via homogeneous forms and lmi omtimizations," *IEEE Transactions on Robotics and Automation*, vol. 25, no. 2, pp. 281–291, 2009.

[100] A. Hafez, A. Nelakanti, and C. Jawahar, "Path planning approach to visual servoing with feature visibility constraints: A convex optimization based solution," *IEEE International Conference on Intelligent Robots and Systems*, pp. 1981–1986, 2007.

[101] R. Sharma and H. Sutanto, "A framework for robot motion planning with sensor constraints," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 1, pp. 61–73, 1997.

[102] H. Zhang and J. Ostrowski, "Visual motion planning for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 2, pp. 199–208, 2002.

[103] G. Chesi, D. Prattichizzo, and A. Vicino, "Straight line path-planning in visual servoing," *Journal of Dynamic Systems, Measurement and Control*, vol. 129, no. 4, pp. 541–543, 2007.

[104] G. Chesi and Y. Hung, "Global path-planning for constrained and optimal visual servoing," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 1050–1060, 2007.

[105] P. Salaris, F. Belo, D. Fontanelli, L. Greco, and A. Bicchi, "Optimal paths in a constrained image plane for purely image-based parking," *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pp. 1673–1680, 2008.

[106] S. Bhattacharya, R. Murrieta-Cid, and S. Hutchinson, "Optimal paths for landmark-based navigation by differential-drive vehicles with field-of-view constraints," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 47–59, 2007.

[107] G. Allibert, E. Courtial, and F. Chaumette, "Predictive control for constrained image-based visual servoing," *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 933–939, 2010.

[108] S. Heshmati-Alamdari, G. Karavas, A. Eqtami, M. Drossakis, and K. Kyriakopoulos, "Robustness analysis of model predictive control for constrained image-based visual servoing," *IEEE International Conference on Robotics and Automation*, pp. 4469–4474, 2014.

[109] G. Allibert, E. Courtial, and Y. Touré, "Visual predictive control for manipulators with catadioptric camera," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 510–515, 2008.

[110] H. L. D. Lee and H. Kim, "Obstacle avoidance using image-based visual servoing integrated with nonlinear model predective control," *IEEE Conf. on Decision and Control and European Control Conference*, pp. 5689–5694, 2011.

[111] E. C. G. Allibert and Y. Toure, "Real-time visual predictive controller for image-based trajectory tracking of a mobile robot," *Proceedings of the 17th IFAC World Congress*, pp. 11 244–11 249, 2008.

[112] C. Lazar, A. Burlacu, and C. Copot, "Predictive control architecture for visual servoing of robot manipulators," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 18, no. PART 1, pp. 9464–9469, 2011.

[113] E. D. M. Sauvee, P. Poignet and E. Courtial, "Image based visual servoing through nonlinear nodel predictive control," *IEEE Conference on Decision and Control*, pp. 1776–1781, 2006.

[114] S. Heshmati-Alamdari, G. Karras, A. Eqtami, and K. Kyriakopoulos, "A robust self triggered image based visual servoing model predictive control scheme for small autonomous robots," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2015-December, pp. 5492–5497, 2015.

[115] A. Mcfadyen, P. Corke, and L. Mejias, "Visual predictive control of spiral motion," *IEEE Transactions on Robotics*, vol. 30, no. 6, pp. 1441–1454, 2014.

[116] A. Eqtami, S. Heshmati-Alamdari, D. Dimarogonas, and K. Kyriakopoulos, "Self-triggered model predictive control for nonholonomic systems," 2013, pp. 638–643.

[117] S. Heshmati-Alamdari, G. Karras, A. Eqtami, and K. Kyriakopoulos, "A robust self triggered image based visual servoing model predictive control scheme for small autonomous robots," vol. 2015-December, 2015, pp. 5492–5497.

[118] S. Heshmati-Alamdari, C. Bechlioulis, M. Liarokapis, and K. Kyriakopoulos, "Prescribed performance image based visual servoing under field of view constraints," *IEEE International Conference on Intelligent Robots and Systems*, pp. 2721–2726, 2014.

