# National Technical University of Athens

School of Mechanical Engineering

Inter-Departmental Program of Postgraduate Studies

**Automation Systems**




# Path planning for Underwater Vehicle - Manipulator Systems based on task-specific pose configurations for efficient manipulation


**NIKOLAOS A. KAMPRAS**

(Diploma in Electrical and Computer Engineering, N.T.U.A.)


Submitted in partial fulfillment of the requirements for the degree of

**MSc in Control and Robotic Systems**

Master Thesis Advisor: Professor Kostas J. Kyriakopoylos

Athens, October 2013

[This page intentionally left blank.]

# Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Μηχανολόγων Μηχανικών

Διατμηματικό Πρόγραμμα Μεταπτυχιακών Σπουδών

**Συστήματα Αυτοματισμού**




# Σχεδιασμός πορείας Υποβρύχιων Συστημάτων Οχήματος – Βραχίονα βασισμένος σε κατάλληλες διατάξεις για αποτελεσματικό χειρισμό


## ΝΙΚΟΛΑΟΣ Α. ΚΑΜΠΡΑΣ

(Διπλωματούχος Ηλεκτρολόγος Μηχανικός
και Μηχανικός Υπολογιστών, Ε.Μ.Π.)


Κατατέθηκε για τη μερική εκπλήρωση των υποχρεώσεων για
την απόκτηση του διπλώματος με τίτλο

**Μεταπτυχιακό Πρόγραμμα Ειδίκευσης**

στα

**Συστήματα Αυτομάτου Ελέγχου και Ρομποτικής**

Υπεύθυνος Καθηγητής: Κωνσταντίνος Ι. Κυριακόπουλος

Αθήνα, Οκτώβριος 2013

# Path planning for Underwater Vehicle - Manipulator Systems based on task-specific pose configurations for efficient manipulation

## Master of Science Thesis



**Contact info:**

**nikkampras@gmail.com**

Submitted in Partial Fulfillment of the Requirements

for the Degree of MSc in Control and Robotic Systems

in the School of Mechanical Engineering at the

National Technical University of Athens, 2013

Athens, Greece

*Στη μνήνη του παππού μου,*

*Γιώργου*

# Περίληψη*

Η ανάπτυξη αυτόνομων υποβρύχιων ρομποτικών συστημάτων με στόχο την επίτευξη υποθαλάσσιων επεμβατικών εργασιών είναι ένα πολύ σημαντικό θέμα έρευνας με ραγδαία ανάπτυξη στο πεδίο της ρομποτικής. Για την εκτέλεση αποστολών που απαιτούν αλληλεπίδραση με το φυσικό περιβάλλον χρησιμοποιούνται υποβρύχια οχήματα εξοπλισμένα με ρομποτικό βραχίονα (Underwater Vehicle - Manipulator System, UVMS). Όπως είναι αναμενόμενο, πολλά ενδιαφέροντα ζητήματα προκύπτουν κατά την σχεδίαση τεχνικών ελέγχου αυτόνομων UVMS, στοχεύοντας στην βελτίωση της ικανότητας τους για αλληλεπίδραση.

Το αντικείμενο αυτής της μεταπτυχιακής εργασίας είναι η ανάπτυξη ενός προβλήματος βελτιστοποίησης από το οποίο προκύπτει η βέλτιστη διάταξη ενός UVMS όσον αφορά στην αποτελεσματική αλληλεπίδραση με το φυσικό περιβάλλον λαμβάνοντας υπ' όψιν ότι πρόκειται για ένα σύστημα με πλεονάζοντες βαθμούς ελευθερίας. Ταυτόχρονα, διασφαλίζεται ότι διαφόρων ειδών περιορισμοί ικανοποιούνται. Αυτή η βέλτιστη διάταξη του UVMS αντιστοιχεί στις απαιτήσεις της εκάστοτε εργασίας και σε συγκεκριμένο κριτήριο βελτιστοποίησης το οποίο εξασφαλίζει την μεγιστοποίηση μίας συγκεκριμένης νόρμας του διανύσματος γενικευμένης δύναμης/ροπής που μπορεί να ασκήσει ο βραχίονας.

Το παραπάνω πρόβλημα βελτιστοποίησης συμπεριλαμβάνεται σε έναν αλγόριθμο σχεδιασμού δρόμου για UVMS που αλληλεπιδρούν με το περιβάλλον. Δεδομένου ότι η διαδικασία ελέγχου ενός UVMS πραγματοποιείται στο χώρο των αρθρώσεων, ο προτεινόμενος αλγόριθμος παρέχει σε πραγματικό χρόνο μία ακολουθία βέλτιστων διατάξεων η οποία οδηγεί σταδιακά το σύστημα στην τελική βέλτιστη θέση και διάταξη προκειμένου να μπορεί να ασκήσει την επιθυμητή γενικευμένη δύναμη/ροπή στο φυσικό περιβάλλον.

Λαμβάνοντας υπ' όψιν ότι οι μη γραμμικοί περιορισμοί που πρέπει να ικανοποιούνται καθιστούν την διαδικασία βελτιστοποίησης πολύπλοκη και υπολογιστικά απαιτητική, προτείνεται ένας εναλλακτικός αλγόριθμος σχεδιασμού δρόμου στο χώρο των

* Λόγω της πληθώρας εξειδικευμένων όρων που εμπεριέχονται στην παρούσα εργασία, οι οποίοι δυστυχώς δεν τυγχάνουν επιτυχημένης μετάφρασης στα Ελληνικά, κρίθηκε προτιμότερο αυτή να γραφτεί εξ' ολοκλήρου στην Αγγλική γλώσσα. Παρ' όλα αυτά, σε αυτήν την ενότητα περιέχεται μια σύντομη περίληψη στα Ελληνικά.

αρθρώσεων. Αυτή η προσέγγιση επιταχύνει τη διαδικασία υπολογισμού και φαίνεται να είναι καταλληλότερη για τεχνικές σχεδιασμού δρόμου σε πραγματικό χρόνο. Η βασική ιδέα της έγκειται στην εφαρμογή της ανάλυσης ευαισθησίας σε μία επαναληπτική διαδικασία προκειμένου να υπολογιστεί η ακολουθία βέλτιστων διατάξεων του UVMS.

Η απόδοση των προτεινόμενων αλγόριθμων σχεδιασμού δρόμου και του προβλήματος βελτιστοποίησης ως ένα μέρος του συνολικού σχήματος ελέγχου αλληλεπίδρασης καταδεικνύεται μέσω μιας σειράς προσομοιώσεων στο MATLAB όπου διάφορα σενάρια μελετώνται. Το αυτόνομο UVMS, το μοντέλο του οποίου υιοθετείται στις προσομοιώσεις, αποτελείται από το Girona500 AUV και το βραχίονα ARM 5E Micro.

**Λέξεις Κλειδιά:** Υποβρύχια Συστήματα Οχήματος-Βραχίονα (UVMS), Έλεγχος αλληλεπίδρασης, Μη Γραμμικός προγραμματισμός, Αλγόριθμος σχεδιασμού δρόμου, Ανάλυση ευαισθησίας

# Abstract

The development of completely autonomous underwater robotic systems to accomplish complex subsea intervention tasks is one of the most important and rapidly increasing topics in underwater robotics research field. In case of missions that require interaction with the environment, underwater vehicles are equipped with a robotic arm to perform manipulation tasks; in this case the system is usually called Underwater Vehicle-Manipulator System (UVMS). As expected, many challenging issues arise from designing interaction control schemes for autonomous UVMS, aiming at the improvement of their intervention capability.

The scope of this master thesis is the development of an optimization scheme that provides the optimal hovering pose configuration of an UVMS for efficient interaction with the environment, exploiting the redundant dofs of the combined system and ensuring that several constraints are satisfied. This optimal hovering pose configuration corresponds to several intervention requirements and a certain performance criterion ensuring the maximization of a meaningfully defined norm of end-effector interaction wrench vector.

This optimization scheme is incorporated as part of a path planning scheme for UVMS interacting with the environment. Since the control action on the UVMS is carried out in the joint space, a suitable algorithm is proposed to provide in real time a sequence of UVMS pose configurations that leads in a smooth way to the final optimal configuration for efficient interaction wrt the pre-specified performance criterion. Thus, this algorithm includes the aforementioned optimization scheme and plays the role of an on-line path planner in the joint space of the UVMS during the "*reach-to-grasp*" phase.

Considering that the nonlinear constraints to be respected make the optimization procedure complicated and computationally intense, a second path planning scheme in the joint space of the UVMS is proposed. This approach speeds up the computation procedure and seems to be more convenient for on-line motion planning schemes. The key idea behind this approach lies in applying sensitivity analysis in an iterative process to derive the sequence of optimal UVMS pose configurations.

Finally, the performance of the developed motion planning algorithms and optimization scheme as part of the overall interaction control scheme is demonstrated through a series of simulation studies in MATLAB where various underwater scenarios are considered. The autonomous UVMS, used for the simulation studies, is composed of the Girona500 AUV and the ARM 5E Micro manipulator.

**Keywords:** Underwater Vehicle-Manipulator Systems (UVMS), Interaction control, Nonlinear Programming Problem, Path planning scheme, Sensitivity analysis

# Acknowledgements

First of all, I would like to thank my advisor Professor Kostas J. Kyriakopoulos, for our collaboration over the past year. Through this I was given the chance to develop my background, while being introduced in the challenging field of robotics and automation. His deep knowledge and long experience in the field were of indispensable value during the problem-solving process and brought me new insight on how to accomplish my goals. Furthermore, his strictly structured working scheme inspired me and comprised a valuable pattern during my first steps in research, showing me the way to success. For all those things I would like to express my frank gratitude.

Special thanks should be given to the postdoctoral researcher Charalampos Bechlioulis, as an essential consultant during the elaboration of this task. His support and guidance, both at a technical and personal level, were of decisive importance to help me overcome difficulties that arose, improve my skills, get more effective and finally reach my goals. I definitely consider myself lucky to have had the opportunity to work together with such a great scientist and great character at the same time.

To continue with, I would like to thank the whole team of the Control System Lab, namely Shahab Heshmati, George Karavas, George Karras, Alina Eqtami, George Zogopoulos and Alexandros Nikou, for being valuable colleagues, quite cooperative and all sustaining a very pleasant atmosphere to work in.

Moreover, I would like to express my deep gratitude to my parents, Aggelos and Helen, for offering their unreserved support from all aspects during all those years of my studies. Without them standing by my side, many of my greatest achievements up to now would have just remained a dream. I would also like to thank a close friend of mine, George, for his valuable support and company that made the hard work of those past months feel a little lighter and easier to handle. Last but not least, I would like to thank a very special person, Ntora, for her endless encouragement and faith in me. I cannot find the words to express my love and gratitude. I will just say *Ευχαριστώ*.

Nikolaos A. Kampras

Athens, October 2013

# List of Figures

# List of Tables

# Contents

# CHAPTER 1

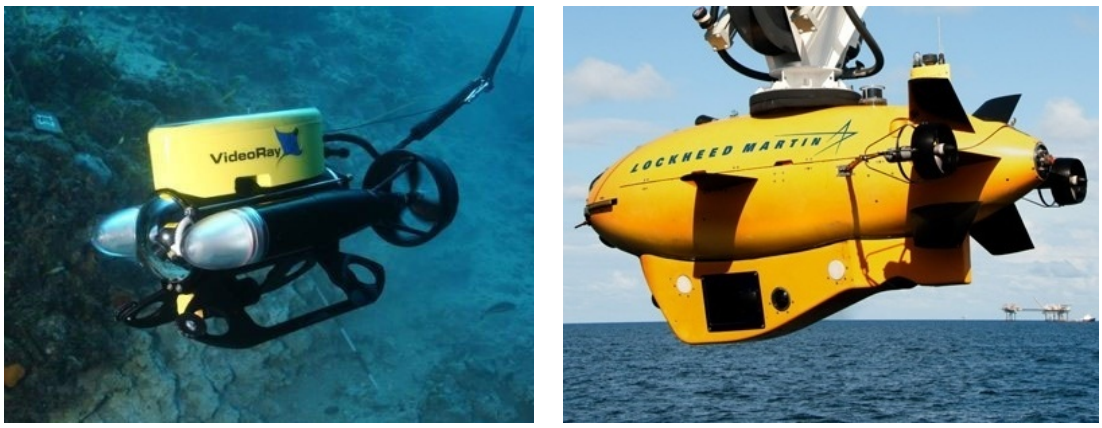
# Preface

## 1.1 Introduction

**Underwater Vehicles**

Exploring underwater environments in traditional ways using manned systems and human divers is unequivocally a quite difficult and hazardous task. Therefore, Unmanned Underwater Vehicles (UUV), that can perform difficult missions without risking human lives, are gradually becoming popular and effective tools to help people see and touch this unfamiliar world. Currently, underwater vehicles are utilized in many fields, such as in the scientific, military and industrial ones. In the scientific field, they are often used in oceanographic researches to map seafloors and monitor ocean environments, while in the military field they are used for surveillance and reconnaissance. Finally, in the industrial field they serve as a means for surveying and exploring undersea resources as well as for inspecting and maintaining offshore/subsea plants and structures.

Such unmanned underwater vehicles are often classified into two types - Remotely Operated Vehicles (ROV) and Autonomous Underwater Vehicles (AUV). The term ROV denotes an underwater vehicle physically linked (Fig. 1.1a), via the tether, to an operator that can be on a mother surface ship or submarine. The tether is in charge of giving power to the vehicle as well as closing the manned control loop. Such vehicles are suitable for performing energy-intensive tasks, but their range of activities and speed capabilities are limited. On the other hand, AUV may be defined as a not-tethered holonomic (i.e. hover capable) or non-holonomic (i.e. torpedo shape, propulsor at the rear) vehicle (Fig. 1.1b) having the potential to operate autonomously, at low cost and unconstrained by various disturbances. Therefore, their power capabilities and operating times are limited, relying on onboard power system and intelligence. However, AUV are superior to ROV in speed and coverage, requiring little or no human supervision. In [1-3] the state of the art of several existing AUV and their control architectures are presented.




(a) (b)

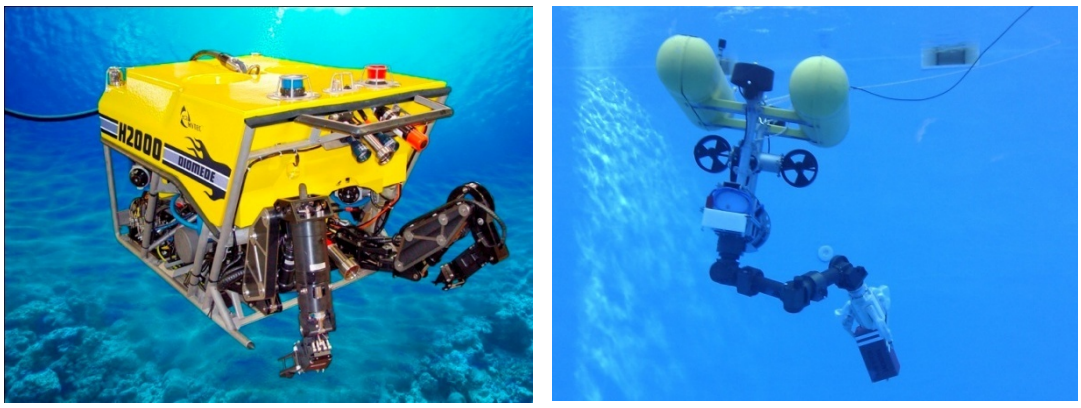
**Figure 1.1:** (a) VideoRay Pro 3 ROV [4], (b) Marlin AUV [5].

**Underwater Vehicle – Manipulator Systems (UVMS)**

In case of missions that require interaction with the environment, UUV can be equipped with one or more manipulators to perform complex underwater tasks; in this case the system is usually called Underwater Vehicle-Manipulator System (UVMS) that correspond to redundant non-inertial (i.e. free-flying base) robotic arms. The growing interest in undersea technology has resulted in the development of many new approaches for improving the undersea intervention capability of UVMS. Developments and studies on UVMS modeling and control can be found in the literature [3, 6-37].

Currently, the state of the art of underwater manipulation is based on ROV carrying a tele-operated manipulator (Fig. 1.2a) [38-40]. Human operators are in charge of remotely controlling the vehicle actuators and the manipulator by, e.g., a master–slave technique. This human-piloted scheme causes several operational difficulties, including inaccurate trajectory tracking and force control, time delays in the man–machine control loop, and fatigue and reduced performance of the human operator. Moreover, the operators are often required to be physically near to the vehicle-manipulator system, which raises the risks and costs involved with the mission to be executed.

To overcome the above problems, research efforts are aimed at developing completely autonomous UVMS, using the so-called intervention AUV (Fig. 1.2b). This fascinating scenario, where I-AUV do the work autonomously, comes at the cost of endowing the robot with the intelligence needed to keep the operator out of the control loop. UVMS must be capable of assessing a situation, including self-calibration based on sensory information, and executing or revising a course of manipulating action without continuous human intervention and supervision.




(a) (b)

**Figure 1.2:** (a) H2000 ROV equipped with 2 manipulator arms [41],
(b) TRIDENT project I-AUV [42].

In particular, the UVMS must simultaneously control the position of the end-effector and the force applied to the environment. Therefore, one of the basic tasks to be accomplished is handling of motion coordination of the bodies constituting the UVMS considering the nonlinear and coupled dynamics between underwater vehicles and their manipulator systems. A detailed mathematical model of an UVMS is required to develop a control scheme that considers the nature of the system. Another main issue in designing and implementing a control system for autonomous manipulation of an UVMS is ensuring a reliable behavior within the workspace, and avoiding collisions, system instabilities and unwanted motions while performing the required task that is theoretically executable. Furthermore, UVMS must not be carried away by ocean currents. However, this is not an easy task to accomplish in practice. Many underwater vehicles are underactuated, which implies that the system has fewer control inputs than the number of degrees of freedom (dofs). As a result, it gets complicated to control such systems when they are exposed to environmental forces. Hence, the controller for dynamic positioning of underactuated vehicles must be designed considering the nature of underactuated systems.

In conclusion, UVMS are complex systems characterized by several strong constraints that must be taken into account when designing a interaction control scheme:

- Uncertainty in the model knowledge, mainly due to the poor description of the hydrodynamic effects
- Complexity of the mathematical model
- Structural redundancy of the system
- Difficulty to control the vehicle in hovering, mainly due to the poor thrusters' performance

- Dynamic coupling between vehicle and manipulator
- Low sensors' bandwidth
- Underactuation

In literature there are only few examples of UVMS such as ODIN, OTTER I-AUV, SAUVIM [6], VORTEX [19], ALIVE [43] and RAUVI [44].

## 1.2 Verbal Problem Statement

In the framework of the development of completely autonomous UVMS, one of the main issues to be resolved is handling of motion coordination between the vehicle and the manipulator to successfully execute intervention tasks. Some common underwater intervention tasks to be accomplished by UVMS could be the manipulation of valves and switches on underwater facilities such as control panels on hydrocarbon underwater sites, the inspection and maintenance of underwater structures and the object recovery and sampling from the sea bottom. In order to perform an intervention task the vehicle has to temporarily dock or hover near the desired target point. The determination of the optimal docking/hovering position of the vehicle or more precisely the appropriate pose configuration of the UVMS wrt several intervention tasks' requirements is a matter of increased interest in the research field of undersea intervention technology and has attracted a lot of attention during the last years [45-47].

When UVMS are expected to perform underwater intervention tasks two issues arise even if the desired target point (i.e. end-effector pose) is considered as known:

- In most cases, although the interaction wrench vector (combined force and torque) may be known in terms of direction, based on a priori information (e.g. CAD plans of a valve in a particular installation), sensory information or a fusion of them, it is usually unknown in terms of magnitude (e.g. the torque needed to turn a valve may change in time due to corrosion etc.).
- When approaching the interaction spot (i.e. during the "*reach-to-grasp*" phase) or when interacting, disturbances possibly induced by underwater currents should be rejected.