[119] C. Bechlioulis, S. Heshmati-Alamdari, G. Karras, and K. Kyriakopoulos, "Robust image based visual servoing with prescribed performance under field of view constraints," *Transactions on Robotics, (Under revision).*

[120] D. Rizzo, F. Bruno, L. Barbieri, and M. Muzzupappa, "Kinematic performances evaluation of a hydraulic underwater manipulator," in *OCEANS 2017-Aberdeen.* IEEE, 2017, pp. 1–6.

[121] F. Zhang, G. Marani, R. Smith, and H. Choi, "Future trends in marine robotics," *IEEE Robotics and Automation Magazine*, vol. 22, no. 1, pp. 14–21 and 122, 2015.

[122] G. Marani, S. Choi, and J. Yuh, "Real-time center of buoyancy identification for optimal hovering in autonomous underwater intervention," *Intelligent Service Robotics*, vol. 3, no. 3, pp. 175–182, 2010.

[123] M. Dunnigan and G. 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.

[124] B. Siciliano, L. Sciavicco, and L. Villani, "Robotics: modelling, planning and control." *Springer Verlag*, 2009.

[125] J. Yuh, "Design and control of autonomous underwater robots: A survey," *Autonomous Robots*, vol. 8, no. 1, pp. 7–24, 2000.

[126] I. Schjølberg and T. I. Fossen, "Modelling and control of underwater vehicle-manipulator systems," in *in Proc. rd Conf. on Marine Craft maneuvering and control.* Citeseer, 1994.

[127] T. J. Tarn, G. Shoults, and S. Yang, "A dynamic model of an underwater vehicle with a robotic manipulator using kane's method," *Autonomous Robots*, vol. 3, no. 2, pp. 269–283, 1996.

[128] 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, no. 8, pp. 1194–1206, 1995.

[129] M. Dunnigan, D. Lane, A. Clegg, and I. Edwards, "Hybrid position/force control of a hydraulic underwater manipulator," *Control Theory and Applications, IEE Proceedings* -, vol. 143, no. 2, pp. 145–151, Mar 1996.

[130] A. Clegg, M. Dunnigan, and D. Lane, "Self-tuning position and force control of an underwater hydraulic manipulator," *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 4, pp. 3226–3231 vol.4, 2001.

[131] J.-H. Ryu, D.-S. Kwon, and P.-M. Lee, "Control of underwater manipulators mounted on an rov using base force information," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, pp. 3238–3243, 2001.

[132] H. Kajita and K. Kosuge, "Force control of robot floating on the water utilizing vehicle restoring force," *Intelligent Robots and Systems, 1997. IROS '97., Proceedings of the 1997 IEEE/RSJ International Conference on*, vol. 1, pp. 162–167 vol.1, Sep 1997.

[133] L. Lapierre, P. Fraisse, and P. Dauchez, "Position/force control of an underwater mobile manipulator," *Journal of Robotic Systems*, vol. 20, no. 12, pp. 707–722, 2003.

[134] Y. Cui, T. Podder, and N. Sarkar, "Impedance Control of Underwater Vehicle-Manipulator Systems (uvms)," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 148–153 vol.1, 1999.

[135] Y. Cui and N. Sarkar, "A unified force control approach to autonomous underwater manipulation," *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, vol. 2, pp. 1263–1268 vol.2, 2000.

[136] G. Antonelli and S. Chiaverini, "Task-priority redundancy resolution for underwater vehicle-manipulator systems," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 1, pp. 768–773, 1998.

[137] D. R. Yoerger, J. G. Cooke, and J.-J. E. Slotine, "Influence of thruster dynamics on underwater vehicle behavior and their incorporation into control system design," *IEEE Journal of Oceanic Engineering*, vol. 15, no. 3, pp. 167–178, 1990.

[138] L. L. Whitcomb and D. R. Yoerger, "Preliminary experiments in model-based thruster control for underwater vehicle positioning," *IEEE Journal of Oceanic Engineering*, vol. 24, no. 4, pp. 495–506, 1999.

[139] R. Bachmayer, L. L. Whitcomb, and M. A. Grosenbaugh, "An accurate four-quadrant nonlinear dynamical model for marine thrusters: Theory and experimental validation," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 146–159, 2000.

[140] A. Liegeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *IEEE transactions on systems, man, and cybernetics*, vol. 7, no. 12, pp. 868–871, 1977.

[141] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, 1987.

[142] P. Baerlocher and R. Boulic, "An inverse kinematics architecture enforcing an arbitrary number of strict priority levels," *The visual computer*, vol. 20, no. 6, pp. 402–417, 2004.

[143] E. Olguin-Diaz, G. Arechavaleta, G. Jarquin, and V. Parra-Vega, "A passivity-based model-free force–motion control of underwater vehicle-manipulator systems," *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1469–1484, 2013.

[144] O. Kanoun, F. Lamiraux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 785–792, 2011.

[145] E. Simetti and G. Casalino, "A novel practical technique to integrate inequality control objectives and task transitions in priority based control," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 84, no. 1-4, pp. 877–902, 2016.

[146] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1006–1028, 2014.

[147] S. Chiaverini, "Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 3, pp. 398–410, 1997.

[148] N. Sarkar and T. Podder, "Coordinated motion planning and control of autonomous underwater vehicle-manipulator systems subject to drag optimization," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 2, pp. 228–239, 2001.

[149] Z. Ismail and M. Dunnigan, "Tracking control scheme for an underwater vehicle-manipulator system with single and multiple sub-regions and sub-task objectives," *IET Control Theory and Applications*, vol. 5, no. 5, pp. 721–735, 2011.

[150] J. Han, J. Park, and W. Chung, "Robust coordinated motion control of an underwater vehicle-manipulator system with minimizing restoring moments," *Ocean Engineering*, vol. 38, no. 10, pp. 1197–1206, 2011.

[151] S. Soylu, B. Buckham, and R. Podhorodeski, "Redundancy resolution for underwater mobile manipulators," *Ocean Engineering*, vol. 37, no. 2-3, pp. 325–343, 2010.

[152] S. Heshmati-alamdari, A. Nikou, K. Kyriakopoulos, and D. Dimarogonas, "A robust force control approach for underwater vehicle manipulator systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 11 197–11 202, 2017.

[153] S. Heshmati-Alamdari, C. P. Bechlioulis, G. C. Karras, A. Nikou, D. V. Dimarogonas, and K. J. Kyriakopoulos, "A robust interaction control approach for underwater vehicle manipulator systems," *Annual Reviews in Control*, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1367578818300543

[154] T. Fossen, "Guidance and control of ocean vehicles," *Wiley, New York*, 1994.

[155] T. Padir and A. Koivo, "Modeling of two underwater vehicles with manipulators on-board," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, pp. 1359–1364, 2003.

[156] N. McClamroch, "Singular systems of differential equations as dynamic models for constrained robot systems." pp. 21–28, 1986.

[157] T. Padir, "Kinematic redundancy resolution for two cooperating underwater vehicles with on-board manipulators," *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, pp. 3137–3142, 2005.

[158] T. Padir and J. Nolff, "Manipulability and maneuverability ellipsoids for two cooperating underwater vehicles with on-board manipulators," *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, pp. 3656–3661, 2007.

[159] R. Cui, S. Ge, B. Voon Ee How, and Y. Sang Choo, "Leader-follower formation control of underactuated autonomous underwater vehicles," *Ocean Engineering*, vol. 37, no. 17-18, pp. 1491–1502, 2010.

[160] G. Pereira, B. Pimentel, L. Chaimowicz, and M. Campos, "Coordination of multiple mobile robots in an object carrying task using implicit communication," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 281–286, 2002.

[161] D. J. Stilwell and B. E. Bishop, "Framework for decentralized control of autonomous vehicles," *In Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2358–2363, 2000.