Thus, since the UVMS actuators: (a) will need to "undertake" simultaneously those issues and (b) have limited capability range, one should consider the problem of appropriately posing the combined vehicle–manipulator system in such a way that it

allows for “maximization” of the allowed interaction wrench between the UVMS and the environment (solid or fluid) since this wrench has to be compensated by the UVMS actuation system.

Furthermore, it must be noted that the UVMS used to perform the manipulation task are kinematically redundant systems due to the mobility provided by the vehicle itself in addition to that provided by the manipulator arm. In other words, UVMS possess more dofs in the joint-space than those strictly required to execute a task specified in the Cartesian space.

Consequently, the ultimate objective of this thesis is the design of an optimization scheme that provides an optimal pose configuration of UVMS for efficient interaction with the environment wrt a certain performance criterion (maximization of a meaningfully defined norm of interaction wrench vector), exploiting the redundant dofs of the combined system and ensuring that several constraints are satisfied.

This work has been inspired by the PANDORA research project [48]. The autonomous UVMS the interaction control scheme is designed for is composed of the Girona500 AUV and the ARM 5E Micro manipulator (Fig. 1.3). The UVMS‘s goal is to locate the correct valve panel of a subsea manifold, grasp the correct valve and open it. On each panel a selection of valve heads are exposed, each with a T bar attached for grasping. The vehicle identifies the state of the valves (open, close, in-between) from the T bar orientations, and if appropriate, uses the gripper to grasp the correct valve and open it. It does not dock, because there are no docking bars on the panel, thus it hovers near the valve panel, counteracting by thrusters any reaction forces from the turning as well as any disturbances possibly induced by underwater currents.

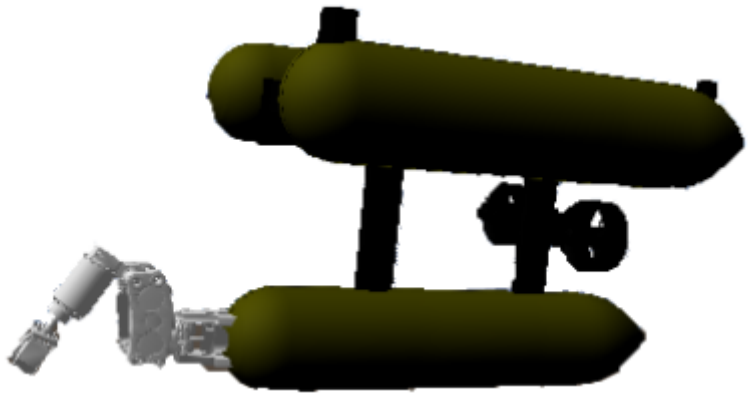


**Figure 1.3:** The integrated UVMS prototype composed of GIRONA 500 AUV and Light-Weight ARM 5E.

## 1.3 Approach of Solution

As it has already been mentioned, the goal of this thesis is the design of an optimization scheme that provides an optimal pose configuration of UVMS for efficient interaction with the environment wrt a certain performance criterion. The proposed algorithm will be incorporated as part of a two-stage interaction control structure for UVMS performing intervention tasks that involve interaction with the environment.

For autonomous manipulation in water, an UVMS should be able to generate trajectories for the vehicle and manipulators in real time and track the planned trajectories accurately. Thus, the two-stage interaction control structure is composed of a path or motion planning algorithm and a robust coordinated motion control algorithm. The motion planning algorithm is formulated as follows.

A manipulation task is usually given in terms of position and orientation trajectory of the end effector. When the UVMS approaches the interaction spot (i.e. during the "*reach-to-grasp*" phase), an image based controller using model predictive control (MPC) is used to generate a sequence of variables over time that describe end-effector position and orientation, navigating it towards the target point. Since the control action on the UVMS is carried out in the joint space, a suitable algorithm is to be used to provide in real time the joint space variables corresponding to the desired end-effector pose denoted by the image based controller.

Considering the redundancy of the combined system vehicle-manipulator, this problem has an infinite number of solutions. Therefore, it can be formulated as a nonlinear constrained optimization problem due to the nonlinear nature of the constraints that need to be satisfied. In particular, the optimization algorithm will take as input the desired end-effector pose and will generate an appropriate pose configuration of the UVMS for efficient interaction with the environment. In accordance with the problem verbally stated in §1.2, the UVMS configuration must be optimal wrt a objective function, the maximization of which guarantees the maximization of a meaningfully defined norm of the interaction wrench vector. At the same time, constraints imposed by the mechanical structure of the UVMS and the surrounding physical environment as well as hardware limitations of the system concerning the limited capability range of the actuators need to be satisfied and are considered in the optimization scheme.

It must be noted that, in fact, the image based controller produces in real time a sequence of end-effector poses, from the initial pose when the UVMS enters the "*reach-to-grasp*" phase to the final desired pose when the end effector is ready to interact with

the environment. In parallel, the proposed optimization scheme produces a sequence of UVMS pose configurations that leads in a smooth way to the final optimal configuration for efficient interaction wrt the pre-specified performance criterion.

As customary in kinematic control approaches, the output of the motion planning algorithm (i.e. the integration of the image based controller with the proposed optimization scheme) constitutes the reference joint space variables of the UVMS motion controller.

In Figure 1.4 the two-stage interaction control structure described above is schematically represented using a block diagram format. The efficacy of the developed optimization algorithm as part of the overall interaction control scheme will be demonstrated through a series of simulation studies in MATLAB where various navigation scenarios will be considered for the model of the UVMS used in the PANDORA research project.

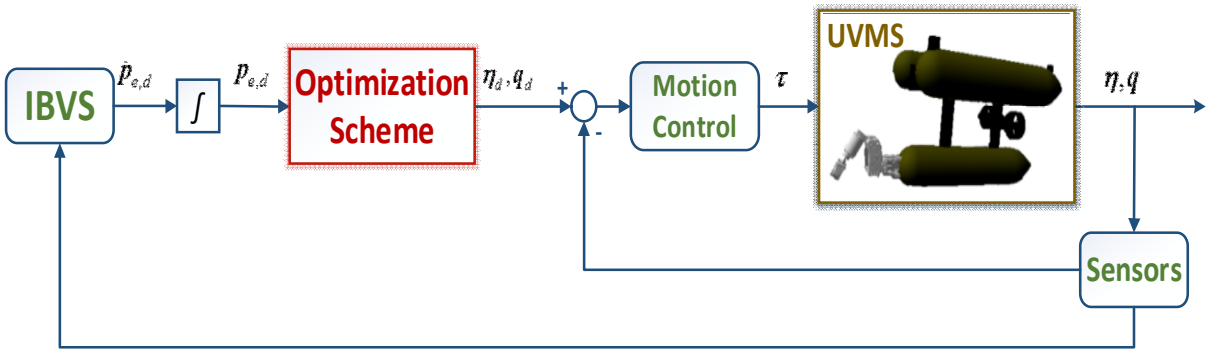



**Figure 1.4:** Block diagram of the interaction control structure including the proposed optimization scheme.

## 1.4 Thesis Structure

The rest of this thesis is organized as follows:

- In Chapter 2 an analysis is made on the kinematics and dynamics of an UVMS. At first, several reference frames are defined in §2.1. In §2.2 the UVMS differential kinematics equation is derived. Consequently, §2.3 presents and briefly explains the dynamic equations of motion of an UVMS. Finally, the vector of gravity and buoyancy generalized forces is mathematically expressed in §2.4.

- In Chapter 3 the proposed motion planning algorithms and optimization scheme are presented. In §3.1, the problem of producing the appropriate UVMS pose configuration for efficient interaction with the environment is being modeled. A summary of all the notations adopted is also provided. For the sake of completeness, §3.2 presents basic elements of Nonlinear Programming to provide some background on this field. Finally, in §3.3 and §3.4 two approaches of solution process including the proposed optimization scheme are described.
- In Chapter 4 the performance of the developed motion planning algorithms and optimization scheme as part of the overall interaction controller is demonstrated through a series of simulation studies in MATLAB where various underwater scenarios are considered. In §4.1 the UVMS used for the simulations studies is described. Consequently, the simulation results and performance of the proposed algorithms are provided in §4.2 and §4.3, where the UWSim simulator is used for illustrative purposes.
- Finally, in Chapter 5 an overall description of this work is presented citing the problems encountered and the goals achieved. Moreover, suggestions for further work are stated.

# Chapter 2

# Modelling of UVMS

## 2.1 Reference Frames

In order to analyze the motion of UVMS in 3D-space, several reference frames need to be defined. As in all problems in robotics, various quantities are represented in different coordinate frames and there is the need for transformation of them between frames. In Figure 2.1 a sketch of an UVMS with reference frames is shown. Consider an underwater vehicle with an *n*-link mounted manipulator.

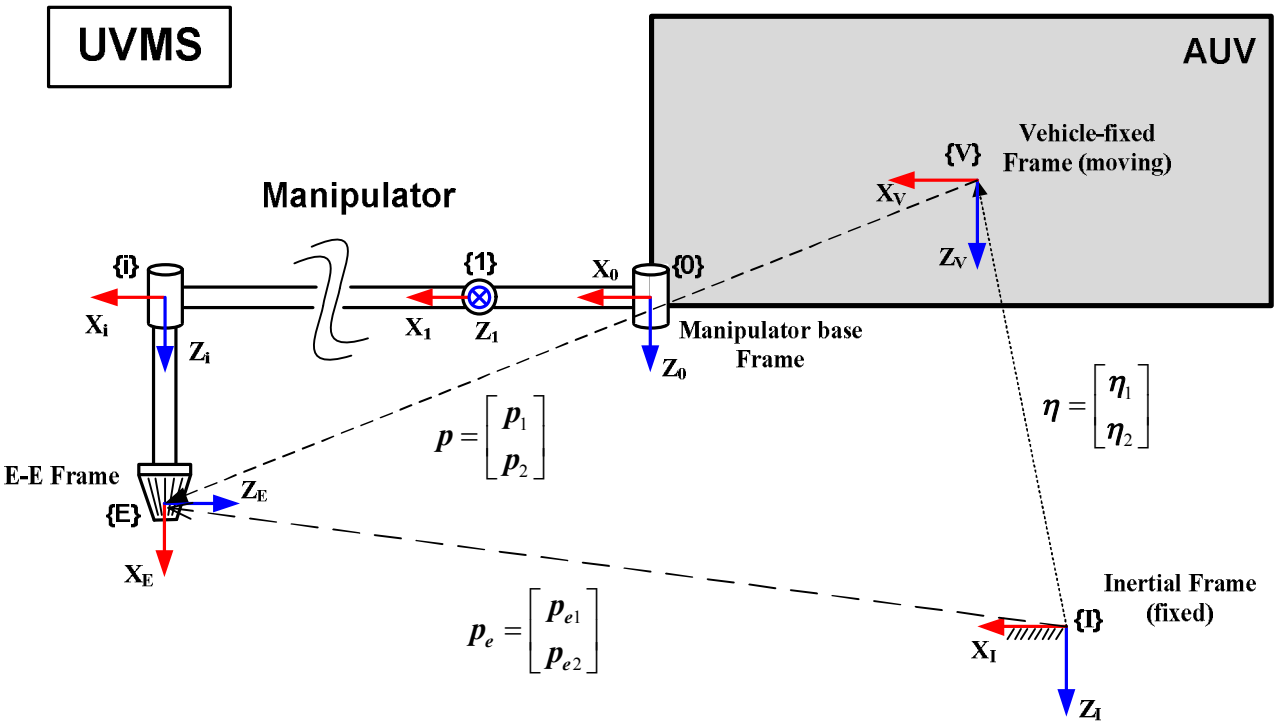



**Figure 2.1:** Schematic diagram of UVMS with coordinate frames attached.

**Earth-fixed frame {*I*}**

Frame {*I*} denotes the earth-fixed reference frame with the $X_I$ axis directing North and $Z_I$ axis pointing straight down towards the earth, aligned with the gravity direction. Considering that the motion of the Earth hardly affects low speed marine vehicles, the earth-fixed reference frame {*I*} can be regarded as the inertial.

**Vehicle-fixed frame {*V*}**

The moving coordinate frame {*V*} is conveniently fixed to the AUV and is called the body-fixed or vehicle-fixed reference frame. The origin of the body-fixed frame is usually chosen to coincide with the center of gravity ($CG_V$) of the vehicle when $CG_V$ is in the principal plane of symmetry. The body axes coincide with the principal axes of inertia. Thus, the longitudinal axis $X_V$ is directed from aft to fore, the transverse axis $Y_V$ is directed to starboard and the normal axis $Z_V$ is directed from top to bottom.

**Manipulator frames**

The 0-th frame, {0}, is the reference frame 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 {*E*}.

## 2.2 Kinematics of UVMS

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.

The vehicle position and orientation are described relative to the inertial reference frame {*I*}, while the linear and angular velocities of the vehicle are expressed in the vehicle-fixed frame {*V*}.

Let define the vehicle pose vector $\boldsymbol{\eta}$ as:

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\eta}_1^T & \boldsymbol{\eta}_2^T \end{bmatrix}^T \in \mathbb{R}^6 \quad \begin{matrix} \nearrow \; \boldsymbol{\eta}_1 = \begin{bmatrix} x & y & z \end{bmatrix}^T \in \mathbb{R}^3 \\ \searrow \; \boldsymbol{\eta}_2 = \begin{bmatrix} \varphi & \theta & \psi \end{bmatrix}^T \in \mathbb{R}^3 \end{matrix}$$

where,

$\boldsymbol{\eta}_1$ is the position vector of the vehicle-fixed frame {*V*} relative to the inertial frame {*I*} and $\boldsymbol{\eta}_2$ is the vector of Euler-angles of the vehicle-fixed frame {*V*} relative to the inertial frame {*I*}.

The vectors $\dot{\boldsymbol{\eta}}_1$ and $\dot{\boldsymbol{\eta}}_2$ are the corresponding time derivatives expressed in the inertial frame {*I*} and $\dot{\boldsymbol{\eta}} = \begin{bmatrix} \dot{\boldsymbol{\eta}}_1^T & \dot{\boldsymbol{\eta}}_2^T \end{bmatrix}^T \in \mathbb{R}^6$.

Let also define the vector of the vehicle-fixed velocities $\boldsymbol{v}$ as:

$$\boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_1^T & \boldsymbol{v}_2^T \end{bmatrix}^T \in \mathbb{R}^6 \quad \begin{matrix} \nearrow \; \boldsymbol{v}_1 = \begin{bmatrix} u & v & w \end{bmatrix}^T \in \mathbb{R}^3 \\ \searrow \; \boldsymbol{v}_2 = \begin{bmatrix} p & q & r \end{bmatrix}^T \in \mathbb{R}^3 \end{matrix}$$

where,

$\boldsymbol{v}_1$ is the linear velocity of the vehicle-fixed frame {*V*} with respect to the inertial frame {*I*} expressed in the vehicle-fixed frame and

$\boldsymbol{v}_2$ is the angular velocity of the vehicle-fixed frame {*V*} with respect to the inertial frame {*I*} expressed in the vehicle-fixed frame.

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:

$$\begin{bmatrix} \dot{\boldsymbol{\eta}}_1 \\ \dot{\boldsymbol{\eta}}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) & \mathbf{0} \\ \mathbf{0} & \boldsymbol{J}_{v2}(\boldsymbol{\eta}_2) \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix} \quad \Leftrightarrow \quad \dot{\boldsymbol{\eta}} = \boldsymbol{J}_v(\boldsymbol{\eta}_2) \cdot \boldsymbol{v} \tag{2.1}$$

where,

$$\boldsymbol{J}_{v1}(\boldsymbol{\eta}_2) = {}^I\boldsymbol{R}_V(\boldsymbol{\eta}_2) = \begin{bmatrix} c\psi c\theta & -s\psi c\varphi + c\psi s\theta s\varphi & s\psi s\varphi + c\psi s\theta c\varphi \\ s\psi c\theta & c\psi c\varphi + s\psi s\theta s\varphi & -c\psi s\varphi + s\psi s\theta c\varphi \\ -s\theta & c\theta s\varphi & c\theta c\varphi \end{bmatrix} \tag{2.2}$$

$$\boldsymbol{J}_{v2}(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & s\varphi t\theta & c\varphi t\theta \\ 0 & c\varphi & -s\varphi \\ 0 & s\varphi/c\theta & c\varphi/c\theta \end{bmatrix} \tag{2.3}$$

and ${}^{I}\boldsymbol{R}_{V}\left(\boldsymbol{\eta}_{2}\right)$ is the rotation matrix[1] expressing the transformation from the vehicle-fixed frame {*V*} to the inertial frame {*I*}.

For a *n*-link manipulator, the joint angular position state vector is defined by $\boldsymbol{q}_{m}=\begin{bmatrix} q_{1} & q_{2} & \cdots & q_{n_m} \end{bmatrix}^{T} \in \mathbb{R}^{n_m}$.

Next, the end-effector local pose vector $\boldsymbol{p}$ is defined as:

$$\boldsymbol{p}=\begin{bmatrix} \boldsymbol{p}_{1}^{T} & \boldsymbol{p}_{2}^{T} \end{bmatrix}^{T} \in \mathbb{R}^{6} \quad \begin{matrix} \nearrow \; \boldsymbol{p}_{1}=\begin{bmatrix} x_{l} & y_{l} & z_{l} \end{bmatrix}^{T} \in \mathbb{R}^{3} \\ \searrow \; \boldsymbol{p}_{2}=\begin{bmatrix} \varphi_{l} & \theta_{l} & \psi_{l} \end{bmatrix}^{T} \in \mathbb{R}^{3} \end{matrix}$$

where,

$\boldsymbol{p}_{1}$ is the local position vector of the e-e frame {*E*} expressed in the vehicle-fixed frame {*V*} and

$\boldsymbol{p}_{2}$ is the local orientation (Euler-angles) vector of the e-e frame {*E*} expressed in the vehicle-fixed frame {*V*}.

The vectors $\dot{\boldsymbol{p}}_{1}$ and $\dot{\boldsymbol{p}}_{2}$ are the corresponding time derivatives expressed in the vehicle-fixed frame {*V*} and $\dot{\boldsymbol{p}}=\begin{bmatrix} \dot{\boldsymbol{p}}_{1}^{T} & \dot{\boldsymbol{p}}_{2}^{T} \end{bmatrix}^{T} \in \mathbb{R}^{6}$.

Accordingly, the end-effector pose vector $\boldsymbol{p}_{e}$ is defined as:

$$\boldsymbol{p}_{e}=\begin{bmatrix} \boldsymbol{p}_{e1}^{T} & \boldsymbol{p}_{e2}^{T} \end{bmatrix}^{T} \in \mathbb{R}^{6} \quad \begin{matrix} \nearrow \; \boldsymbol{p}_{e1}=\begin{bmatrix} x_{e} & y_{e} & z_{e} \end{bmatrix}^{T} \in \mathbb{R}^{3} \\ \searrow \; \boldsymbol{p}_{e2}=\begin{bmatrix} \varphi_{e} & \theta_{e} & \psi_{e} \end{bmatrix}^{T} \in \mathbb{R}^{3} \end{matrix}$$

where,

$\boldsymbol{p}_{e1}$ is the position vector of the e-e frame {*E*} expressed in the inertial frame {*I*} and

$\boldsymbol{p}_{e2}$ is the orientation (Euler-angles) vector of the e-e frame {*E*} expressed in the inertial frame {*I*}.

[1] In general, ${}^{j}\boldsymbol{R}_{i} \in SO(3)$ is the rotation matrix describing the orientation of frame {i} wrt the frame {j}.

The vectors $\dot{\boldsymbol{p}}_{e1}$ and $\dot{\boldsymbol{p}}_{e2}$ are the corresponding time derivatives expressed in the inertial frame {*I*} and $\dot{\boldsymbol{p}}_e = \left[ \dot{\boldsymbol{p}}_{e1}^T \quad \dot{\boldsymbol{p}}_{e2}^T \right]^T \in \mathbb{R}^6$.

Next, we proceed to define the relation between the e-e local velocity vector $\dot{\boldsymbol{p}}$ and the joint velocity vector $\dot{\boldsymbol{q}}_m$.