[162] M. Uchiyama and P. Dauchez, "Symmetric hybrid position/force control scheme for the coordination of two robots." *In Proceedings of the IEEE International Conference on Robotics and Automation.*, pp. 350–356, 1988.

[163] O. Khatib, "Object manipulation in a multi-effector robot system," *in proceeding of the 4th International Symposium on Robotic Research*, vol. 4, pp. 137–144, 1988.

[164] H. Tanner, S. Loizou, and K. Kyriakopoulos, "Nonholonomic navigation and control of cooperating mobile manipulators," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 53–64, 2003.

[165] S. Schneider and J. Cannon, R.H., "Object impedance control for cooperative manipulation: Theory and experimental results," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 383–394, 1992.

[166] J. Luh and Y. Zheng, "Constrained relations between two coordinated industrial robots for motion control." *International Journal of Robotics Research*, vol. 6, no. 3, pp. 60–70, 1987.

[167] T. Sugar and V. Kumar, "Decentralized control of cooperating mobile manipulators," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, pp. 2916–2921, 1998.

[168] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Vehicle/arm coordination and multiple mobile manipulator decentralized cooperation," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2, pp. 546–553, 1996.

[169] W. C. Dickson, R. H. Cannon Jr., and S. M. Rock, "Decentralized object impedance controller for object/robot-team systems: Theory and experiments," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, pp. 3589–3596, 1997.

[170] Y.-H. Liu, S. Arimoto, and T. Ogasawara, "Decentralized cooperation control: non-communication object handling," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2414–2419, 1996.

[171] R. Conti, E. Meli, A. Ridolfi, and B. Allotta, "An innovative decentralized strategy for i-auvs cooperative manipulation tasks," *Robotics and Autonomous Systems*, vol. 72, pp. 261–276, 2015.

[172] R. Furferi, R. Conti, E. Meli, and A. Ridolfi, "Optimization of potential field method parameters through networks for swarm cooperative manipulation tasks," *International Journal of Advanced Robotic Systems*, vol. 13, no. 5, pp. 1–13, 2016.

[173] E. Simetti and G. Casalino, "Manipulation and transportation with cooperative underwater vehicle manipulator systems," *IEEE Journal of Oceanic Engineering*, 2016.

[174] N. Manerikar, G. Casalino, E. Simetti, S. Torelli, and A. Sperindé, "On autonomous cooperative underwater floating manipulation systems," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, 2015.

[175] N. Manerikar, G. Casalino, E. Simetti, S. Torelli, and A. Sperinde, "On cooperation between autonomous underwater floating manipulation systems," *2015 IEEE Underwater Technology, UT 2015*, 2015.

[176] S. Heshmati-Alamdari, C. Bechlioulis, G. Karras, and K. Kyriakopoulos, "Decentralized impedance control for cooperative manipulation of multiple underwater vehicle manipulator systems under lean communication," *IEEE OES Autonomous Underwater Vehicle Symposium, Accepted*, 2018.

[177] ——, "Cooperative impedance control for multiple underwater vehicle manipulator systems under lean communication," *IEEE Journal of Oceanic Engineering, submitted*, 2018.

[178] S. Heshmati-Alamdari, G. Karras, and K. Kyriakopoulos, "A decentralized predictive control approach for cooperative manipulation of multiple underwater vehicle manipulator systems," *IEEE International Conference on Robotics and Automation 2019, Submitted*.

[179] ——, "A decentralized predictive control approach for cooperative transportation by multiple underwater vehicle manipulator systems," *Under preparation*, 2018.

[180] M. W. Spong and M. Vidyasagar, *Robot dynamics and control*. John Wiley & Sons, 2008.

[181] *Naitonal HF RADAR network - surface currents*. [Online]. Available: http://cordc.ucsd.edu/projects/mapping/maps/

[182] R. Smith, Y. Chao, P. Li, D. Caron, B. Jones, and G. Sukhatme, "Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model," *International Journal of Robotics Research*, vol. 29, no. 12, pp. 1475–1497, 2010.

[183] A. Aguiar and A. Pascoal, "Dynamic positioning and way-point tracking of underactuated auvs in the presence of ocean currents," *International Journal of Control*, vol. 80, no. 7, pp. 1092–1108, 2007.

[184] B. Allotta, R. Costanzi, F. Fanelli, N. Monni, L. Paolucci, and A. Ridolfi, "Sea currents estimation during auv navigation using unscented kalman filter," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 13 668–13 673, 2017.

[185] D. Koditschek and E. Rimon, "Robot navigation functions on manifolds with boundary," *Advances in Applied Mathematics*, vol. 11, no. 4, pp. 412–442, 1990.

[186] G.Pin, L. Magni, T. Parisini, and D. Raimondo, "Robust receding-horizon control of nonlinear systems with state dependent uncertainties: an input-to-state approach," *American Control Conference*, pp. 1667 – 1672, 2008.

[187] E. F. Camacho and C. B. Alba, *Model predictive control*. Springer Science & Business Media, 2013.

[188] D. Limon, T. Alamo, D. M. Raimondo, D. M. de la Peña, J. M. Bravo, A. Ferramosca, and E. F. Camacho, *Input-to-State Stability: A Unifying Framework for Robust Model Predictive Control*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–26.

[189] D. L. Marruedo, T. Alamo, and E. Camacho, "Input-to-state stable mpc for constrained discrete-time nonlinear systems with bounded additive uncertainties," *41st IEEE Conf. Decision and Control*, pp. 4619 – 4624, 2002.

[190] S. Garrido-Jurado, R. M. noz Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280 – 2292, 2014.

[191] P. Marantos, Y. Koveos, and K. Kyriakopoulos, "Uav state estimation using adaptive complementary filters," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 4, pp. 1214–1226, 2016.

[192] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, May 2009.

[193] S. G. Johnson, "The nlopt nonlinear-optimization package, http://ab-initio.mit.edu/wiki/index.php/nlopt."

[194] T. Templeton, D. Shim, C. Geyer, and S. Sastry, "Autonomous vision-based landing and terrain mapping using an mpc-controlled unmanned rotorcraft," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 1349–1356, 2007.

[195] K. Kanjanawanishkul and A. Zell, "Path following for an omnidirectional mobile robot based on model predictive control," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3341–3346, 2009.

[196] W. Heemels, K. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," *IEEE 51st Conference on Decision and Control*, pp. 3270–3285, 2012.

[197] D. Antunes, W. Heemels, and P. Tabuada, "Event-triggered control: Performance guarantees and co-design," *51st IEEE Conference on Decision and Control*, pp. 7212–7217, 2012.

[198] K. Åström, "Event based control," *Analysis and Design of Nonlinear Control Systems*, pp. 127 – 147, 2008.

[199] E. Garcia and P. Antsaklis, "Model-based event-triggered control with time-varying network delays," *50th IEEE Conf. Decision and Control & Eur. Control Conf.*, pp. 1650 – 1655, 2011.

[200] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, 2007.

[201] A. Anta and P. Tabuada, "To sample or not to sample: self-triggered control for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2030–2042, 2010.

[202] J. B. Berglind, T. Gommans, and W. Heemels, "Self-triggered mpc for constrained linear systems and quadratic costs," *4th IFAC Nonlinear Model Predictive Control Conference*, 2012.

[203] P. Millán, L. Orihuela, D. M. noz de la Peña, C. Vivas, and F. Rubio, "Self-triggered sampling selection based on quadratic programming," *Proceedings of the 18th IFAC World Congress*, pp. 8896–8901, 2011.

[204] C. Nowzari and J. Cortès, "Self-triggered coordination of robotic networks for optimal deployment," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 1077–1087, 2012.

[205] A. Eqtami, S. Heshmati-alamdari, D. V. Dimarogonas, and K. J. Kyriakopoulos, "Self-triggered model predictive control for nonholonomic systems," *European Control Conference*, 2013, to appear.