Assuming that matrix $\boldsymbol{J}_m \in \mathbb{R}^{6\times n_m}$ represents the manipulator geometric Jacobian matrix with respect to base frame, we get:

$$\begin{bmatrix} {}^0\boldsymbol{v}_E \\ {}^0\boldsymbol{\omega}_E \end{bmatrix} = \boldsymbol{J}_m \cdot \dot{\boldsymbol{q}}_m = \begin{bmatrix} \boldsymbol{J}_{m1} \\ \boldsymbol{J}_{m2} \end{bmatrix} \cdot \dot{\boldsymbol{q}}_m \tag{2.4}$$

where matrices $\boldsymbol{J}_{m1} \in \mathbb{R}^{3\times n_m}$ and $\boldsymbol{J}_{m2} \in \mathbb{R}^{3\times n_m}$ 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 {*V*}. The relationship between velocities in the two frames is:

$${}^V\mathbf{V}_E = \begin{bmatrix} {}^V\boldsymbol{R}_0 & \mathbf{0} \\ \mathbf{0} & {}^V\boldsymbol{R}_0 \end{bmatrix} \cdot {}^0\mathbf{V}_E \quad \Rightarrow$$

$$\begin{bmatrix} \dot{\boldsymbol{p}}_1 \\ {}^V\boldsymbol{\omega}_E \end{bmatrix} = \begin{bmatrix} {}^V\boldsymbol{R}_0 & \mathbf{0} \\ \mathbf{0} & {}^V\boldsymbol{R}_0 \end{bmatrix} \cdot \begin{bmatrix} {}^0\boldsymbol{v}_E \\ {}^0\boldsymbol{\omega}_E \end{bmatrix} \tag{2.5}$$

where the rotation matrix ${}^V\boldsymbol{R}_0$ denotes the transformation from the manipulator base frame {0} to the vehicle-fixed frame {*V*}.

Substituting Eq. 2.4 into Eq. 2.5, the end-effector linear and angular velocity vector expressed in the vehicle-fixed frame {*V*} is given by:

$$\begin{bmatrix} \dot{\boldsymbol{p}}_1 \\ {}^V\boldsymbol{\omega}_E \end{bmatrix} = \begin{bmatrix} {}^V\boldsymbol{R}_0 & \mathbf{0} \\ \mathbf{0} & {}^V\boldsymbol{R}_0 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{J}_{m1} \\ \boldsymbol{J}_{m2} \end{bmatrix} \cdot \dot{\boldsymbol{q}}_m \tag{2.6}$$

Furthermore, the following equation, relating the angular velocity ${}^{V}\boldsymbol{\omega}_{E}$ to the time derivative of the Euler angles $\dot{\boldsymbol{p}}_2$, holds:

$$ {}^{V}\boldsymbol{\omega}_{E} = \boldsymbol{E}_{r}\left(\boldsymbol{p}_2\right)\cdot\dot{\boldsymbol{p}}_2 \tag{2.7} $$

where,

$$ \boldsymbol{E}_{r}\left(\boldsymbol{p}_2\right) = {}^{V}\boldsymbol{R}_{E}\left(\boldsymbol{p}_2\right)\cdot\boldsymbol{J}_{v2}^{-1}\left(\boldsymbol{p}_2\right) \tag{2.8} $$

rotation matrix ${}^{V}\boldsymbol{R}_{E}$ denotes the transformation from the e-e frame {*E*} to the vehicle-fixed frame {*V*} and $\boldsymbol{J}_{v2}\left(\boldsymbol{p}_2\right)$ is defined as in Eq. 2.3.

Inverting Eq. 2.7 and considering Eq. 2.8 gives that:

$$ \dot{\boldsymbol{p}}_2 = \boldsymbol{J}_{v2}\left(\boldsymbol{p}_2\right)\cdot{}^{E}\boldsymbol{R}_{V}\cdot{}^{V}\boldsymbol{\omega}_{E} \tag{2.9} $$

Combining Eq. 2.6 with Eq. 2.9, 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:

$$ \dot{\boldsymbol{p}} = \begin{bmatrix}\dot{\boldsymbol{p}}_1\\ \dot{\boldsymbol{p}}_2\end{bmatrix} = \begin{bmatrix}{}^{V}\boldsymbol{R}_0\cdot\boldsymbol{J}_{m1}\\ \boldsymbol{J}_{v2}\left(\boldsymbol{p}_2\right)\cdot{}^{E}\boldsymbol{R}_{V}\cdot{}^{V}\boldsymbol{R}_0\cdot\boldsymbol{J}_{m2}\end{bmatrix}\cdot\dot{\boldsymbol{q}}_m \tag{2.10} $$

Next, we proceed to define the differential kinematic relationship between the end-effector velocities expressed in the earth-fixed frame {*I*} and the body-fixed system velocity.

The coordinate transformation matrix[2] from the end-effector frame {*E*} to the inertial frame {*I*} is expressed as below:

$$ {}^{I}\boldsymbol{T}_{E} = \begin{bmatrix}{}^{I}\boldsymbol{R}_{V}\cdot{}^{V}\boldsymbol{R}_{E} & \boldsymbol{\eta}_1 + {}^{I}\boldsymbol{R}_{V}\cdot\boldsymbol{p}_1\\ \boldsymbol{0}_{1\times3} & 1\end{bmatrix} \tag{2.11} $$

[2] In general, ${}^{J}\boldsymbol{T}_{i}\in SE\left(3\right)$ is the homogeneous transformation matrix describing the position and orientation of frame {i} wrt the frame {j}.

From the coordinate transformation Eq. 2.11, the position vector of the end-effector in the inertial frame {*I*} is written as:

$$\boldsymbol{p}_{e1} = \boldsymbol{\eta}_1 + {}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{p}_1 \tag{2.12}$$

Differentiating Eq. 2.12 with respect to time yields:

$$\dot{\boldsymbol{p}}_{e1} = \dot{\boldsymbol{\eta}}_1 + {}^{I}\dot{\boldsymbol{R}}_{V} \cdot \boldsymbol{p}_1 + {}^{I}\boldsymbol{R}_{V} \cdot \dot{\boldsymbol{p}}_1 \tag{2.13}$$

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

$${}^{I}\dot{\boldsymbol{R}}_{V} = {}^{I}\boldsymbol{\omega}_{V} \times {}^{I}\boldsymbol{R}_{V} \tag{2.14}$$

we get the following relation:

$$\dot{\boldsymbol{p}}_{e1} = \dot{\boldsymbol{\eta}}_1 + \left({}^{I}\boldsymbol{\omega}_{V}\right) \times {}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{p}_1 + {}^{I}\boldsymbol{R}_{V} \cdot \dot{\boldsymbol{p}}_1 \quad \Rightarrow$$

$$\dot{\boldsymbol{p}}_{e1} = \dot{\boldsymbol{\eta}}_1 - \left({}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{p}_1\right) \times {}^{I}\boldsymbol{\omega}_{V} + {}^{I}\boldsymbol{R}_{V} \cdot \dot{\boldsymbol{p}}_1 \tag{2.15}$$

Considering that

$${}^{I}\boldsymbol{\omega}_{V} = {}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{v}_2 \tag{2.16}$$

and substituting Eq. 2.1 and Eq. 2.10 into Eq. 2.15, we get:

$$\dot{\boldsymbol{p}}_{e1} = \boldsymbol{J}_{v1}\left(\boldsymbol{\eta}_2\right) \cdot \boldsymbol{v}_1 - \left({}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{p}_1\right) \times {}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{v}_2 + {}^{I}\boldsymbol{R}_{V} \cdot {}^{V}\boldsymbol{R}_{0} \cdot \boldsymbol{J}_{m1} \cdot \dot{\boldsymbol{q}}_m \quad \Rightarrow$$

$$\dot{\boldsymbol{p}}_{e1} = \boldsymbol{J}_{v1}\left(\boldsymbol{\eta}_2\right) \cdot \boldsymbol{v}_1 - S\left({}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{p}_1\right) \cdot {}^{I}\boldsymbol{R}_{V} \cdot \boldsymbol{v}_2 + {}^{I}\boldsymbol{R}_{0} \cdot \boldsymbol{J}_{m1} \cdot \dot{\boldsymbol{q}}_m \tag{2.17}$$

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

Considering the following property:

$$S\left(\boldsymbol{R} \cdot \boldsymbol{a}\right) = \boldsymbol{R} \cdot S\left(\boldsymbol{a}\right) \cdot \boldsymbol{R}^{T} \quad \text{for any } \boldsymbol{R} \in SO\left(3\right),\ \boldsymbol{a} \in \mathbb{R}^3 \tag{2.18}$$

then Eq. 2.17 is restated as follows:

$$\dot{\boldsymbol{p}}_{e1} = \boldsymbol{J}_{v1}(\boldsymbol{\eta}_2)\cdot\boldsymbol{v}_1 - {}^{I}\boldsymbol{R}_{V}\cdot S(\boldsymbol{p}_1)\cdot\boldsymbol{v}_2 \; + {}^{I}\boldsymbol{R}_{0}\cdot\boldsymbol{J}_{m1}\cdot\dot{\boldsymbol{q}}_m \tag{2.19}$$

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

$${}^{I}\boldsymbol{\omega}_{E} = {}^{I}\boldsymbol{\omega}_{V} + {}^{I}\boldsymbol{R}_{V}\cdot{}^{V}\boldsymbol{\omega}_{E} \tag{2.20}$$

Substituting Eq. 2.6 and Eq. 2.16 into Eq. 2.20, we get:

$${}^{I}\boldsymbol{\omega}_{E} = {}^{I}\boldsymbol{R}_{V}\cdot\boldsymbol{v}_2 + {}^{I}\boldsymbol{R}_{V}\cdot{}^{V}\boldsymbol{R}_{0}\cdot\boldsymbol{J}_{m2}\cdot\dot{\boldsymbol{q}}_m \tag{2.21}$$

Moreover, as with Eq. 2.7, it holds:

$${}^{I}\boldsymbol{\omega}_{E} = \boldsymbol{E}_{r}(\boldsymbol{p}_{e2})\cdot\dot{\boldsymbol{p}}_{e2} \quad \Rightarrow$$

$${}^{I}\boldsymbol{\omega}_{E} = {}^{I}\boldsymbol{R}_{E}(\boldsymbol{p}_{e2})\cdot\boldsymbol{J}_{v2}^{-1}(\boldsymbol{p}_{e2})\cdot\dot{\boldsymbol{p}}_{e2} \tag{2.22}$$

Combining Eq. 2.21 with Eq. 2.22 and solving for $\dot{\boldsymbol{p}}_{e2}$ we derive:

$${}^{I}\boldsymbol{R}_{E}(\boldsymbol{p}_{e2})\cdot\boldsymbol{J}_{v2}^{-1}(\boldsymbol{p}_{e2})\cdot\dot{\boldsymbol{p}}_{e2} = {}^{I}\boldsymbol{R}_{V}\cdot\boldsymbol{v}_2 + {}^{I}\boldsymbol{R}_{V}\cdot{}^{V}\boldsymbol{R}_{0}\cdot\boldsymbol{J}_{m2}\cdot\dot{\boldsymbol{q}}_m \quad \Rightarrow$$

$$\dot{\boldsymbol{p}}_{e2} = \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2})\cdot{}^{E}\boldsymbol{R}_{I}\cdot{}^{I}\boldsymbol{R}_{V}\cdot\boldsymbol{v}_2 + \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2})\cdot{}^{E}\boldsymbol{R}_{I}\cdot{}^{I}\boldsymbol{R}_{V}\cdot{}^{V}\boldsymbol{R}_{0}\cdot\boldsymbol{J}_{m2}\cdot\dot{\boldsymbol{q}}_m \quad \Rightarrow$$

$$\dot{\boldsymbol{p}}_{e2} = \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2})\cdot{}^{E}\boldsymbol{R}_{V}\cdot\boldsymbol{v}_2 + \boldsymbol{J}_{v2}(\boldsymbol{p}_{e2})\cdot{}^{E}\boldsymbol{R}_{0}\cdot\boldsymbol{J}_{m2}\cdot\dot{\boldsymbol{q}}_m \tag{2.23}$$

Now, let define the body-fixed system velocity vector $\boldsymbol{\zeta} = \begin{bmatrix}\boldsymbol{v}^T & \dot{\boldsymbol{q}}_m{}^T\end{bmatrix}^T \in \mathbb{R}^{n_m+6}$.

Finally, considering Eq. 2.19 and Eq. 2.23, it turns out that the e-e velocities $\dot{\boldsymbol{p}}_e = \begin{bmatrix}\dot{\boldsymbol{p}}_{e1}^T & \dot{\boldsymbol{p}}_{e2}^T\end{bmatrix}^T$ expressed in the inertial frame {*I*}, are related to the body-fixed system velocity vector $\boldsymbol{\zeta} = \begin{bmatrix}\boldsymbol{v}^T & \dot{\boldsymbol{q}}_m{}^T\end{bmatrix}^T$ by the following Jacobian matrix:

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

$$\dot{\boldsymbol{p}}_e = \boldsymbol{J}_W(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} \tag{2.24}$$

Eq. 2.24 is also known as the UVMS differential kinematics equation.

## 2.3 Dynamics of UVMS

The analysis of this section aims only to present and briefly explain the dynamic equations of motion of an UVMS. Their derivation and mathematical expression will not be studied in detail.

The equations of motion of the UVMS in a body-fixed frame {*V*} can be conveniently written in a matrix form as:

$$\boldsymbol{M}(\boldsymbol{q}_m) \cdot \dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) = \boldsymbol{\tau} \tag{2.25}$$

where,

$\boldsymbol{M}(\boldsymbol{q}_m) \in \mathbb{R}^{(n_m+6)\times(n_m+6)}$ is the inertia matrix including added mass,

$\boldsymbol{C}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} \in \mathbb{R}^{n_m+6}$ is the vector of Coriolis and centripetal terms,

$\boldsymbol{D}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} \in \mathbb{R}^{n_m+6}$ is the vector of dissipative effects,

$\boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \in \mathbb{R}^{n_m+6}$ is the vector of gravity and buoyancy effects,

$\boldsymbol{\tau} \in \mathbb{R}^{n_m+6}$ is the vector of force/moment acting on the vehicle as well as joint torques and $\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{v}^T & \dot{\boldsymbol{q}}_m{}^T \end{bmatrix}^T \in \mathbb{R}^{n_m+6}$ is the body-fixed system velocity vector.

For advanced and systematic decentralized control algorithms, the partitioned submatrices of dynamic terms are absolutely needed to design the controller. Using this partitioned dynamics, we can easily deal with the coupling effect between two subsystems. So, at the following, the dynamic decomposition will be treated. Firstly, the vehicle and manipulator dynamic effects are simply obtained from many references and calculations. Those are expressed as follows:

$$\begin{bmatrix} \boldsymbol{M}_v & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \cdot \dot{\boldsymbol{\zeta}} + \begin{bmatrix} \boldsymbol{C}_v(\boldsymbol{v}) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \cdot \boldsymbol{\zeta} + \begin{bmatrix} \boldsymbol{D}_v(\boldsymbol{v}) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \cdot \boldsymbol{\zeta} + \begin{bmatrix} \boldsymbol{g}_v(\boldsymbol{\eta}_2) \\ \boldsymbol{0} \end{bmatrix} = \boldsymbol{\tau}_1 \tag{2.26}$$

$$\begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}_m(\boldsymbol{q}_m) \end{bmatrix} \cdot \dot{\boldsymbol{\zeta}} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C}_m(\boldsymbol{q}_m, \dot{\boldsymbol{q}}_m) \end{bmatrix} \cdot \boldsymbol{\zeta} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D}_m(\boldsymbol{q}_m, \dot{\boldsymbol{q}}_m) \end{bmatrix} \cdot \boldsymbol{\zeta} + \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{g}_m(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \end{bmatrix} = \boldsymbol{\tau}_2 \tag{2.27}$$

Then, the effect of the added dynamics to the vehicle by manipulator has to be considered. This dynamic effect can be introduced as follows:

$$\begin{bmatrix} \boldsymbol{M}_s(\boldsymbol{q}_m) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \cdot \dot{\boldsymbol{\zeta}} + \begin{bmatrix} \boldsymbol{C}_s(\boldsymbol{v}, \boldsymbol{q}_m) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \cdot \boldsymbol{\zeta} + \begin{bmatrix} \boldsymbol{D}_s(\boldsymbol{v}, \boldsymbol{q}_m) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \cdot \boldsymbol{\zeta} + \begin{bmatrix} \boldsymbol{g}_s(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \\ \boldsymbol{0} \end{bmatrix} = \boldsymbol{\tau}_3 \tag{2.28}$$

In addition, the interaction dynamics between two subsystems are expressed in the form:

$$\begin{bmatrix} \boldsymbol{0} & \boldsymbol{M}_{12}(\boldsymbol{q}_m) \\ \boldsymbol{M}_{12}^T(\boldsymbol{q}_m) & \boldsymbol{0} \end{bmatrix} \cdot \dot{\boldsymbol{\zeta}} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{C}_{12}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \\ \boldsymbol{C}_{21}(\boldsymbol{\zeta}, \boldsymbol{q}_m) & \boldsymbol{0} \end{bmatrix} \cdot \boldsymbol{\zeta} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{D}_{12}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \\ \boldsymbol{D}_{21}(\boldsymbol{\zeta}, \boldsymbol{q}_m) & \boldsymbol{0} \end{bmatrix} \cdot \boldsymbol{\zeta} = \boldsymbol{\tau}_4 \tag{2.29}$$

So, each of the dynamic and static terms in Eq. 2.25 can be written as:

$$\boldsymbol{M}(\boldsymbol{q}_m) = \begin{bmatrix} \boldsymbol{M}_v + \boldsymbol{M}_s(\boldsymbol{q}_m) & \boldsymbol{M}_{12}(\boldsymbol{q}_m) \\ \boldsymbol{M}_{12}^T(\boldsymbol{q}_m) & \boldsymbol{M}_m(\boldsymbol{q}_m) \end{bmatrix} \tag{2.30}$$

$$\boldsymbol{C}(\boldsymbol{\zeta}, \boldsymbol{q}_m) = \begin{bmatrix} \boldsymbol{C}_v(\boldsymbol{v}) + \boldsymbol{C}_s(\boldsymbol{v}, \boldsymbol{q}_m) & \boldsymbol{C}_{12}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \\ \boldsymbol{C}_{21}(\boldsymbol{\zeta}, \boldsymbol{q}_m) & \boldsymbol{C}_m(\boldsymbol{q}_m, \dot{\boldsymbol{q}}_m) \end{bmatrix} \tag{2.31}$$

$$\boldsymbol{D}(\boldsymbol{\zeta}, \boldsymbol{q}_m) = \begin{bmatrix} \boldsymbol{D}_v(\boldsymbol{v}) + \boldsymbol{D}_s(\boldsymbol{v}, \boldsymbol{q}_m) & \boldsymbol{D}_{12}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \\ \boldsymbol{D}_{21}(\boldsymbol{\zeta}, \boldsymbol{q}_m) & \boldsymbol{D}_m(\boldsymbol{q}_m, \dot{\boldsymbol{q}}_m) \end{bmatrix} \tag{2.32}$$

$$\boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) = \begin{bmatrix} \boldsymbol{g}_v(\boldsymbol{\eta}_2) + \boldsymbol{g}_s(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \\ \boldsymbol{g}_m(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \end{bmatrix} \tag{2.33}$$

$$\boldsymbol{\tau} = \boldsymbol{\tau}_1 + \boldsymbol{\tau}_2 + \boldsymbol{\tau}_3 + \boldsymbol{\tau}_4 \tag{2.34}$$

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 [3]:

$$\boldsymbol{M}(\boldsymbol{q}_m) \cdot \dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{\zeta}, \boldsymbol{q}_m) \cdot \boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) + \boldsymbol{J}_W^T(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{h}_e = \boldsymbol{\tau} \tag{2.35}$$

where $\boldsymbol{J}_W \in \mathbb{R}^{6\times(n_m+6)}$ is the Jacobian matrix defined in Eq. 2.24 and $\boldsymbol{h}_e$ is the vector of force/moment at the end effector expressed in the inertial frame {*I*} and defined as follows:

$$\boldsymbol{h}_e = \begin{bmatrix} \boldsymbol{f}_e^T & \boldsymbol{\mu}_e^T \end{bmatrix}^T \in \mathbb{R}^6 \quad \begin{matrix} \nearrow \ \boldsymbol{f}_e \in \mathbb{R}^3 \\ \searrow \ \boldsymbol{\mu}_e \in \mathbb{R}^3 \end{matrix}$$

## 2.4 Restoring Forces

In this section the vector of gravity and buoyancy generalized forces, also known as restoring vector, of an UVMS is presented. The derivation and mathematical expression of it will be studied in detail. All the position vectors used in the following analysis for the derivation of the restoring vector are depicted in Figure 2.2.