[206] S. Heshmati-alamdari, A. Eqtami, G. C. Karras, D. V. Dimarogonas, and K. J. Kyriakopoulos, "A self-triggered visual servoing model predictive control scheme for under-actuated underwater robotic vehicles," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3826–3831, 2014.

[207] S. Durand, J. Guerrero-Castellanos, N. Marchand, and W. Guerrero-S nchez, "Event-based control of the inverted pendulum: Swing up and stabilization," *Control Engineering and Applied Informatics*, vol. 15, no. 3, pp. 96–104, 2013.

[208] C. Santos, M. Mazo Jr., and F. Espinosa, "Adaptive self-triggered control of a remotely operated robot," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7429 LNAI, pp. 61–72, 2012.

[209] S. D. N. M. J.J. Tèllez Guzman, J.F. Guerrero-Castellanos and R. L. Maya, "Event-based lqr control for attitude stabilization of a quadrotor," *15th IFAC Latinamerican Control Conference*, 2012.

[210] J. Guerrero-Castellanos, J. T llez-Guzm n, S. Durand, N. Marchand, J. Alvarez-Mu oz, and V. Gonz lez-D az, "Attitude stabilization of a quadrotor by means of event-triggered nonlinear control," *Journal of Intelligent and Robotic Systems: Theory and Applications*, pp. 1–13, 2013.

[211] R. Postoyan, M. Bragagnolo, E. Galbrun, J. Daafouz, D. Ne? c, and E. Castelan, "Nonlinear event-triggered tracking control of a mobile robot: Design, analysis and experimental results," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 9, no. PART 1, pp. 318–323, 2013.

[212] *The Society of Naval Architects and Marine Engineers, Nomenclature for treating the motion of a submerged body through a fluid (Technical and Research Bulletin No 1 5)*, 1950. [Online]. Available: http://www.sname.org

[213] D.Panagou and K. J. Kyriakopoulos, "Control of underactuated systems with viability constraints," *Proc. of the 50th IEEE Conference on Decision and Control and European Control Conference*, pp. 5497–5502, 2011.

[214] Z.-P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 37, no. 6, pp. 857–869, 2001.

[215] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," *ICRA Workshop on Open Source Software*, 2009. [Online]. Available: http://www.ros.org

[216] K. G. M. Kazemi and M. Mehranzadeh, "Global path planning for robust visual servoing in complex environments," *IEEE Int. Conf. on Robotics and Automation*, pp. 326–332, 2009.

[217] S. Heshmati-alamdari, C. P. Bechlioulis, M. V. Liarokapis, and K. J. Kyriakopoulos, "Prescribed performance image based visual servoing under field of view constraints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, September 2014.

[218] S. Heshmati-alamdari, G. K. Karavas, A. Eqtami, M. Drossakis, and K. J. Kyriakopoulos, "Robustness analysis of model predictive control for constrained image-based visual servoing," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 4469–4474, 2014.

[219] B. Siciliano and J. J. E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," *Advanced Robotics, 1991. 'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on*, pp. 1211–1216 vol.2, 1991.

[220] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The confluence of vision and control*. Springer, 1998, pp. 66–78.

[221] Y.-H. Liu, H. Wang, C. Wang, and K. K. Lam, "Uncalibrated visual servoing of robots using a depth-independent interaction matrix," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 804–817, 2006.

[222] B. Espiau, "Effect of camera calibration errors on visual servoing in robotics," in *The 3rd International Symposium on Experimental Robotics III*. London, UK, UK: Springer-Verlag, 1994, pp. 182–192.

[223] V. Kyrki, D. Kragic, and H. I. Christensen, "Measurement errors in visual servoing," *Robotics and Autonomous Systems*, vol. 54, no. 10, pp. 815–827, 2006.

[224] E. Zergeroglu, D. Dawson, M. De Queiroz, and A. Behal, "Vision-based nonlinear tracking controllers with uncertain robot-camera parameters," *IEEE/ASME Transactions on Mechatronics*, vol. 6, no. 3, pp. 322–337, 2001.