According to Eq. 2.33, it holds:

$$\boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) = \begin{bmatrix} \boldsymbol{g}_v(\boldsymbol{\eta}_2) + \boldsymbol{g}_s(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \\ \boldsymbol{g}_m(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \end{bmatrix}$$

where ,

$\boldsymbol{g}_v(\boldsymbol{\eta}_2)$ is the restoring vector of the AUV,

$\boldsymbol{g}_s(\boldsymbol{\eta}_2, \boldsymbol{q}_m)$ is the restoring vector of the AUV due to the manipulator and

$\boldsymbol{g}_m(\boldsymbol{\eta}_2, \boldsymbol{q}_m)$ is the restoring vector of the manipulator.

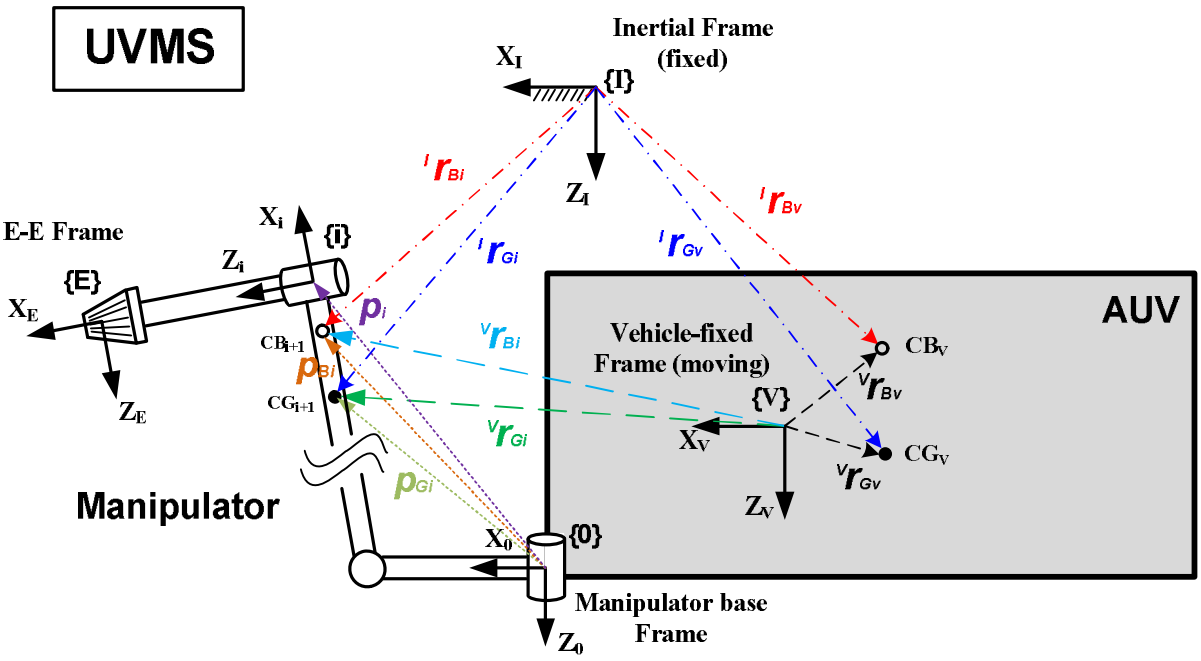



**Figure 2.2:** Schematic diagram of UVMS with position vectors attached.

**Restoring vector of AUV including the restoring vector due to manipulator**

At first, we proceed to express analytically the restoring vector of the AUV including the restoring vector due to the manipulator, i.e. $\boldsymbol{g}_v(\boldsymbol{\eta}_2)+\boldsymbol{g}_s(\boldsymbol{\eta}_2,\boldsymbol{q}_m)$.

GRAVITY

- The gravity forces expressed in the inertial frame $\{I\}$ are:

Vehicle:

$$^I\boldsymbol{f}_{Gv} = m_v \cdot \boldsymbol{g}_I = {}^I\begin{bmatrix} 0 \\ 0 \\ m_v \cdot g \end{bmatrix} = {}^I\begin{bmatrix} 0 \\ 0 \\ W_v \end{bmatrix} \tag{2.36}$$

Link *i*:

$$^I\boldsymbol{f}_{Gi} = m_{li} \cdot \boldsymbol{g}_I = {}^I\begin{bmatrix} 0 \\ 0 \\ m_{li} \cdot g \end{bmatrix} = {}^I\begin{bmatrix} 0 \\ 0 \\ W_{li} \end{bmatrix} \tag{2.37}$$

where,

$m_v$ is the mass of vehicle,

$m_{li}$ is the mass of link *i* and

$\boldsymbol{g}_I = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$ is the gravitational acceleration vector expressed in the inertial frame {*I*} .

- The gravity torques expressed in the inertial frame {*I*} are:

Vehicle:

$$^I\boldsymbol{\tau}_{Gv} = {}^I\boldsymbol{r}_{Gv} \times {}^I\boldsymbol{f}_{Gv} \tag{2.38}$$

Link *i*:

$$^I\boldsymbol{\tau}_{Gi} = {}^I\boldsymbol{r}_{Gi} \times {}^I\boldsymbol{f}_{Gi} \tag{2.39}$$

where,

$^I\boldsymbol{r}_{Gv}$ is the position vector of the center of gravity of vehicle ($CG_V$) with respect to the inertial frame {*I*} and

$^I\boldsymbol{r}_{Gi}$ is the position vector of the center of gravity of link *i* ($CG_i$) with respect to the inertial frame {*I*}.

- Both gravity forces and torques are expressed in the vehicle-fixed frame {*V*} as:

Vehicle:

$$^V\boldsymbol{f}_{Gv} = {}^V\boldsymbol{R}_I \cdot {}^I\boldsymbol{f}_{Gv} \tag{2.40}$$

$$^V\boldsymbol{\tau}_{Gv} = {}^V\boldsymbol{r}_{Gv} \times {}^V\boldsymbol{f}_{Gv} \tag{2.41}$$

Link *i*:

$$^V\boldsymbol{f}_{Gi} = {}^V\boldsymbol{R}_I \cdot {}^I\boldsymbol{f}_{Gi} \tag{2.42}$$

$$^V\boldsymbol{\tau}_{Gi} = {}^V\boldsymbol{r}_{Gi} \times {}^V\boldsymbol{f}_{Gi} \tag{2.43}$$

where,

$^V\boldsymbol{R}_I$ is the rotation matrix from the inertial frame {*I*} to vehicle-fixed frame {*V*},

${}^{V}\boldsymbol{r}_{Gv} = {}^{V}\begin{bmatrix} x_{Gv} \\ y_{Gv} \\ z_{Gv} \end{bmatrix}$ is the position vector of the center of gravity of vehicle with respect to {*V*}

and

${}^{V}\boldsymbol{r}_{Gi} = {}^{V}\begin{bmatrix} x_{Gi} \\ y_{Gi} \\ z_{Gi} \end{bmatrix}$ is the position vector of the center of gravity of link *i* with respect to {*V*}.

So it turns out,

Vehicle:

$$ {}^{V}\boldsymbol{f}_{Gv} = {}^{V}\boldsymbol{R}_{I} \cdot {}^{I}\begin{bmatrix} 0 \\ 0 \\ W_v \end{bmatrix} = \begin{bmatrix} c\psi c\theta & s\psi c\theta & -s\theta \\ -s\psi c\varphi + c\psi s\theta s\varphi & c\psi c\varphi + s\psi s\theta s\varphi & c\theta s\varphi \\ s\psi s\varphi + c\psi s\theta c\varphi & -c\psi s\varphi + s\psi s\theta c\varphi & c\theta c\varphi \end{bmatrix} \cdot {}^{I}\begin{bmatrix} 0 \\ 0 \\ W_v \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{f}_{Gv} = {}^{V}\begin{bmatrix} -s\theta \cdot W_v \\ c\theta s\varphi \cdot W_v \\ c\theta c\varphi \cdot W_v \end{bmatrix} \tag{2.44} $$

$$ {}^{V}\boldsymbol{\tau}_{Gv} = {}^{V}\begin{bmatrix} x_{Gv} \\ y_{Gv} \\ z_{Gv} \end{bmatrix} \times {}^{V}\begin{bmatrix} -s\theta \cdot W_v \\ c\theta s\varphi \cdot W_v \\ c\theta c\varphi \cdot W_v \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{\tau}_{Gv} = \begin{bmatrix} y_{Gv} \cdot c\theta c\varphi \cdot W_v - z_{Gv} \cdot c\theta s\varphi \cdot W_v \\ -z_{Gv} \cdot s\theta \cdot W_v - x_{Gv} \cdot c\theta c\varphi \cdot W_v \\ x_{Gv} \cdot c\theta s\varphi \cdot W_v + y_{Gv} \cdot s\theta \cdot W_v \end{bmatrix} \tag{2.45} $$

Link *i*:

$$ {}^{V}\boldsymbol{f}_{Gi} = {}^{V}\boldsymbol{R}_{I} \cdot {}^{I}\begin{bmatrix} 0 \\ 0 \\ W_{li} \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{f}_{Gi} = {}^{V}\begin{bmatrix} -s\theta \cdot W_{li} \\ c\theta s\varphi \cdot W_{li} \\ c\theta c\varphi \cdot W_{li} \end{bmatrix} \tag{2.46} $$

$$ {}^{V}\boldsymbol{\tau}_{Gi} = {}^{V}\begin{bmatrix} x_{Gi} \\ y_{Gi} \\ z_{Gi} \end{bmatrix} \times {}^{V}\begin{bmatrix} -s\theta \cdot W_{li} \\ c\theta s\varphi \cdot W_{li} \\ c\theta c\varphi \cdot W_{li} \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{\tau}_{Gi} = \begin{bmatrix} y_{Gi} \cdot c\theta c\varphi \cdot W_{li} - z_{Gi} \cdot c\theta s\varphi \cdot W_{li} \\ -z_{Gi} \cdot s\theta \cdot W_{li} - x_{Gi} \cdot c\theta c\varphi \cdot W_{li} \\ x_{Gi} \cdot c\theta s\varphi \cdot W_{li} + y_{Gi} \cdot s\theta \cdot W_{li} \end{bmatrix} \tag{2.47} $$

BUOYANCY

- The buoyancy forces expressed in the earth-fixed frame $\{I\}$ are :

Vehicle:

$$ {}^{I}\boldsymbol{f}_{Bv} = -\rho \cdot V_v \cdot \boldsymbol{g}_I = {}^{I}\begin{bmatrix} 0 \\ 0 \\ -\rho \cdot V_v \cdot g \end{bmatrix} = {}^{I}\begin{bmatrix} 0 \\ 0 \\ -B_v \end{bmatrix} \tag{2.48} $$

Link *i*:

$$ {}^{I}\boldsymbol{f}_{Bi} = -\rho \cdot V_{li} \cdot \boldsymbol{g}_I = {}^{I}\begin{bmatrix} 0 \\ 0 \\ -\rho \cdot V_{li} \cdot g \end{bmatrix} = {}^{I}\begin{bmatrix} 0 \\ 0 \\ -B_{li} \end{bmatrix} \tag{2.49} $$

where,

$V_v$ is the volume of vehicle,

$V_{li}$ is the volume of link *i* and

$\rho$ is the fluid density.

- The buoyancy torques expressed in the earth-fixed frame {*I*} are :

Vehicle:

$$^{I}\boldsymbol{\tau}_{Bv} = {}^{I}\boldsymbol{r}_{Bv} \times {}^{I}\boldsymbol{f}_{Bv} \tag{2.50}$$

Link *i*:

$$^{I}\boldsymbol{\tau}_{Bi} = {}^{I}\boldsymbol{r}_{Bi} \times {}^{I}\boldsymbol{f}_{Bi} \tag{2.51}$$

where,

$^{I}\boldsymbol{r}_{Bv}$ is the position vector of the center of buoyancy of vehicle ($CB_V$) with respect to {*I*} and

$^{I}\boldsymbol{r}_{Bi}$ is the position vector of the center of buoyancy of link *i* ($CB_i$) with respect to {*I*}.

- Both buoyancy forces and torques are expressed in the vehicle-fixed frame {*V*} as follows:

Vehicle:

$$^{V}\boldsymbol{f}_{Bv} = {}^{V}\boldsymbol{R}_{I} \cdot {}^{I}\boldsymbol{f}_{Bv} \tag{2.52}$$

$$^{V}\boldsymbol{\tau}_{Bv} = {}^{V}\boldsymbol{r}_{Bv} \times {}^{V}\boldsymbol{f}_{Bv} \tag{2.53}$$

Link *i*:

$$^{V}\boldsymbol{f}_{Bi} = {}^{V}\boldsymbol{R}_{I} \cdot {}^{I}\boldsymbol{f}_{Bi} \tag{2.54}$$

$$^{V}\boldsymbol{\tau}_{Bi} = {}^{V}\boldsymbol{r}_{Bi} \times {}^{V}\boldsymbol{f}_{Bi} \tag{2.55}$$

where,

$^{V}\boldsymbol{r}_{Bv} = {}^{V}\begin{bmatrix} x_{Bv} \\ y_{Bv} \\ z_{Bv} \end{bmatrix}$ is the position of the center of buoyancy of vehicle with respect to {*V*} and

$^{V}\boldsymbol{r}_{Bi} = {}^{V}\begin{bmatrix} x_{Bi} \\ y_{Bi} \\ z_{Bi} \end{bmatrix}$ is the position of the center of buoyancy of link *i* with respect to {*V*}.

So it turns out,

Vehicle:

$$ {}^{V}\boldsymbol{f}_{Bv} = {}^{V}\boldsymbol{R}_{I} \cdot {}^{I}\begin{bmatrix} 0 \\ 0 \\ -B_v \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{f}_{Bv} = {}^{V}\begin{bmatrix} s\theta \cdot B_v \\ -c\theta s\varphi \cdot B_v \\ -c\theta c\varphi \cdot B_v \end{bmatrix} \tag{2.56} $$

$$ {}^{V}\boldsymbol{\tau}_{Bv} = {}^{V}\begin{bmatrix} x_{Bv} \\ y_{Bv} \\ z_{Bv} \end{bmatrix} \times {}^{V}\begin{bmatrix} s\theta \cdot B_v \\ -c\theta s\varphi \cdot B_v \\ -c\theta c\varphi \cdot B_v \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{\tau}_{Bv} = \begin{bmatrix} -y_{Bv} \cdot c\theta c\varphi \cdot B_v + z_{Bv} \cdot c\theta s\varphi \cdot B_v \\ z_{Bv} \cdot s\theta \cdot B_v + x_{Bv} \cdot c\theta c\varphi \cdot B_v \\ -x_{Bv} \cdot c\theta s\varphi \cdot B_v - y_{Bv} \cdot s\theta \cdot B_v \end{bmatrix} \tag{2.57} $$

Link *i*:

$$ {}^{V}\boldsymbol{f}_{Bi} = {}^{V}\boldsymbol{R}_{I} \cdot {}^{I}\begin{bmatrix} 0 \\ 0 \\ -B_{li} \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{f}_{Bi} = {}^{V}\begin{bmatrix} s\theta \cdot B_{li} \\ -c\theta s\varphi \cdot B_{li} \\ -c\theta c\varphi \cdot B_{li} \end{bmatrix} \tag{2.58} $$

$$ {}^{V}\boldsymbol{\tau}_{Bi} = {}^{V}\begin{bmatrix} x_{Bi} \\ y_{Bi} \\ z_{Bi} \end{bmatrix} \times {}^{V}\begin{bmatrix} s\theta \cdot B_{li} \\ -c\theta s\varphi \cdot B_{li} \\ -c\theta c\varphi \cdot B_{li} \end{bmatrix} \quad \Rightarrow $$

$$ {}^{V}\boldsymbol{\tau}_{Bi} = \begin{bmatrix} -y_{Bi} \cdot c\theta c\varphi \cdot B_{li} + z_{Bi} \cdot c\theta s\varphi \cdot B_{li} \\ z_{Bi} \cdot s\theta \cdot B_{li} + x_{Bi} \cdot c\theta c\varphi \cdot B_{li} \\ -x_{Bi} \cdot c\theta s\varphi \cdot B_{li} - y_{Bi} \cdot s\theta \cdot B_{li} \end{bmatrix} \tag{2.59} $$

It must be noted that the position vectors of the centers of gravity and buoyancy of link *i* with respect to {*V*} could be expressed as follows:

$$ {}^{V}\boldsymbol{r}_{Gi} = {}^{V}\boldsymbol{r}_{i-1} + \lambda_{Gi}\left({}^{V}\boldsymbol{r}_{i} - {}^{V}\boldsymbol{r}_{i-1}\right) \quad ,i = 1,...,n_m \tag{2.60} $$

$$ {}^{V}\boldsymbol{r}_{Bi} = {}^{V}\boldsymbol{r}_{i-1} + \lambda_{Bi}\left({}^{V}\boldsymbol{r}_{i} - {}^{V}\boldsymbol{r}_{i-1}\right) \quad ,i = 1,...,n_m \tag{2.61} $$

where,

the parameters $0 \le \lambda_{Gi}, \lambda_{Bi} \le 1$ define the positions of the centers of gravity and buoyancy of link *i*, respectively and

${}^{V}\boldsymbol{r}_{i}$ is the position vector of the origin of coordinate frame {i} with respect to vehicle-fixed frame {*V*}, derived from the above coordinate transformation matrix:

$$ {}^{V}\boldsymbol{T}_{i} = \begin{bmatrix} & & & \vdots & \\ & {}^{V}\boldsymbol{R}_{i} & & \vdots & {}^{V}\boldsymbol{r}_{i} \\ .... & ..... & .... & & .... \\ 0 & \boldsymbol{0}_{1\times 3} & 0 & \vdots & 1 \end{bmatrix} \quad ,i = 1,...,n_m \tag{2.62} $$

Finally, the (6x1) restoring vector of the AUV in body-fixed frame {*V*} is represented by:

$$ \boldsymbol{g}_{v}\left(\boldsymbol{\eta}_2\right) = -\begin{bmatrix} {}^{V}\boldsymbol{f}_{Gv} + {}^{V}\boldsymbol{f}_{Bv} \\ {}^{V}\boldsymbol{\tau}_{Gv} + {}^{V}\boldsymbol{\tau}_{Bv} \end{bmatrix} = -\begin{bmatrix} {}^{V}\boldsymbol{f}_{Gv} + {}^{V}\boldsymbol{f}_{Bv} \\ {}^{V}\boldsymbol{r}_{Gv} \times {}^{V}\boldsymbol{f}_{Gv} + {}^{V}\boldsymbol{r}_{Bv} \times {}^{V}\boldsymbol{f}_{Bv} \end{bmatrix} \quad \Rightarrow $$

$$ \boldsymbol{g}_{v}\left(\boldsymbol{\eta}_2\right) = -\begin{bmatrix} -s\theta \cdot W_v + s\theta \cdot B_v \\ +c\theta s\varphi \cdot W_v - c\theta s\varphi \cdot B_v \\ +c\theta c\varphi \cdot W_v - c\theta c\varphi \cdot B_v \\ +y_{Gv} \cdot c\theta c\varphi \cdot W_v - z_{Gv} \cdot c\theta s\varphi \cdot W_v - y_{Bv} \cdot c\theta c\varphi \cdot B_v + z_{Bv} \cdot c\theta s\varphi \cdot B_v \\ -z_{Gv} \cdot s\theta \cdot W_v - x_{Gv} \cdot c\theta c\varphi \cdot W_v + z_{Bv} \cdot s\theta \cdot B_v + x_{Bv} \cdot c\theta c\varphi \cdot B_v \\ +x_{Gv} \cdot c\theta s\varphi \cdot W_v + y_{Gv} \cdot s\theta \cdot W_v - x_{Bv} \cdot c\theta s\varphi \cdot B_v - y_{Bv} \cdot s\theta \cdot B_v \end{bmatrix} \quad \Rightarrow $$

$$\boldsymbol{g}_v(\boldsymbol{\eta}_2) = -\begin{bmatrix} s\theta\left(B_v - W_v\right) \\ -c\theta s\varphi\left(B_v - W_v\right) \\ -c\theta c\varphi\left(B_v - W_v\right) \\ c\theta c\varphi\left(y_{Gv}\cdot W_v - y_{Bv}\cdot B_v\right) - c\theta s\varphi\left(z_{Gv}\cdot W_v - z_{Bv}\cdot B_v\right) \\ -s\theta\left(z_{Gv}\cdot W_v - z_{Bv}\cdot B_v\right) - c\theta c\varphi\left(x_{Gv}\cdot W_v - x_{Bv}\cdot B_v\right) \\ c\theta s\varphi\left(x_{Gv}\cdot W_v - x_{Bv}\cdot B_v\right) + s\theta\left(y_{Gv}\cdot W_v - y_{Bv}\cdot B_v\right) \end{bmatrix} \tag{2.63}$$

It is worth mentioning that the sign of the restoring forces and moments is changed since this term is included in the left-hand side of Newton's 2nd law as formed in Eq. 2.25.