[225] A. Astolfi, L. Hsu, M. Netto, and R. Ortega, "Two solutions to the adaptive visual servoing problem," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 3, pp. 387–392, 2002.

[226] N. Gans, G. Hu, J. Shen, Y. Zhang, and W. Dixon, "Adaptive visual servo control to simultaneously stabilize image and pose error," *Mechatronics*, vol. 22, no. 4, pp. 410–422, 2012.

[227] H. Wang, Y.-H. Liu, and D. Zhou, "Adaptive visual servoing using point and line features with an uncalibrated eye-in-hand camera," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 843–857, 2008.

[228] ——, "Dynamic visual tracking for manipulators using an uncalibrated fixed camera," *IEEE Transactions on Robotics*, vol. 23, no. 3, pp. 610–617, 2007.

[229] X. Zhong, X. Zhong, and X. Peng, "Robots visual servo control with features constraint employing kalman-neural-network filtering scheme," *Neurocomputing*, vol. 151, no. P1, pp. 268–277, 2015.

[230] Z. Miljković, M. Mitić, M. Lazarević, and B. Babić, "Neural network reinforcement learning for visual control of robot manipulators," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1721–1736, 2013.

[231] E. Malis and P. Rives, "Robustness of image-based visual servoing with respect to depth distribution errors," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 1, pp. 1056–1061, 2003.

[232] E. Malis and F. Chaumette, "Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 2, pp. 176–186, 2002.

[233] F. Schramm and G. Morel, "Ensuring visibility in calibration-free path planning for image-based visual servoing," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 848–854, 2006.

[234] A. Farahmand, A. Shademan, and M. Jägersand, "Global visual-motor estimation for uncalibrated visual servoing," *IEEE International Conference on Intelligent Robots and Systems*, pp. 1969–1974, 2007.

[235] G. Morel, P. Zanne, and F. Plestan, "Robust visual servoing: Bounding the task function tracking errors," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 6, pp. 998–1009, 2005.

[236] C. Bechlioulis and G. Rovithakis, "Robust adaptive control of feedback linearizable mimo nonlinear systems with prescribed performance," *IEEE Transactions on Automatic Control*, vol. 53, no. 9, pp. 2090–2099, 2008.

[237] ——, "Adaptive control with guaranteed transient and steady state tracking error bounds for strict feedback systems," *Automatica*, vol. 45, no. 2, pp. 532–538, 2009.

[238] ——, "Prescribed performance adaptive control for multi-input multi-output affine in the control nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1220–1226, 2010.

[239] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, 1992.

[240] H. K. Khalil, *Nonlinear Systems*, 1st ed. Macmillan, 1992.

[241] G. Antonelli, S. Chiaverini, and N. Sarkar, "External force control for underwater vehicle-manipulator systems," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 6, pp. 931–938, 2001.

[242] K. L. Johnson and K. L. Johnson, *Contact mechanics*. Cambridge university press, 1987.

[243] S. Arimoto, P. T. A. Nguyen, H.-Y. Han, and Z. Doulgeri, "Dynamics and control of a set of dual fingers with soft tips," *Robotica*, vol. 18, no. 1, pp. 71–80, 2000.

[244] 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.

[245] L. Sciavicco and B. Siciliano, *Modelling and control of robot manipulators*. Springer Science & Business Media, 2012.

[246] C. Bechlioulis and G. Rovithakis, "Prescribed performance adaptive control for multi-input multi-output affine in the control nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 5, pp. 1220–1226, 2010.

[247] C. P. Bechlioulis and G. A. Rovithakis, "Robust partial-state feedback prescribed performance control of cascade systems with unknown nonlinearities," *IEEE Transactions on Automatic Control*, vol. 56, 2011.

[248] C. Bechlioulis and G. Rovithakis, "A low-complexity global approximation-free control scheme with prescribed performance for unknown pure feedback systems," *Automatica*, vol. 50, no. 4, pp. 1217–1226, 2014.