Then, as far as the restoring vector of the AUV due to the manipulator is concerned, we have:

$$\boldsymbol{g}_s(\boldsymbol{\eta}_2, \boldsymbol{q}_m) = -\begin{bmatrix} \sum_{i=1}^{n_m}\left({}^V\boldsymbol{f}_{Gi} + {}^V\boldsymbol{f}_{Bi}\right) \\ \sum_{i=1}^{n_m}\left({}^V\boldsymbol{\tau}_{Gi} + {}^V\boldsymbol{\tau}_{Bi}\right) \end{bmatrix} = -\begin{bmatrix} \sum_{i=1}^{n_m}\left({}^V\boldsymbol{f}_{Gi} + {}^V\boldsymbol{f}_{Bi}\right) \\ \sum_{i=1}^{n_m}\left({}^V\boldsymbol{r}_{Gi}\times{}^V\boldsymbol{f}_{Gi} + {}^V\boldsymbol{r}_{Bi}\times{}^V\boldsymbol{f}_{Bi}\right) \end{bmatrix} \Rightarrow$$

$$\boldsymbol{g}_s(\boldsymbol{\eta}_2, \boldsymbol{q}_m) = -\begin{bmatrix} \sum_{i=1}^{n_m}\left(-s\theta\cdot W_{li} + s\theta\cdot B_{li}\right) \\ \sum_{i=1}^{n_m}\left(c\theta s\varphi\cdot W_{li} - c\theta s\varphi\cdot B_{li}\right) \\ \sum_{i=1}^{n_m}\left(c\theta c\varphi\cdot W_{li} - c\theta c\varphi\cdot B_{li}\right) \\ \sum_{i=1}^{n_m}\left(y_{Gi}\cdot c\theta c\varphi\cdot W_{li} - z_{Gi}\cdot c\theta s\varphi\cdot W_{li} - y_{Bi}\cdot c\theta c\varphi\cdot B_{li} + z_{Bi}\cdot c\theta s\varphi\cdot B_{li}\right) \\ \sum_{i=1}^{n_m}\left(-z_{Gi}\cdot s\theta\cdot W_{li} - x_{Gi}\cdot c\theta c\varphi\cdot W_{li} + z_{Bi}\cdot s\theta\cdot B_{li} + x_{Bi}\cdot c\theta c\varphi\cdot B_{li}\right) \\ \sum_{i=1}^{n_m}\left(x_{Gi}\cdot c\theta s\varphi\cdot W_{li} + y_{Gi}\cdot s\theta\cdot W_{li} - x_{Bi}\cdot c\theta s\varphi\cdot B_{li} - y_{Bi}\cdot s\theta\cdot B_{li}\right) \end{bmatrix} \Rightarrow$$

$$\boldsymbol{g}_s\left(\boldsymbol{\eta}_2,\boldsymbol{q}_m\right)=-\begin{bmatrix}\sum_{i=1}^{n_m}\left(s\theta\left(B_{li}-W_{li}\right)\right)\\ \sum_{i=1}^{n_m}\left(-c\theta s\varphi\left(B_{li}-W_{li}\right)\right)\\ \sum_{i=1}^{n_m}\left(-c\theta c\varphi\left(B_{li}-W_{li}\right)\right)\\ \sum_{i=1}^{n_m}\left(c\theta c\varphi\left(y_{Gi}\cdot W_{li}-y_{Bi}\cdot B_{li}\right)-c\theta s\varphi\left(z_{Gi}\cdot W_{li}-z_{Bi}\cdot B_{li}\right)\right)\\ \sum_{i=1}^{n_m}\left(-s\theta\left(z_{Gi}\cdot W_{li}-z_{Bi}\cdot B_{li}\right)-c\theta c\varphi\left(x_{Gi}\cdot W_{li}-x_{Bi}\cdot B_{li}\right)\right)\\ \sum_{i=1}^{n_m}\left(c\theta s\varphi\left(x_{Gi}\cdot W_{li}-x_{Bi}\cdot B_{li}\right)+s\theta\left(y_{Gi}\cdot W_{li}-y_{Bi}\cdot B_{li}\right)\right)\end{bmatrix} \tag{2.64}$$

**Restoring vector of manipulator**

Consequently, we proceed to express analytically the restoring vector of the manipulator, i.e. $\boldsymbol{g}_m\left(\boldsymbol{\eta}_2,\boldsymbol{q}_m\right)$.

The potential energy of link *i* due to gravity and buoyancy is:

$$U_{li}=-m_{li}\cdot\boldsymbol{g}_0{}^T\cdot\boldsymbol{p}_{Gi}+\rho\cdot V_{li}\cdot\boldsymbol{g}_0{}^T\cdot\boldsymbol{p}_{Bi} \tag{2.65}$$

where,

$\boldsymbol{p}_{Gi}$ is the position vector of the center of gravity of link *i* with respect to {0},

$\boldsymbol{p}_{Bi}$ is the position vector of the center of buoyancy of link *i* with respect to {0} and

the gravitational acceleration vector $\boldsymbol{g}_0$ expressed in the manipulator base frame {0} is defined as follows:

$$\boldsymbol{g}_0={}^0\boldsymbol{R}_I\cdot\boldsymbol{g}_I={}^0\boldsymbol{R}_I\cdot\begin{bmatrix}0\\0\\g\end{bmatrix} \tag{2.66}$$

where,

${}^0\boldsymbol{R}_I$ is the rotation matrix from inertial frame {*I*} to manipulator base frame {0}.

The position vectors of the centers of gravity and buoyancy of link *i* with respect to {0} could be expressed as follows:

$$\boldsymbol{p}_{Gi} = \boldsymbol{p}_{i-1} + \lambda_{Gi}\left(\boldsymbol{p}_i - \boldsymbol{p}_{i-1}\right) \quad ,i = 1,...,n_m \tag{2.67}$$

$$\boldsymbol{p}_{Bi} = \boldsymbol{p}_{i-1} + \lambda_{Bi}\left(\boldsymbol{p}_i - \boldsymbol{p}_{i-1}\right) \quad ,i = 1,...,n_m \tag{2.68}$$

where,

$\boldsymbol{p}_i$ is the position vector of the origin of coordinate frame {i} with respect to manipulator base frame {0}, derived from the above coordinate transformation matrix:

$${}^0\boldsymbol{T}_i = \begin{bmatrix} & & \vdots & \\ & {}^0\boldsymbol{R}_i & \vdots & \boldsymbol{p}_i \\ .... & ..... & .... & .... \\ & \boldsymbol{0}_{1\times 3} & \vdots & 1 \end{bmatrix} \quad ,i = 1,...,n_m \tag{2.69}$$

The overall potential energy of the manipulator is:

$$U = \sum_{j=1}^{n_m}\left(-m_{lj}\cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{p}_{Gj} + \rho \cdot V_{lj} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{p}_{Bj}\right) \tag{2.70}$$

According to Lagrangian dynamic formulation, the restoring generalized forces are expressed by:

$$\frac{\partial U}{\partial q_i} = \sum_{j=1}^{n_m}\left(-m_{lj}\cdot \boldsymbol{g}_0{}^T \cdot \frac{\partial \boldsymbol{p}_{Gj}}{\partial q_i} + \rho \cdot V_{lj} \cdot \boldsymbol{g}_0{}^T \cdot \frac{\partial \boldsymbol{p}_{Bj}}{\partial q_i}\right) \quad \Rightarrow$$

$$\frac{\partial U}{\partial q_i} = \sum_{j=1}^{n_m}\left(-m_{lj}\cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Gj)} + \rho \cdot V_{lj} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Bj)}\right) \triangleq g_i\left(\boldsymbol{q}_m\right) \tag{2.71}$$

where $\boldsymbol{J}_{Pi}^{(Gj)}$ and $\boldsymbol{J}_{Pi}^{(Bj)}$ are the (3x1) column vectors of the manipulator geometric Jacobian expressing the contributions of the j-th joint velocity to the linear velocities of the i-th centers of gravity and buoyancy, respectively. It holds:

$$\boldsymbol{J}_{Pj}^{(Gi)} = \boldsymbol{z}_{j-1} \times \left( \boldsymbol{p}_{Gi} - \boldsymbol{p}_{j-1} \right) \tag{2.72}$$

$$\boldsymbol{J}_{Pj}^{(Bi)} = \boldsymbol{z}_{j-1} \times \left( \boldsymbol{p}_{Bi} - \boldsymbol{p}_{j-1} \right) \tag{2.73}$$

where $\boldsymbol{z}_{j-1}$ is the unit vector of z-axis of coordinate frame {j-1}.

For the sake of ease of comprehension, it seems useful to express analytically the linear velocities of each center of gravity and buoyancy:

$$\dot{\boldsymbol{p}}_{G1} = \boldsymbol{J}_{P1}^{(G1)} \cdot \dot{q}_1 + 0 \cdot \dot{q}_2 + .... + 0 \cdot \dot{q}_{n_m} \tag{2.74a}$$

$$\dot{\boldsymbol{p}}_{Gi} = \boldsymbol{J}_{P1}^{(Gi)} \cdot \dot{q}_1 + .... + \boldsymbol{J}_{Pi}^{(Gi)} \cdot \dot{q}_i + 0 \cdot \dot{q}_{i+1} + .... + 0 \cdot \dot{q}_{n_m} \tag{2.74b}$$

$$\dot{\boldsymbol{p}}_{Gn_m} = \boldsymbol{J}_{P1}^{(Gn_m)} \cdot \dot{q}_1 + \boldsymbol{J}_{P2}^{(Gn_m)} \cdot \dot{q}_2 + .... + \boldsymbol{J}_{Pn_m}^{(Gn_m)} \cdot \dot{q}_{n_m} \tag{2.74c}$$

$$\dot{\boldsymbol{p}}_{B1} = \boldsymbol{J}_{P1}^{(B1)} \cdot \dot{q}_1 + 0 \cdot \dot{q}_2 + .... + 0 \cdot \dot{q}_{n_m} \tag{2.75a}$$

$$\dot{\boldsymbol{p}}_{Bi} = \boldsymbol{J}_{P1}^{(Bi)} \cdot \dot{q}_1 + .... + \boldsymbol{J}_{Pi}^{(Bi)} \cdot \dot{q}_i + 0 \cdot \dot{q}_{i+1} + .... + 0 \cdot \dot{q}_{n_m} \tag{2.75b}$$

$$\dot{\boldsymbol{p}}_{Bn_m} = \boldsymbol{J}_{P1}^{(Bn_m)} \cdot \dot{q}_1 + \boldsymbol{J}_{P2}^{(Bn_m)} \cdot \dot{q}_2 + .... + \boldsymbol{J}_{Pn_m}^{(Bn_m)} \cdot \dot{q}_{n_m} \tag{2.75c}$$

Therefore, the restoring vector of the manipulator is defined as follows:

$$\boldsymbol{g}_m \left( \boldsymbol{\eta}_2, \boldsymbol{q}_m \right) = \begin{bmatrix} g_1 \\ \vdots \\ g_i \\ \vdots \\ g_{n_m} \end{bmatrix} \tag{2.76}$$

where,

$$g_1 = \frac{\partial U}{\partial q_1} = -m_{l1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{P1}^{(G1)} + \rho \cdot V_{l1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{P1}^{(B1)} -$$

$$\vdots$$

$$- m_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{P1}^{(Gn_m)} + \rho \cdot V_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{P1}^{(Bn_m)} =$$

$$= -m_{l1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_0 \times \left( \boldsymbol{p}_{G1} - \boldsymbol{p}_0 \right) + \rho \cdot V_{l1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_0 \times \left( \boldsymbol{p}_{B1} - \boldsymbol{p}_0 \right) -$$

$$\vdots$$

$$- m_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_0 \times \left( \boldsymbol{p}_{Gn_m} - \boldsymbol{p}_0 \right) + \rho \cdot V_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_0 \times \left( \boldsymbol{p}_{Bn_m} - \boldsymbol{p}_0 \right) \tag{2.77a}$$

$$g_i = \frac{\partial U}{\partial q_i} = -m_{li} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Gi)} + \rho \cdot V_{li} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Bi)}$$
$$- m_{li+1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Bi+1)} + \rho \cdot V_{li+1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Bi+1)}$$
$$\vdots$$
$$- m_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Gn_m)} + \rho \cdot V_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pi}^{(Bn_m)} \quad =$$

$$= -m_{li} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{i-1} \times \left( \boldsymbol{p}_{Gi} - \boldsymbol{p}_{i-1} \right) + \rho \cdot V_{li} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{i-1} \times \left( \boldsymbol{p}_{Bi} - \boldsymbol{p}_{i-1} \right)$$
$$- m_{li+1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{i-1} \times \left( \boldsymbol{p}_{Gi+1} - \boldsymbol{p}_{i-1} \right) + \rho \cdot V_{li+1} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{i-1} \times \left( \boldsymbol{p}_{Bi+1} - \boldsymbol{p}_{i-1} \right)$$
$$\vdots$$
$$- m_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{i-1} \times \left( \boldsymbol{p}_{Gn_m} - \boldsymbol{p}_{i-1} \right) + \rho \cdot V_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{i-1} \times \left( \boldsymbol{p}_{Bn_m} - \boldsymbol{p}_{i-1} \right) \qquad (2.77b)$$

$$g_{n_m} = \frac{\partial U}{\partial q_{n_m}} = -m_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pn_m}^{(Gn_m)} + \rho \cdot V_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{J}_{Pn_m}^{(Bn_m)} =$$
$$= - m_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{n_m-1} \times \left( \boldsymbol{p}_{Gn_m} - \boldsymbol{p}_{n_m-1} \right) + \rho \cdot V_{ln_m} \cdot \boldsymbol{g}_0{}^T \cdot \boldsymbol{z}_{n_m-1} \times \left( \boldsymbol{p}_{Bn_m} - \boldsymbol{p}_{n_m-1} \right) \qquad (2.77c)$$

**Restoring vector of UVMS**

So, the restoring vector of an UVMS is:

$$\boldsymbol{g}\left(\boldsymbol{\eta}_2, \boldsymbol{q}_m\right) = \begin{bmatrix} \boldsymbol{g}_v\left(\boldsymbol{\eta}_2\right) + \boldsymbol{g}_s\left(\boldsymbol{\eta}_2, \boldsymbol{q}_m\right) \\ \boldsymbol{g}_m\left(\boldsymbol{\eta}_2, \boldsymbol{q}_m\right) \end{bmatrix} \quad \Rightarrow$$

$$\boldsymbol{g}\left(\boldsymbol{\eta}_2, \boldsymbol{q}_m\right) = \begin{bmatrix} -\left( {}^V\boldsymbol{f}_{Gv} + {}^V\boldsymbol{f}_{Bv} + \sum_{i=1}^{n_m} \left( {}^V\boldsymbol{f}_{Gi} + {}^V\boldsymbol{f}_{Bi} \right) \right) \\ -\left( {}^V\boldsymbol{r}_{Gv} \times {}^V\boldsymbol{f}_{Gv} + {}^V\boldsymbol{r}_{Bv} \times {}^V\boldsymbol{f}_{Bv} + \sum_{i=1}^{n_m} \left( {}^V\boldsymbol{r}_{Gi} \times {}^V\boldsymbol{f}_{Gi} + {}^V\boldsymbol{r}_{Bi} \times {}^V\boldsymbol{f}_{Bi} \right) \right) \\ g_1 \\ \vdots \\ g_{n_m} \end{bmatrix} \qquad (2.78)$$

# Chapter 3

# Solution Process

## 3.1 Problem Formulation

As it has already been stated in Chapter 1, the ultimate objective of this thesis is the design of a control scheme that provides an optimal pose configuration of UVMS for efficient interaction with the environment wrt a certain performance criterion, ensuring that several constraints are satisfied and exploiting the redundant dofs of the combined system.

The proposed algorithm could be embedded in the interaction control structure of UVMS performing intervention tasks that involve interaction with the environment. In particular, the proposed optimization algorithm will be incorporated as part of a two-stage interaction control structure composed of a path planning algorithm and a robust coordinated motion controller.

When the UVMS approaches the interaction spot, an image based controller using model predictive control (MPC) is used to generate a sequence of variables over time that describe end-effector position and orientation, navigating it towards the target point. Since the control action on the UVMS is carried out in the joint space, a suitable optimization algorithm will be designed to provide in real time the joint space variables corresponding to the desired end-effector pose denoted by the image based controller. The UVMS pose configuration will be optimal wrt an objective function, the maximization of which guarantees the maximization of a meaningfully defined norm of the interaction wrench vector. At the same time, the proposed optimization scheme, exploiting the kinematic redundancy of UVMS, will satisfy several constraints imposed by the mechanical structure of the UVMS and the surrounding physical environment as well as hardware limitations of the system concerning the limited capability range of the actuators.

In other words, the image based controller as a path planning algorithm in the operational space produces in real time a sequence of end-effector poses from the initial pose, when the UVMS enters the "*reach-to-grasp*" phase, to the final target pose, when the end effector is ready to interact with the environment. On the other hand, the proposed optimization scheme plays the role of a path planner in the joint space receiving as input the above sequence of end effector poses and producing a sequence of UVMS pose configurations that leads in a smooth way to the final optimal configuration for efficient interaction wrt the pre-specified performance criterion.

In this section, the problem of producing the appropriate UVMS pose configuration for efficient interaction with the environment is being modeled. The basic parameters and quantities as well as the analytical expressions of the performance criterion and the various constraints that will be later used in the proposed optimization scheme are defined, considering the kinematic and dynamic analysis of the UVMS presented in Chapter 2 and adopting the corresponding notations.

### 3.1.1 Models and Definitions

For the sake of completeness, some of the pose and velocity vectors presented in §2.2 and used in the following analysis are redefined. In particular, the vehicle pose vector is defined by the vector $\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{\eta}_1^T & \boldsymbol{\eta}_2^T \end{bmatrix}^T \in \mathbb{R}^6$, where $\boldsymbol{\eta}_1 = \begin{bmatrix} x & y & z \end{bmatrix}^T \in \mathbb{R}^3$ is the position vector of the vehicle-fixed frame {*V*} relative to the inertial frame {*I*} and $\boldsymbol{\eta}_2 = \begin{bmatrix} \varphi & \theta & \psi \end{bmatrix}^T \in \mathbb{R}^3$ is the vector of Euler-angles of the vehicle-fixed frame {*V*} relative to the inertial frame {*I*}. The end-effector pose vector is defined by the vector $\boldsymbol{p}_e = \begin{bmatrix} \boldsymbol{p}_{e1}^T & \boldsymbol{p}_{e2}^T \end{bmatrix}^T \in \mathbb{R}^6$, where $\boldsymbol{p}_{e1} = \begin{bmatrix} x_e & y_e & z_e \end{bmatrix}^T \in \mathbb{R}^3$ is the position vector of the e-e frame {*E*} expressed in the inertial frame {*I*} and $\boldsymbol{p}_{e2} = \begin{bmatrix} \varphi_e & \theta_e & \psi_e \end{bmatrix}^T \in \mathbb{R}^3$ is the orientation (Euler-angles) vector of the e-e frame {*E*} expressed in the inertial frame {*I*}. Moreover, for a *n*-link manipulator, the joint angular position state vector is defined by $\boldsymbol{q}_m = \begin{bmatrix} q_1 & q_2 & \dots & q_{n_m} \end{bmatrix}^T \in \mathbb{R}^{n_m}$. Finally, the UVMS pose configuration vector is defined as $\boldsymbol{q} = \begin{bmatrix} \boldsymbol{\eta}^T & \boldsymbol{q}_m^T \end{bmatrix}^T \in \mathbb{R}^{n_m+6}$.