[249] Seabotix lbv150. [Online]. Available: http://www.teledynemarine.com/lbv150-4/?BrandID=19

[250] K. Kosuge, T. Oosumi, and H. Seki, "Decentralized control of multiple manipulators handling an object in coordination based on impedance control of each arm," *IEEE International Conference on Intelligent Robots and Systems*, vol. 1, pp. 17–22, 1997.

[251] K. Kosuge, T. Oosumi, and K. Chiba, "Load sharing of decentralized-controlled multiple mobile robots handling a single object," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, 1997.

[252] B. Siciliano, L. Sciavicco, and L. Villani, *Robotics : modelling, planning and control*, ser. Advanced Textbooks in Control and Signal Processing. Springer, 2009, 013-81159.

[253] J.-J. E. Slotine and W. Li, "Adaptive strategies in constrained manipulation." pp. 595–601, 1987.

[254] E. Tatlicioglu, D. Braganza, T. Burg, and D. Dawson, "Adaptive control of redundant robot manipulators with sub-task objectives," *Robotica*, vol. 27, no. 6, pp. 873–881, 2009.

[255] A. De Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE, 2005, pp. 999–1004.

[256] C. Bechlioulis and G. Rovithakis, "Robust partial-state feedback prescribed performance control of cascade systems with unknown nonlinearities," *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2224–2230, 2011.

[257] J. Gudiño-Lau, M. A. Arteaga, L. A. Munoz, and V. Parra-Vega, "On the control of cooperative robots without velocity measurements," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 4, pp. 600–608, 2004.

[258] F. Caccavale, P. Chiacchio, and S. Chiaverini, "Task-space regulation of cooperative manipulators," *Automatica*, vol. 36, no. 6, pp. 879–887, 2000.

[259] H. Farivarnejad and S. A. A. Moosavian, "Multiple impedance control for object manipulation by a dual arm underwater vehicle–manipulator system," *Ocean Engineering*, vol. 89, pp. 82–98, 2014.

[260] S. A. A. Moosavian and E. Papadopoulos, "Multiple impedance control for object manipulation," in *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, vol. 1. IEEE, 1998, pp. 461–466.

[261] ——, "Cooperative object manipulation with contact impact using multiple impedance control," *International Journal of Control, Automation and Systems*, vol. 8, no. 2, pp. 314–327, 2010.

[262] M. Prats, J. Perez, J. Fernandez, and P. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 2577–2582.

[263] G. C. Karras, P. Marantos, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Unsupervised online system identification for underwater robotic vehicles," *IEEE Journal of Oceanic Engineering*, 2018.

[264] E. D. Sontag, *Mathematical Control Theory.* London, U.K.: Springer, 1998.

[265] M. Logothetis, , G. Karras, S. Heshmati-Alamdari, P. Vlantis, and K. Kyriakopoulos, "A model predictive control approach for vision-based object grasping via mobile manipulator," 2018, Accepted.

[266] A. Eqtami, S. Heshmati-Alamdari, D. Dimarogonas, and K. Kyriakopoulos, "Self-triggered model predictive control framework for the cooperation of distributed nonholonomic agents," *Proceedings of the IEEE Conference on Decision and Control*, pp. 7384–7389, 2013.

[267] N. Hurtós, N. Palomeras, A. Carrera, M. Carreras, C. Bechlioulis, G. Karras, S. Hesmati-Alamdari, and K. Kyriakopoulos, "Sonar-based chain following using an autonomous underwater vehicle," 2014, pp. 1978–1983.

[268] A. Nikou, C. Verginis, S. Heshmati-Alamdari, and D. Dimarogonas, "A nonlinear model predictive control scheme for cooperative manipulation with singularity and collision avoidance," 2017, pp. 707–712.

[269] A. Nikou, S. Heshmati-Alamdari, C. Verginis, and D. Dimarogonas, "Decentralized abstractions and timed constrained planning of a general class of coupled multi-agent systems," vol. 2018-January, 2018, pp. 990–995.