As far as the body-fixed system velocity is concerned, the vector $\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\nu}^T & \dot{\boldsymbol{q}}_m^T \end{bmatrix}^T \in \mathbb{R}^{n_m+6}$ is defined, where $\boldsymbol{\nu} = \begin{bmatrix} \boldsymbol{\nu}_1^T & \boldsymbol{\nu}_2^T \end{bmatrix}^T \in \mathbb{R}^6$ is the vehicle-fixed

velocity vector of the vehicle, $\boldsymbol{v}_1 = \begin{bmatrix} u & v & w \end{bmatrix}^T \in \mathbb{R}^3$ is the linear velocity of the vehicle-fixed frame {*V*} with respect to the inertial frame {*I*} expressed in the vehicle-fixed frame, $\boldsymbol{v}_2 = \begin{bmatrix} p & q & r \end{bmatrix}^T \in \mathbb{R}^3$ is the angular velocity of the vehicle-fixed frame {*V*} with respect to the inertial frame {*I*} expressed in the vehicle-fixed frame and $\dot{\boldsymbol{q}}_m = \begin{bmatrix} \dot{q}_1 & \dot{q}_2 & \dots & \dot{q}_{n_m} \end{bmatrix}^T \in \mathbb{R}^{n_m}$ is the joint angular velocity vector of the *n*-link manipulator.

Now, considering an UVMS that is needed to interact with the environment, the following dynamic equations of motion expressed in the vehicle-fixed frame {*V*} hold, as expressed in Eq. 2.35.

$$\boldsymbol{M}(\boldsymbol{q}_m)\cdot\dot{\boldsymbol{\zeta}} + \boldsymbol{C}(\boldsymbol{\zeta},\boldsymbol{q}_m)\cdot\boldsymbol{\zeta} + \boldsymbol{D}(\boldsymbol{\zeta},\boldsymbol{q}_m)\cdot\boldsymbol{\zeta} + \boldsymbol{g}(\boldsymbol{\eta}_2,\boldsymbol{q}_m) + \boldsymbol{J}_W^T(\boldsymbol{\eta}_2,\boldsymbol{q}_m)\cdot\boldsymbol{h}_e = \boldsymbol{\tau} \qquad (3.1)$$

Under the assumption that the UVMS is stationary while it is applying a force/moment $\boldsymbol{h}_e$ to an object, the UVMS dynamics can be reduced to statics. Thus, considering that $\boldsymbol{\zeta} = \dot{\boldsymbol{\zeta}} = 0$ the following statics equation of the UVMS is derived from Eq. 3.1:

$$\boldsymbol{g}(\boldsymbol{\eta}_2,\boldsymbol{q}_m) + \boldsymbol{J}_W^T(\boldsymbol{\eta}_2,\boldsymbol{q}_m)\cdot\boldsymbol{h}_e = \boldsymbol{\tau} \qquad (3.2)$$

where,

$\boldsymbol{g}(\boldsymbol{\eta}_2,\boldsymbol{q}_m) \in \mathbb{R}^{n_m+6}$ is the vector of gravity and buoyancy generalized forces of the UVMS,

$\boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{\tau}_v^T & \boldsymbol{\tau}_m^T \end{bmatrix}^T \in \mathbb{R}^{n_m+6}$ is the vector of the propulsion forces and moments $\boldsymbol{\tau}_v \in \mathbb{R}^6$ acting on the vehicle in the vehicle-fixed frame as well as the joint torques $\boldsymbol{\tau}_m \in \mathbb{R}^{n_m}$,

$\boldsymbol{J}_W(\boldsymbol{\eta}_2,\boldsymbol{q}_m) \in \mathbb{R}^{6\times(n_m+6)}$ is the Jacobian matrix defined in Eq. 2.24,

$\boldsymbol{h}_e = \begin{bmatrix} \boldsymbol{f}_e^T & \boldsymbol{\mu}_e^T \end{bmatrix}^T \in \mathbb{R}^6$ is the end-effector wrench vector expressed in the inertial frame and $\boldsymbol{f}_e, \boldsymbol{\mu}_e \in \mathbb{R}^3$ are the interaction force and torque vectors respectively.

More specifically, the restoring forces vector $\boldsymbol{g}(\boldsymbol{\eta}_2,\boldsymbol{q}_m)$ of an UVMS equipped with a *n*-link manipulator as has already been defined in Eq. 2.78, is expressed by the following relation:

$$\boldsymbol{g}\left(\boldsymbol{\eta}_2,\boldsymbol{q}_m\right)=\begin{bmatrix} -\left({}^{V}\boldsymbol{f}_{Gv}+{}^{V}\boldsymbol{f}_{Bv}+\sum_{i=1}^{n_m}\left({}^{V}\boldsymbol{f}_{Gi}+{}^{V}\boldsymbol{f}_{Bi}\right)\right) \\ -\left({}^{V}\boldsymbol{r}_{Gv}\times{}^{V}\boldsymbol{f}_{Gv}+{}^{V}\boldsymbol{r}_{Bv}\times{}^{V}\boldsymbol{f}_{Bv}+\sum_{i=1}^{n_m}\left({}^{V}\boldsymbol{r}_{Gi}\times{}^{V}\boldsymbol{f}_{Gi}+{}^{V}\boldsymbol{r}_{Bi}\times{}^{V}\boldsymbol{f}_{Bi}\right)\right) \\ g_1 \\ \vdots \\ g_{n_m} \end{bmatrix} \tag{3.3}$$

where the analytical expressions of the terms that each component of the vector is consisted of, are stated in §2.4.

Furthermore, the UVMS Jacobian matrix $\boldsymbol{J}_W\left(\boldsymbol{\eta}_2,\boldsymbol{q}_m\right)$ defined in Eq. 2.24 is restated below:

$$\dot{\boldsymbol{p}}_e=\boldsymbol{J}_W\left(\boldsymbol{\eta}_2,\boldsymbol{q}\right)\cdot\boldsymbol{\zeta} \tag{3.4}$$

and

$$\boldsymbol{J}_W\left(\boldsymbol{\eta}_2,\boldsymbol{q}\right)=\begin{bmatrix} \boldsymbol{J}_{v1}\left(\boldsymbol{\eta}_2\right) & -{}^{I}\boldsymbol{R}_V\cdot S\left(\boldsymbol{p}_1\right) & {}^{I}\boldsymbol{R}_0\cdot\boldsymbol{J}_{m1} \\ \boldsymbol{0} & \boldsymbol{J}_{v2}\left(\boldsymbol{p}_{e2}\right)\cdot{}^{E}\boldsymbol{R}_V & \boldsymbol{J}_{v2}\left(\boldsymbol{p}_{e2}\right)\cdot{}^{E}\boldsymbol{R}_0\cdot\boldsymbol{J}_{m2} \end{bmatrix} \tag{3.5}$$

where each element comprising the UVMS Jacobian matrix is defined in detail in §2.2.

As it has already been mentioned, when UVMS are expected to perform underwater intervention tasks requiring interaction with the environment two issues to be considered arise. The UVMS must simultaneously control the position of the end-effector and the force applied to the environment. However, in most cases, although the interaction wrench vector (combined force and torque) may be known in terms of direction, it is usually unknown in terms of magnitude (e.g. the torque needed to turn a valve may change in time). Besides, when approaching the interaction spot (i.e. during the "*reach-to-grasp*" phase) or when interacting, disturbances possibly induced by underwater currents should be rejected. Since the UVMS actuation system will need to undertake simultaneously those issues, our approach aims to find out a pose configuration of the combined vehicle–manipulator system that it allows for "maximization" of the allowed interaction wrench between the UVMS and the

environment. Thus, we have to select an appropriate objective function to maximize in our proposed optimization algorithm.

Let define the end-effector unit wrench vector expressed in {*I*} as follows:

$$\boldsymbol{e}_h = \begin{bmatrix} \boldsymbol{e}_f{}^T & \boldsymbol{e}_\mu{}^T \end{bmatrix}^T \in \mathbb{R}^6 \quad \begin{matrix} \nearrow \ \boldsymbol{e}_f \in \mathbb{R}^3 \\ \searrow \ \boldsymbol{e}_\mu \in \mathbb{R}^3 \end{matrix}$$

The force and torque magnitudes are assigned as $\lambda_f$ and $\lambda_\mu$ respectively and the corresponding vector is defined by $\boldsymbol{\lambda} = \begin{bmatrix} \lambda_f & \lambda_\mu \end{bmatrix}^T \in \mathbb{R}^2$. The following holds:

$$\boldsymbol{h}_e = \begin{bmatrix} \boldsymbol{f}_e \\ \boldsymbol{\mu}_e \end{bmatrix} = \begin{bmatrix} \lambda_f \cdot \boldsymbol{e}_f \\ \lambda_\mu \cdot \boldsymbol{e}_\mu \end{bmatrix} = \begin{bmatrix} \lambda_f \cdot \boldsymbol{I}_3 & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \lambda_\mu \cdot \boldsymbol{I}_3 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{e}_f \\ \boldsymbol{e}_\mu \end{bmatrix} = \boldsymbol{\Lambda} \cdot \boldsymbol{e}_h \qquad (3.6)$$

Consequently, in order to find out the appropriate objective function to ensure the maximization of the interaction wrench the end-effector could apply, several types of norms of vector $\boldsymbol{\lambda}$ can be considered:

- Maximum norm:

$$\|\boldsymbol{\lambda}\|_\infty = \max\left\{|\lambda_f|, |\lambda_\mu|\right\} \qquad (3.7)$$

- Euclidean norm with $w$ appropriate weight factor:

$$\|\boldsymbol{\lambda}\|_2 = \left((1-w)\cdot|\lambda_f|^2 + w\cdot|\lambda_\mu|^2\right)^{1/2} \qquad (3.8)$$

- 1- norm with $w$ appropriate weight factor:

$$\|\boldsymbol{\lambda}\|_1 = (1-w)\cdot|\lambda_f| + w\cdot|\lambda_\mu| \qquad (3.9)$$

Now, regarding the constraints that need to be satisfied, as already mentioned the proposed optimization algorithm will be designed to provide the joint space variables of the UVMS ensuring the desired pose of the end-effector. Considering that the combined vehicle-manipulator system forms an open kinematic chain, the end-effector pose $\boldsymbol{p}_e \in \mathbb{R}^6$ expressed in the inertial frame can be computed as a function of the UVMS joint

space state vector $\boldsymbol{q} \in \mathbb{R}^{n_m+6}$ through the forward kinematics. The forward kinematics function, which is actually the coordinate transformation describing the position and orientation of the end-effector frame with respect to the inertial frame, is obtained in a systematic manner by simple products of the following homogeneous transformation matrices.

$$ {}^{I}\boldsymbol{T}_{E}\left(\boldsymbol{q}\right) = {}^{I}\boldsymbol{T}_{V}\left(\boldsymbol{\eta}\right) \cdot {}^{V}\boldsymbol{T}_{0} \cdot {}^{0}\boldsymbol{T}_{1}\left(\boldsymbol{q}_{1}\right) \cdot \ \ldots \ \cdot {}^{n_m}\boldsymbol{T}_{E}\left(q_{1},\ldots,q_{n_m}\right) \tag{3.10} $$

Consequently, if we define the UVMS forward kinematics functions as follows:

- End-effector position: $\boldsymbol{p}_{e1}\left(\boldsymbol{q}\right): \mathbb{R}^{n_m+6} \rightarrow \mathbb{R}^{3}$, and
- End-effector orientation: $\boldsymbol{p}_{e2}\left(\boldsymbol{q}\right): \mathbb{R}^{n_m+6} \rightarrow \mathbb{R}^{3}$,

then the end-effector position and orientation constraints are:

$$ \boldsymbol{p}_{e1}\left(\boldsymbol{q}\right) = \boldsymbol{p}_{e1,d} \tag{3.11} $$

$$ \boldsymbol{p}_{e2}\left(\boldsymbol{q}\right) = \boldsymbol{p}_{e2,d} \tag{3.12} $$

Besides, the proposed algorithm has to lead to configurations that are compatible with several constraints imposed by the mechanical structure of the combined system, the structure of the surrounding physical environment as well as the grasped object's properties and also geometric constraints ensuring that there are no collisions between various modules of the system.

Obviously, regarding the mechanical constraints, the joint angle vector $\boldsymbol{q}_{m} \in \mathbb{R}^{n_m}$ of the manipulator is bounded with respect to the angular limitations of the joint motors. Moreover, the orientation angles of the vehicle $\boldsymbol{\eta}_{2} = \left[\varphi \ \theta \ \psi\right]^{T} \in \mathbb{R}^{3}$ have to be kept bounded for various reasons. Next, although there are no mechanical reasons to consider any limitations for the position of the vehicle, the physical environment as well as the geometry of the target objects may set some inequality constraints for the position vector of the position vector $\boldsymbol{\eta}_{1} = \left[x \ y \ z\right]^{T} \in \mathbb{R}^{3}$. Thus, the aforementioned constraints can be formulated as follows:

$$\boldsymbol{q}_{\min} \le \boldsymbol{q} \le \boldsymbol{q}_{\max} \tag{3.13}$$

where $\boldsymbol{q}_{\min}, \boldsymbol{q}_{\max} \in \mathbb{R}^{n_m+6}$ represent the lower and upper bounds of the UVMS pose configuration vector respectively.

In regard to the geometric constraints ensuring that there are no collisions between various modules of the system, those can be formulated as linear or nonlinear inequality constraints as follows:

$$\boldsymbol{A}_{ineq} \cdot \boldsymbol{q} \le \boldsymbol{b}_{ineq} \tag{3.14}$$

$$\boldsymbol{G}(\boldsymbol{q}) \le \boldsymbol{0} \tag{3.15}$$

Finally, the additional hardware limitations concerning the limited capability range of the actuators will be formulated. Let us denote the vector of the control inputs of the UVMS actuators by:

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_v^{\,T} & \boldsymbol{u}_m^{\,T} \end{bmatrix}^T \in \mathbb{R}^{p_v+n_m} \quad \begin{matrix} \nearrow \ \boldsymbol{u}_v \in \mathbb{R}^{p_v} \\ \searrow \ \boldsymbol{u}_m \in \mathbb{R}^{n_m} \end{matrix}$$

where,

$\boldsymbol{u}_v \in \mathbb{R}^{p_v}$ is the vector of the control inputs of the AUV thrusters,

$\boldsymbol{u}_m \in \mathbb{R}^{n_m}$ is the vector of the control inputs of the manipulator motors.

The relationship between the force/moment acting on the vehicle $\boldsymbol{\tau}_v$ and the control inputs of the vehicle thrusters $\boldsymbol{u}_v \in \mathbb{R}^{p_v}$ is highly nonlinear. A common simplification is to consider a linear relationship between $\boldsymbol{\tau}_v$ and $\boldsymbol{u}_v$:

$$\boldsymbol{\tau}_v = \boldsymbol{S}_v \cdot \boldsymbol{u}_v \tag{3.16}$$

where $\boldsymbol{S}_v \in \mathbb{R}^{6\times p_v}$ is a constant matrix describing the thruster allocation, known as the Thruster Configuration Matrix (TCM).

The relationship between the generalized forces $\boldsymbol{\tau} \in \mathbb{R}^{n_m+6}$ and the control inputs $\boldsymbol{u} \in \mathbb{R}^{p_V+n_m}$ is described by the following Thruster Configuration Matrix, $\boldsymbol{S} \in \mathbb{R}^{(n_m+6)\times(p_V+n_m)}$, for the whole UVMS :

$$\boldsymbol{\tau} = \boldsymbol{S} \cdot \boldsymbol{u} \quad \Rightarrow$$

$$\begin{bmatrix} \boldsymbol{\tau}_v \\ \boldsymbol{\tau}_m \end{bmatrix} = \begin{bmatrix} \boldsymbol{S}_v & \boldsymbol{0}_{6\times n_m} \\ \boldsymbol{0}_{n_m\times p_V} & \boldsymbol{I}_{n_m} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{u}_v \\ \boldsymbol{u}_m \end{bmatrix} \tag{3.17}$$

Notice that for the manipulator it is supposed that $n_m$ joint motors are available.

Let us consider that the control input vector $\boldsymbol{u} \in \mathbb{R}^{p_V+n_m}$ has the following limits:

$$\boldsymbol{u}_{\min} \le \boldsymbol{u} \le \boldsymbol{u}_{\max} \quad \Rightarrow \quad \begin{bmatrix} \boldsymbol{u}_{\min\_th} \\ \boldsymbol{u}_{\min\_m} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{u}_v \\ \boldsymbol{u}_m \end{bmatrix} \le \begin{bmatrix} \boldsymbol{u}_{\max\_th} \\ \boldsymbol{u}_{\max\_m} \end{bmatrix} \quad \Rightarrow$$

$$\begin{bmatrix} u_{\min\_th_1} \\ \vdots \\ u_{\min\_th_i} \\ \vdots \\ u_{\min\_th_{p_V}} \\ u_{\min\_m_1} \\ \vdots \\ u_{\min\_m_i} \\ \vdots \\ u_{\min\_m_{n_m}} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{u}_v \\ \boldsymbol{u}_m \end{bmatrix} \le \begin{bmatrix} u_{\max\_th_1} \\ \vdots \\ u_{\max\_th_i} \\ \vdots \\ u_{\max\_th_{p_V}} \\ u_{\max\_m_1} \\ \vdots \\ u_{\max\_m_i} \\ \vdots \\ u_{\max\_m_{n_m}} \end{bmatrix} \tag{3.18}$$

where $u_{\min\_th_i}$ and $u_{\max\_th_i}$ are the lower and upper limits of the control input of the AUV *i*-thruster respectively, while $u_{\min\_m_i}$ and $u_{\max\_m_i}$ are the lower and upper limits of the control input of the manipulator *i*-motor respectively.

Thus, it turns out that the limits for the generalized forces $\boldsymbol{\tau} \in \mathbb{R}^{n_m+6}$ are given by:

$$\boldsymbol{\tau}_{\min} \le \boldsymbol{\tau} \le \boldsymbol{\tau}_{\max} \quad \Rightarrow \quad \begin{bmatrix} \boldsymbol{\tau}_{\min\_v} \\ \boldsymbol{\tau}_{\min\_m} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{\tau}_v \\ \boldsymbol{\tau}_m \end{bmatrix} \le \begin{bmatrix} \boldsymbol{\tau}_{\max\_v} \\ \boldsymbol{\tau}_{\max\_m} \end{bmatrix} \quad \Rightarrow$$

$$\begin{bmatrix} \tau_{\min\_v_x} \\ \tau_{\min\_v_y} \\ \tau_{\min\_v_z} \\ \tau_{\min\_v_\varphi} \\ \tau_{\min\_v_\theta} \\ \tau_{\min\_v_\psi} \\ u_{\min\_m_1} \\ \vdots \\ u_{\min\_m_i} \\ \vdots \\ u_{\min\_m_{n_m}} \end{bmatrix} \leq \begin{bmatrix} \boldsymbol{\tau}_v \\ \boldsymbol{\tau}_m \end{bmatrix} = \boldsymbol{S} \cdot \begin{bmatrix} \boldsymbol{u}_v \\ \boldsymbol{u}_m \end{bmatrix} \leq \begin{bmatrix} \tau_{\max\_v_x} \\ \tau_{\max\_v_y} \\ \tau_{\max\_v_z} \\ \tau_{\max\_v_\varphi} \\ \tau_{\max\_v_\theta} \\ \tau_{\max\_v_\psi} \\ u_{\max\_m_1} \\ \vdots \\ u_{\max\_m_i} \\ \vdots \\ u_{\max\_m_{n_m}} \end{bmatrix} \tag{3.19}$$

where $\tau_{\min\_v_x}$ and $\tau_{\max\_v_x}$ are the lower and upper limits of the propulsion force acting along the surge axis of the vehicle, while the corresponding lower and upper limits of the propulsion forces acting along the sway and heave axes as well as the limits of the rolling, pitching and yawing propulsion moments are denoted accordingly.

It is important to point out that the bounds of the propulsion forces and moments $\boldsymbol{\tau}_{\min\_v}, \boldsymbol{\tau}_{\max\_v} \in \mathbb{R}^6$ acting on the vehicle depend on the limits of the thrusters' control inputs $\boldsymbol{u}_{\min\_th}, \boldsymbol{u}_{\max\_th} \in \mathbb{R}^6$ and the Thruster Configuration Matrix of the vehicle.

Considering the limits of the generalized forces and the statics equation of the UVMS (Eq. 3.2), the following constraints, regarding the limited capability range of the actuators, turn out:

$$\boldsymbol{\tau}_{\min} \leq \boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) + \boldsymbol{J}_W^T(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{h}_e \leq \boldsymbol{\tau}_{\max} \tag{3.20}$$

If $\boldsymbol{J}_W^T$ is partitioned appropriately then one can easily deduce that term $\boldsymbol{J}_W^T \cdot \boldsymbol{h}_e$ of inequality (3.20), in conjunction to Eq. 3.6, is written as:

$$\boldsymbol{J}_W^T(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{h}_e = \begin{bmatrix} \boldsymbol{J}_f(\boldsymbol{\eta}_2, \boldsymbol{q}_m) & \boldsymbol{J}_\mu(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{f}_e \\ \boldsymbol{\mu}_e \end{bmatrix} = \boldsymbol{E}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{\lambda} \tag{3.21}$$

where $\boldsymbol{E} \in \mathbb{R}^{(n_m+6)\times 2}$ is defined as:

$$\boldsymbol{E}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) = \left[ \boldsymbol{J}_f(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{e}_f \quad \boldsymbol{J}_\mu(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{e}_\mu \right] \tag{3.22}$$

Using Eq. 3.21, inequality (3.20) is restated as:

$$\boldsymbol{\tau}_{\min} \le \boldsymbol{g}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) + \boldsymbol{E}(\boldsymbol{\eta}_2, \boldsymbol{q}_m) \cdot \boldsymbol{\lambda} \le \boldsymbol{\tau}_{\max} \tag{3.23}$$

### 3.1.2 Summary of Adopted Notations

In this section the most significant notations adopted so far are summarized (Table 3.1). In the following sections the developed methodologies will be presented in a general form, with the symbols and notations of this section.

**Table 3.1:** Adopted notations and their definitions

| Notation | Definition |
|---|---|
| $n_m$ | Number of manipulator joints |
| $p_v$ | Number of vehicle thrusters |
| $\{I\}$ | Earth-fixed or inertial reference frame |
| $\{V\}$ | Vehicle-fixed or body-fixed reference frame |
| $\{0\}$ | Reference frame located at the manipulator base |
| $\{i\}$ | Reference frame located at the i-th link of the manipulator along the D-H convention |
| $\{E\}$ | End-effector fixed reference frame |
| ${}^{j}\boldsymbol{R}_i \in SO(3)$ | Rotation matrix describing the orientation of frame {i} with respect to the frame {j} |
| ${}^{j}\boldsymbol{T}_i \in SE(3)$ | Homogeneous transformation matrix describing the position and orientation of frame {i} with respect to the frame {j} |
| $\boldsymbol{\eta} \in \mathbb{R}^6$ | Vehicle pose vector expressed in the $\{I\}$ |
| $\boldsymbol{\eta}_1 \in \mathbb{R}^3$ | Position vector of the $\{V\}$ expressed in the $\{I\}$ |
| $\boldsymbol{\eta}_2 \in \mathbb{R}^3$ | Vector of Euler-angles of the $\{V\}$ expressed in the $\{I\}$ |
| $\boldsymbol{v} \in \mathbb{R}^6$ | Vehicle-fixed velocity vector of the vehicle |

| | |
|---|---|
| $\boldsymbol{v}_1 \in \mathbb{R}^3$ | Linear velocity of the {*V*} with respect to the {*I*} expressed in the {*V*} |
| $\boldsymbol{v}_2 \in \mathbb{R}^3$ | Angular velocity of the {*V*} with respect to the {*I*} expressed in the {*V*} |
| $\boldsymbol{p}_e \in \mathbb{R}^6$ | End-effector pose vector expressed in the {*I*} |
| $\boldsymbol{p}_{e1} \in \mathbb{R}^3$ | Position vector of the {*E*} expressed in the {*I*} |
| $\boldsymbol{p}_{e2} \in \mathbb{R}^3$ | Vector of Euler-angles of the {*E*} expressed in the {*I*} |
| $\boldsymbol{q}_m \in \mathbb{R}^{n_m}$ | Manipulator joint angular position state vector |
| $\boldsymbol{q} \in \mathbb{R}^{n_m+6}$ | Pose configuration vector of UVMS |
| $\boldsymbol{\zeta} \in \mathbb{R}^{n_m+6}$ | Body-fixed velocity vector of UVMS |
| $\boldsymbol{M} \in \mathbb{R}^{(n_m+6)\times(n_m+6)}$ | Inertia matrix of UVMS including added mass |
| $\boldsymbol{C} \in \mathbb{R}^{(n_m+6)\times(n_m+6)}$ | Matrix of Coriolis and centripetal terms of UVMS including added mass |
| $\boldsymbol{D} \in \mathbb{R}^{(n_m+6)\times(n_m+6)}$ | Matrix of dissipative effects of UVMS |
| $\boldsymbol{g} \in \mathbb{R}^{n_m+6}$ | Vector of gravity and buoyancy effects of UVMS |
| $\boldsymbol{\tau} \in \mathbb{R}^{n_m+6}$ | Vector of propulsion force/moment acting on the vehicle in {*V*} as well as joint torques |
| $\boldsymbol{\tau}_v \in \mathbb{R}^6$ | Vector of propulsion force/moment acting on the vehicle in {*V*} |
| $\boldsymbol{\tau}_m \in \mathbb{R}^{n_m}$ | Vector of manipulator joint torques |
| $\boldsymbol{J}_W \in \mathbb{R}^{6\times(n_m+6)}$ | UVMS Jacobian matrix |
| $\boldsymbol{h}_e \in \mathbb{R}^6$ | End-effector wrench vector expressed in the {*I*} |
| $\boldsymbol{f}_e \in \mathbb{R}^3$ | End-effector force vector expressed in the {*I*} |
| $\boldsymbol{\mu}_e \in \mathbb{R}^3$ | End-effector moment vector expressed in the {*I*} |
| $\boldsymbol{e}_h \in \mathbb{R}^6$ | End-effector unit wrench vector expressed in the {*I*} |
| $\boldsymbol{e}_f \in \mathbb{R}^3$ | End-effector unit force vector expressed in the {*I*} |
| $\boldsymbol{e}_\mu \in \mathbb{R}^3$ | End-effector unit moment vector expressed in the {*I*} |
| $\boldsymbol{\lambda} \in \mathbb{R}^2$ | Vector of end-effector force and moment magnitudes |
| $\lambda_f$ | Magnitude of end-effector force vector |
| $\lambda_\mu$ | Magnitude of end-effector moment vector |
| $\boldsymbol{u} \in \mathbb{R}^{p_V+n_m}$ | Vector of the control inputs of the UVMS actuators |
| $\boldsymbol{u}_v \in \mathbb{R}^{p_V}$ | Vector of the control inputs of the vehicle thrusters |
| $\boldsymbol{u}_m \in \mathbb{R}^{n_m}$ | Vector of the control inputs of the manipulator motors |
| $\boldsymbol{S}_v \in \mathbb{R}^{6\times p_V}$ | Thruster Configuration Matrix (TCM) of vehicle |
| $\boldsymbol{S} \in \mathbb{R}^{(n_m+6)\times(p_V+n_m)}$ | Thruster Configuration Matrix (TCM) of UVMS |

## 3.2 Elements of Nonlinear Programming

The problem we have defined so far can be characterized as a nonlinear constrained optimization problem due to the nonlinear nature of the constraints that need to be satisfied. In such problems, various issues need to be studied in order for the developed algorithms to work efficiently in real-time mode. In particular, the several types of constraints, such as those imposed by the physical environment as well as the hardware and geometric nature of the UVMS, can make the problem strongly constrained, complicated and computationally intense. Besides, even the selection of the objective function and the decision variables can play a role on the complexity of the problem and the computational effort during the optimization procedure. Nevertheless, the objective of this thesis is more fundamental; to design a control scheme that plays the role of a path planner in the joint space providing an optimal pose configuration of UVMS for efficient interaction with the environment wrt the pre-specified performance criterion. Therefore, optimization is treated as a useful tool and not as the main element of interest. However, for the sake of completeness, in this section basic elements of *Nonlinear Programming* are presented to provide some background on this field.

### 3.2.1 Nonlinear Programming Problem Definition

*Optimization* is the act of obtaining the best result under given circumstances. In design, construction, and maintenance of any engineering system, engineers have to take many technological and managerial decisions at several stages. The ultimate goal of all such decisions is either to minimize the effort required or to maximize the desired benefit. Since the effort required or the benefit desired in any practical situation can be expressed as a function of certain variables, a general optimization problem can be defined as the problem of finding the values of those variables that give the maximum or minimum value of a function. In order to proceed with the mathematical formulation of a general optimization problem, it seems important to present some significant definitions.

Design vector: Any engineering system or component is defined by a set of quantities some of which are viewed as variables during the design process. In general, certain quantities are usually fixed at the outset and these are called *preassigned parameters*. All the other quantities are treated as variables in the design process and are called *design* or *decision variables* $x_1, x_2, \ldots, x_{n_x}$. The design variables are collectively represented as a

*design vector* $\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_{n_x} \end{bmatrix}^T \in \mathbb{R}^{n_x}$. If an $n_x$-dimensional Cartesian space with each coordinate axis representing a design variable $x_1, x_2, \dots, x_{n_x}$ is considered, the space is called the *design variable space* or simply *design space*. Each point in the $n_x$ -dimensional design space is called a *design point* and represents either a possible or an impossible solution to the design problem.

Objective function: The conventional design procedures aim at finding an acceptable or adequate design that merely satisfies the functional and other requirements of the problem. In general, there will be more than one acceptable design, and the purpose of optimization is to choose the best one of the many acceptable designs available. Thus a criterion has to be chosen for comparing the different alternative acceptable designs and for selecting the best one. The criterion, with respect to which the design is optimized (minimized or maximized), when expressed as a function of the design variables $f(\boldsymbol{x})$, is known as the *criterion* or *merit* or *objective function*. The choice of objective function is governed by the nature of problem. The selection of the objective function can be one of the most important decisions in the whole optimum design process. In some situations, there may be more than one criteria to be satisfied simultaneously. An optimization problem involving multiple objective functions is known as a *multiobjective programming problem*. With multiple objectives there arises a possibility of conflict, and one simple way to handle the problem is to construct an overall objective function as a linear combination of the conflicting multiple objective functions.

Design constraints: In many practical problems, the design variables cannot be chosen arbitrarily; they rather have to satisfy certain specified functional and other requirements. The restrictions that must be satisfied to produce an acceptable design are collectively called *design constraints*. Constraints that represent limitations on the behavior or performance of the system are termed *behavior* or *functional constraints*. Constraints that represent physical limitations on design variables are known as *geometric* or *side constraints*. These constraints may be simple bounds, more general linear equality/inequality constraints or nonlinear equalities/inequalities that represent complex relationships among the variables.

The process of identifying objective function, decision variables, and constraints for a given problem is known as *modeling*. Construction of an appropriate model is the first step -sometimes the most important- in the optimization process. If the model is too simplistic, it will not give useful insights into the practical problem. If it is too complex, it may be too difficult to solve.

Optimization problems can be classified in several ways. An important distinction is between problems that have constraints on the variables and those that do not, namely *constrained* and *unconstrained problems* respectively. Another important classification of optimization problems is based on the nature of expressions for the objective function and the constraints. This classification is extremely useful from the computational point of view since there are many special methods available for the efficient solution of a particular class of problems. Thus the first task of a designer would be to investigate the class of problem encountered. This will, in many cases, dictate the types of solution procedures to be adopted in solving the problem. When the objective function and all the constraints are linear functions of the decision variables, the problem is a *linear programming problem*. If any of the functions among the objective and constraint functions is nonlinear, the problem is called a *nonlinear programming (NLP) problem*. This is the most general programming problem and all other problems can be considered as special cases of the NLP problem.

In accordance to what it has already been mentioned, the *general nonlinear programming problem* can be stated in minimization form as follows:

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathbb{R}^{n_x}}{\text{minimize}} \quad f(\boldsymbol{x}) \\ &\text{subject to: } h_i(\boldsymbol{x}) = 0\,, \qquad i = 1,2,\ldots,n_h \\ &\qquad\qquad\quad g_j(\boldsymbol{x}) \le 0\,, \qquad j = 1,2,\ldots,n_g \end{aligned} \tag{$P_1$}$$

where $\boldsymbol{x}$ is the $n_x$ -dimensional decision vector, $f(\boldsymbol{x})$ is the real-valued objective function, $h_i$ , $i = 1,2,\ldots,n_h$ are the equality constraints and $g_j$ , $j = 1,2,\ldots,n_g$ are the inequality constraints. For notational simplicity, the vector functions $\boldsymbol{h}: \mathbb{R}^{n_x} \to \mathbb{R}^{n_h}$ and $\boldsymbol{g}: \mathbb{R}^{n_x} \to \mathbb{R}^{n_g}$ are introduced, containing all the equality and inequality constraints respectively. If we define the feasible set $\boldsymbol{X} \subset \mathbb{R}^{n_x}$ to be the set of points $\boldsymbol{x}$ that satisfy all the constraints: that is,

$$\boldsymbol{X} = \{\, \boldsymbol{x} \mid \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}\ ;\ \boldsymbol{g}(\boldsymbol{x}) \le \boldsymbol{0} \,\} \tag{3.24}$$

problem ($P_1$) can be restated more compactly as:

$$\min_{\boldsymbol{x}\in\boldsymbol{X}} f(\boldsymbol{x}) \tag{$P_2$}$$

A vector $\boldsymbol{x} \in \boldsymbol{X}$ satisfying all the constraints is called a *feasible solution* to the problem; the collection of all such points forms the *feasible region*. By $\boldsymbol{x}^*$ a local optimal solution of the NLP, which minimizes[3] the objective function, is denoted.

### 3.2.2 Optimality Conditions for General NLP Problems

Based on the definition of the NLP problem of the previous subsection, the necessary and sufficient conditions for optimality are defined.

**Theorem 1 (First and Second-Order Necessary Conditions)**

First-Order Necessary Conditions

Let $f : \mathbb{R}^{n_x} \to \mathbb{R}$ , $\boldsymbol{h} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_h}$ and $\boldsymbol{g} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_g}$ be twice continuously differentiable functions on $\mathbb{R}^{n_x}$, i.e. $f, \boldsymbol{h}, \boldsymbol{g} \in C^2$. Let define the *Lagrangian* function $\mathcal{L} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_h} \times \mathbb{R}^{n_g} \to \mathbb{R}$ for problem ($P_1$) as:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) + \boldsymbol{\mu}^T \boldsymbol{g}(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x}) \tag{3.25}$$

Consider the problem ($P_1$) to minimize $f(\boldsymbol{x})$ subject to the constraints $\boldsymbol{g}(\boldsymbol{x}) \le \boldsymbol{0}$ and $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}$. If $\boldsymbol{x}^*$ is a local minimum of the ($P_1$) problem and a *regular*[4] point of the constraints, then there exist unique vectors (Lagrange Multipliers) $\boldsymbol{\mu}^* \in \mathbb{R}^{n_g}$ and $\boldsymbol{\lambda}^* \in \mathbb{R}^{n_h}$ such that:

$$\nabla_x \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \nabla_x f(\boldsymbol{x}^*) + \nabla_x \boldsymbol{g}(\boldsymbol{x}^*) \cdot \boldsymbol{\mu}^* + \nabla_x \boldsymbol{h}(\boldsymbol{x}^*) \cdot \boldsymbol{\lambda}^* = \boldsymbol{0} \tag{3.26a}$$

$$\boldsymbol{h}(\boldsymbol{x}^*) = \boldsymbol{0} \tag{3.26b}$$

$$\boldsymbol{g}(\boldsymbol{x}^*) \le \boldsymbol{0} \tag{3.26c}$$

$$\mu_j^* \, g_j(\boldsymbol{x}^*) = 0 \quad j = 1, 2, \ldots, n_g \tag{3.26d}$$

$$\boldsymbol{\mu}^* \ge \boldsymbol{0} \tag{3.26e}$$

[3] Throughout this section the formulation of the optimization problems presented is done in minimization form.

[4] If $J$ is the set of indices $j$ for which $g_j(\boldsymbol{x}^*) = 0$, a local solution $\boldsymbol{x}^*$ is a regular point of the constraints $\boldsymbol{h}(\boldsymbol{x}^*) = \boldsymbol{0}$ and $\boldsymbol{g}(\boldsymbol{x}^*) \le \boldsymbol{0}$ if the gradient vectors $\nabla h_i(\boldsymbol{x}^*)$ and $\nabla g_j(\boldsymbol{x}^*)$, with $i = 1, \ldots, n_h$ and $j \in J$, are linearly independent, that is, they satisfy the linear independent constraint qualification (LICQ).

Equations (3.26) are commonly referred to as the Karush-Kuhn-Tucker (KKT) first order necessary conditions. Conditions (3.26a–3.26c) are called the primal feasibility conditions. Condition (3.26d) is known as the complementary slackness condition. Condition (3.26e) requires the nonnegativity of the multipliers of the inequality constraints and is referred to as the dual feasibility condition.

Second-Order Necessary Condition

Let $J$ be the set of indices $j$ for which $g_j\left(\boldsymbol{x}^*\right)=0$, i.e. the set of *active* (or *binding*) inequality constraints at $\boldsymbol{x}^*$. Then a second-order condition for $\boldsymbol{x}^*$ to be a solution of ($P_1$) is:

$$\boldsymbol{y}^T\nabla^2_{xx}\mathcal{L}\left(\boldsymbol{x}^*,\boldsymbol{\lambda}^*,\boldsymbol{\mu}^*\right)\boldsymbol{y}\geq 0 \quad \Rightarrow$$

$$\boldsymbol{y}^T\left(\nabla^2_{xx}f\left(\boldsymbol{x}^*\right)+\nabla^2_{xx}\boldsymbol{g}\left(\boldsymbol{x}^*\right)^T\boldsymbol{\mu}^*+\nabla^2_{xx}\boldsymbol{h}\left(\boldsymbol{x}^*\right)^T\boldsymbol{\lambda}^*\right)\boldsymbol{y}\geq 0 \qquad (3.27)$$

for all $\boldsymbol{y}$ such that $\nabla_x g_j\left(\boldsymbol{x}^*\right)^T\boldsymbol{y}=0$, $j\in J$ and $\nabla_x\boldsymbol{h}\left(\boldsymbol{x}^*\right)^T\boldsymbol{y}=\boldsymbol{0}$.

**Theorem 2 (Second-Order Sufficient Conditions)**

Let $f:\mathbb{R}^{n_x}\to\mathbb{R}$, $\boldsymbol{h}:\mathbb{R}^{n_x}\to\mathbb{R}^{n_h}$ and $\boldsymbol{g}:\mathbb{R}^{n_x}\to\mathbb{R}^{n_g}$ be twice continuously differentiable functions on $\mathbb{R}^{n_x}$, i.e. $f,\boldsymbol{h},\boldsymbol{g}\in C^2$. Consider the problem ($P_1$) to minimize $f\left(\boldsymbol{x}\right)$ subject to the constraints $\boldsymbol{g}\left(\boldsymbol{x}\right)\leq\boldsymbol{0}$ and $\boldsymbol{h}\left(\boldsymbol{x}\right)=\boldsymbol{0}$. If there exist $\boldsymbol{x}^*,\boldsymbol{\lambda}^*$ and $\boldsymbol{\mu}^*$ satisfying the KKT conditions (3.26), and:

$$\boldsymbol{y}^T\nabla^2_{xx}\mathcal{L}\left(\boldsymbol{x}^*,\boldsymbol{\lambda}^*,\boldsymbol{\mu}^*\right)\boldsymbol{y}>0 \qquad (3.28)$$

for all $\boldsymbol{y}\neq\boldsymbol{0}$ such that

$$\nabla_x g_j\left(\boldsymbol{x}^*\right)^T\boldsymbol{y}=0,\ j\in J \quad \text{with} \quad \mu_j^*>0 \qquad (3.29)$$

$$\nabla_x g_j\left(\boldsymbol{x}^*\right)^T\boldsymbol{y}=0,\ j\in J \quad \text{with} \quad \mu_j^*=0 \qquad (3.30)$$

$$\nabla_x\boldsymbol{h}\left(\boldsymbol{x}^*\right)^T\boldsymbol{y}=\boldsymbol{0} \qquad (3.31)$$

then $\boldsymbol{x}^*$ is a strict local minimum of ($P_1$).

### 3.2.3 Optimization Methods

There is no single method available for solving all optimization problems efficiently. Hence a number of optimization methods have been developed for solving different types of optimization problems. The optimum seeking methods are also known as *mathematical programming techniques*. Several factors are to be considered in deciding a particular method. Some of them are:

- The type of problem to be solved (general nonlinear programming problem, geometric programming problem, etc.)
- The availability of a ready-made computer program
- The calendar time required for the development of a program
- The necessity of derivatives of the objective and constraint functions
- The available knowledge about the efficiency of the method
- The accuracy of the solution desired
- The programming language and quality of coding desired
- The robustness and dependability of the method in finding the true optimum solution
- The generality of the program for solving other problems
- The ease with which the program can be used and its output interpreted

Table 3.2 lists various mathematical programming techniques together with other well-defined areas of operations research. The classification given in Table 3.2 is not unique; it is given mainly for convenience.

**Table 3.2:** Optimization methods [50]

| Mathematical programming or optimization techniques | Stochastic process techniques | Statistical methods |
|---|---|---|
| Calculus methods | Statistical decision theory | Regression analysis |
| Calculus of variations | Markov processes | Cluster analysis, pattern recognition |
| Nonlinear programming | Queueing theory | |
| Geometric programming | Renewal theory | Design of experiments |
| Quadratic programming | Simulation methods | Discriminate analysis (factor analysis) |
| Linear programming | Reliability theory | |
| Dynamic programming | | |
| Integer programming | | |
| Stochastic programming | | |
| Separable programming | | |
| Multiobjective programming | | |
| Network methods: CPM and PERT | | |
| Game theory | | |
| *Modern or nontraditional optimization techniques* | | |
| Genetic algorithms | | |
| Simulated annealing | | |
| Ant colony optimization | | |
| Particle swarm optimization | | |
| Neural networks | | |
| Fuzzy optimization | | |

Mathematical programming techniques are useful in finding the minimum of a function of several variables under a prescribed set of constraints. Stochastic process techniques can be used to analyze problems described by a set of random variables having known probability distributions. Statistical methods enable one to analyze the experimental data and build empirical models to obtain the most accurate representation of the physical situation. A complete, thorough and up to date overview of the most significant works and algorithms can be found in [49, 50].

### 3.2.4 Optimization Software

The solution of most practical optimization problems requires the use of computers. Several commercial software systems [51-53] such as IMSL, ACM and MATLAB are available to solve optimization problems that arise in different engineering areas. The problem that is described and formulated in this thesis is solved using fmincon, a powerful routine for Constrained Nonlinear Optimization problems, developed by Mathworks [54] for the MATLAB Optimization Toolbox [55]. Fmincon includes a series of optimization algorithms, each of which can be more or less suitable depending on the type of the optimization problem that one has to solve.

Throughout this thesis, due to the nonlinear complicated nature of the grasping problems that we are studying, the "Active-Set" algorithm was chosen. Its solution procedure consists of two phases. The first phase involves the calculation of a feasible point. The second phase involves the generation of an iterative sequence of feasible points that converge to the solution. In this method an active set $\overline{A}_k$ is maintained that is an estimate of the active constraints at the solution point. $\overline{A}_k$ is updated at each iteration k, and this is used to form a basis for a search direction $d_k$. Equality constraints always remain in the active set. The search direction $d_k$ is calculated and minimizes the objective function while remaining on any active constraint boundaries. The feasible subspace for $d_k$ is formed from a basis $Z_k$ whose columns are orthogonal to the estimate of the active set $\overline{A}_k$ (i.e., $\overline{A}_k Z_k = 0$). Thus a search direction, which is formed from a linear summation of any combination of the columns of $Z_k$, is guaranteed to remain on the boundaries of the active constraints (more on fmincon, the active-set algorithm and the rest of them that it supports can be found in [55]).

## 3.3 Optimization Scheme

In this section, the mathematical formulation of the proposed optimization scheme is presented adopting the optimization form of constrained nonlinear programming problems stated in §3.2.1. The parameters, as well as the analytical expressions of the performance criterion and the constraints that will be used, have all been defined in §3.1. As already mentioned, the proposed optimization scheme plays the role of an on-line path planner in the joint space. It receives as input a sequence of end effector poses and produces a sequence of UVMS pose configurations which leads in a smooth way to the final optimal configuration of the combined vehicle–manipulator system that allows for "maximization" of the interaction wrench between the UVMS and the environment. The general idea of the proposed optimization scheme is illustrated in the following diagram (Fig. 3.1).

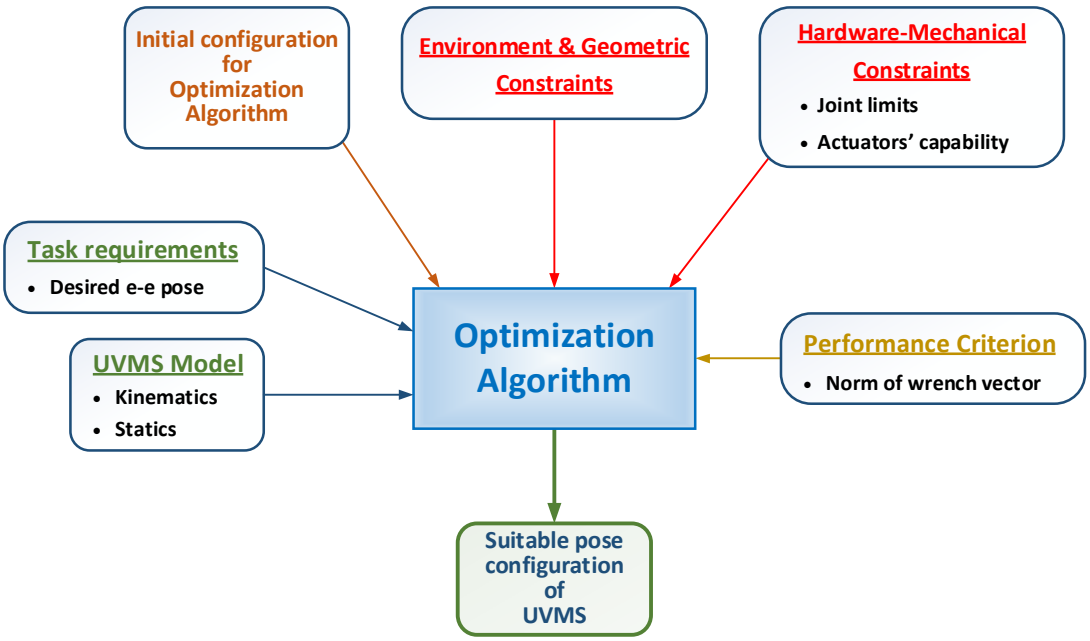



**Figure 3.1:** Diagram of the proposed optimization scheme.

Subsequently, we formulate the optimization problem described in previous sections for an UVMS with an *n*-link mounted manipulator. Let consider the elements of the UVMS pose configuration vector $\boldsymbol{q} \in \mathbb{R}^{n_m+6}$ and the elements of the vector of the end-effector force and moment magnitudes $\boldsymbol{\lambda} \in \mathbb{R}^2$ as decision variables. If we stack them in a vector, the corresponding design vector $\boldsymbol{x} \in \mathbb{R}^{n_m+8}$ of the problem is defined as follows:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\eta}_1^T & \boldsymbol{\eta}_2^T & \boldsymbol{q}_m^T & \boldsymbol{\lambda}^T \end{bmatrix}^T \in \mathbb{R}^{n_m+8} \tag{3.32}$$

Considering the appropriate objective function to ensure the maximization of the interaction wrench the end-effector could apply, we choose the 1-norm of vector $\boldsymbol{\lambda}$ as defined in Eq. 3.9. Thus, let denote the linear objective function we want to maximize by:

$$z = z(\boldsymbol{x}) = (1-w)\cdot|\lambda_f| + w\cdot|\lambda_\mu| \tag{3.33}$$

Since we want to follow the formulation of optimization problems in minimization form as defined in ($P_1$), we have to perform the conversion to this standardized formulation. Without loss of generality, we can accommodate this easily by minimizing $-z(\boldsymbol{x})$. For the sake of simplicity, during the remainder of the analysis the notation $z(\boldsymbol{x})$ will denote the negative of the function defined in Eq. 3.33 and consequently it will express the objective function of the problem formulated in minimization form.

Subsequently, the proposed optimization scheme is formulated as follows:

$$\underset{\boldsymbol{x}\in\mathbb{R}^{n_m+8}}{\text{minimize}} \quad z = -\left((1-w)\cdot|\lambda_f| + w\cdot|\lambda_\mu|\right) \tag{3.34}$$

$$\text{subject to:} \quad \boldsymbol{p}_{e1}(\boldsymbol{q}) = \boldsymbol{p}_{e1,d} \tag{3.35a}$$

$$\boldsymbol{p}_{e2}(\boldsymbol{q}) = \boldsymbol{p}_{e2,d} \tag{3.35b}$$

$$\boldsymbol{q}_{\min} \le \boldsymbol{q} \le \boldsymbol{q}_{\max} \tag{3.35c}$$

$$\boldsymbol{A}_{ineq}\cdot\boldsymbol{q} \le \boldsymbol{b}_{ineq} \tag{3.35d}$$

$$\boldsymbol{G}(\boldsymbol{q}) \le \boldsymbol{0} \tag{3.35e}$$

$$\boldsymbol{\tau}_{\min} \le \boldsymbol{g} + \boldsymbol{E}\cdot\boldsymbol{\lambda} \le \boldsymbol{\tau}_{\max} \tag{3.35f}$$

where, equalities and inequalities denoted by (3.35) are the constraints to be satisfied during the optimization procedure. Specifically, nonlinear equations Eq. 3.35a and Eq. 3.35b express the end-effector position and orientation constraints through the UVMS forward kinematics functions expressed by Eq. 3.10. The vectors $\boldsymbol{p}_{e1,d} \in \mathbb{R}^3$ and $\boldsymbol{p}_{e2,d} \in \mathbb{R}^3$ denote the desired end-effector position and orientation respectively, expressed in the inertial frame and produced by an on-line path planning algorithm. Linear inequality (3.35c) expresses the lower and upper bounds of the UVMS pose configuration vector due to mechanical and geometric constraints as explained in §3.1.1. Besides, linear and nonlinear inequalities (3.35d-3.35e) express the geometric

constraints ensuring that there are no collisions between various modules of the system. Finally, nonlinear inequalities (3.35f) represent the constraints regarding the limited capability range of the UVMS actuators as mentioned in §3.1.1.

## 3.4 Sensitivity Analysis Approach

As it has been mentioned, the proposed optimization scheme plays the role of an on-line motion planner in the joint space of the UVMS during the "reach-to-grasp" phase, providing in real-time a sequence of UVMS pose configurations that constitute the reference inputs for the low-level motion controller. The nonlinear constrained optimization problem, that is described and formulated, is solved iteratively using *fmincon*, a powerful nonlinear programming routine of MATLAB Optimization Toolbox. Considering that the nonlinear equality and inequality constraints to be respected make the problem complicated and the optimization procedure computationally intense, it is important to introduce an alternative approach of problem solution. This approach speeds up the computation procedure and seems to be more convenient for on-line motion planning schemes.

The key idea behind this approach of problem solution lies in applying sensitivity analysis in an iterative process to derive sequentially the optimal UVMS pose configurations during the "reach-to-grasp" phase. Specifically, we adopted the first order sensitivity analysis presented in [56]. The necessary background to deal with this problem is given in §3.4.1, where the method for local sensitivity analysis presented in the aforementioned article is introduced. In §3.4.2 the proposed motion planning scheme is presented.

### 3.4.1 Elements of Sensitivity Analysis

A typical constrained nonlinear optimization problem entails a group of physical quantities, which are used as design variables, and a group of constant quantities, termed parameters of the problem. Sensitivity analysis consists of determining "how" and "how much" specific changes in the parameters of the problem modify the optimal objective function value and the point where the optimum is attained. The problem of the sensitivity analysis in nonlinear programming has been discussed by several authors, for example [57-63].

In reference [56], the authors perform an integrated sensitivity analysis in which all the sensitivities (objective function value, primal and dual variables[5] values with respect to the parameters) are obtained at once, composed in closed and simple formulas. It should be noted that the compact formulas derived are only valid for the case of a specific widespread class of problems as it will described in the following analysis.

Consider the following constrained nonlinear optimization problem:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathbb{R}^{n_x}}{\text{minimize}} \quad & z = z\left(\boldsymbol{x},\boldsymbol{p}\right) \\ \text{subject to:} \quad & \boldsymbol{h}\left(\boldsymbol{x},\boldsymbol{p}\right)=\boldsymbol{0} \\ & \boldsymbol{g}\left(\boldsymbol{x},\boldsymbol{p}\right)\le\boldsymbol{0} \end{aligned} \qquad (P_3)$$

where $\boldsymbol{x}\in\mathbb{R}^{n_x}$ is the decision vector, $\boldsymbol{p}\in\mathbb{R}^{n_p}$ is the vector of problem parameters, $z:\mathbb{R}^{n_x}\times\mathbb{R}^{n_p}\to\mathbb{R}$ is the objective function while vector functions $\boldsymbol{h}:\mathbb{R}^{n_x}\times\mathbb{R}^{n_p}\to\mathbb{R}^{n_h}$ and $\boldsymbol{g}:\mathbb{R}^{n_x}\times\mathbb{R}^{n_p}\to\mathbb{R}^{n_g}$ express the equality and inequality constraints respectively, with $\boldsymbol{h}\left(\boldsymbol{x},\boldsymbol{p}\right)=\left[h_1\left(\boldsymbol{x},\boldsymbol{p}\right),\ \ldots\ ,h_{n_h}\left(\boldsymbol{x},\boldsymbol{p}\right)\right]^T$ and $\boldsymbol{g}\left(\boldsymbol{x},\boldsymbol{p}\right)=\left[g_1\left(\boldsymbol{x},\boldsymbol{p}\right),\ \ldots\ ,g_{n_g}\left(\boldsymbol{x},\boldsymbol{p}\right)\right]^T$. We assume that $n_h\le n_x$ and $z,\boldsymbol{h},\boldsymbol{g}\in C^2$.

Let $\boldsymbol{x}^*$ be a local optimal solution of problem ($P_3$) and assume that it is a regular non-degenerated[6] point of the constraints. As a result, the Karush-Kuhn-Tucker first order necessary conditions hold.

$$\nabla_{\boldsymbol{x}}\ z\left(\boldsymbol{x}^*,\boldsymbol{p}\right)+\sum_{i=1}^{n_h}\lambda_i^*\nabla_{\boldsymbol{x}}\ h_i\left(\boldsymbol{x}^*,\boldsymbol{p}\right)+\sum_{j=1}^{n_g}\mu_j^*\nabla_{\boldsymbol{x}}\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right)=\boldsymbol{0} \qquad (3.36a)$$

$$h_i\left(\boldsymbol{x}^*,\boldsymbol{p}\right)=0 \qquad i=1,2,...,n_h \qquad (3.36b)$$

$$g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\le 0 \qquad j=1,2,...,n_g \qquad (3.36c)$$

$$\mu_j^*\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right)=0 \qquad j=1,2,...,n_g \qquad (3.36d)$$

$$\mu_j^*\ \ge 0 \qquad j=1,2,...,n_g \qquad (3.36e)$$

[5] Primal variables correspond to decision variables, while dual variables correspond to Lagrange multipliers.

[6] A regular optimal point is denoted as non-degenerated if the Lagrange multipliers $\boldsymbol{\mu}^*$ of the active inequality constraints are different from zero.

To obtain the sensitivities of the optimal solution $\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, z^*$ with respect to the changes in the parameters $\boldsymbol{p}$, we perturb or modify $\boldsymbol{x}^*, \boldsymbol{p}, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, z^*$ in such a way that the KKT conditions still hold. Thus, to obtain the sensitivity equations we differentiate the objective function of ($P_3$) and the optimality conditions (3.36), as follows:

$$\left(\nabla_x\, z\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T d\boldsymbol{x} + \left(\nabla_p\, z\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T d\boldsymbol{p} - dz = 0 \tag{3.37}$$

$$\begin{aligned}&\left(\nabla_{xx}\, z\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{i=1}^{n_h} \lambda_i^* \nabla_{xx}\, h_i\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{j=1}^{m_J} \mu_j^* \nabla_{xx}\, g_j\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right) d\boldsymbol{x} \\ &+\left(\nabla_{xp}\, z\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{i=1}^{n_h} \lambda_i^* \nabla_{xp}\, h_i\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{j=1}^{m_J} \mu_j^* \nabla_{xp}\, g_j\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right) d\boldsymbol{p} \\ &+\nabla_x\, \boldsymbol{h}\left(\boldsymbol{x}^*, \boldsymbol{p}\right) d\boldsymbol{\lambda} + \nabla_x\, \boldsymbol{g}\left(\boldsymbol{x}^*, \boldsymbol{p}\right) d\boldsymbol{\mu} = \boldsymbol{0}\end{aligned} \tag{3.38}$$

$$\left(\nabla_x\, \boldsymbol{h}\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T d\boldsymbol{x} + \left(\nabla_p\, \boldsymbol{h}\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T d\boldsymbol{p} = \boldsymbol{0} \tag{3.39}$$

$$\left(\nabla_x\, g_j\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T d\boldsymbol{x} + \left(\nabla_p\, g_j\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T d\boldsymbol{p} = 0 \quad \text{if } \mu_j^* \neq 0,\ j \in J \tag{3.40}$$

where $J$ is the set of indices $j$ of binding inequality constraints, $m_J$ its cardinality, and all the matrices are evaluated at the optimal solution $\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, z^*$.

Note that the constraints (3.39-3.40) force the active constraints to remain active, while the differentiated complementary slackness condition (3.36d) is not present in the previous list since it holds due to the fact that we consider only non-degenerated cases (a detailed explanation can be seen in [63]). The uncommon degenerated case (binding inequality constraints with null multipliers) is analyzed in [63-65].

The linear system of (3.37-3.40) can be written in matrix form as follows:

$$\begin{bmatrix} \boldsymbol{F}_x & \boldsymbol{F}_p & \boldsymbol{0} & \boldsymbol{0} & -1 \\ \boldsymbol{F}_{xx} & \boldsymbol{F}_{xp} & \boldsymbol{H}_x^T & \boldsymbol{G}_x^T & \boldsymbol{0} \\ \boldsymbol{H}_x & \boldsymbol{H}_p & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{G}_x & \boldsymbol{G}_p & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \cdot \begin{bmatrix} d\boldsymbol{x} \\ d\boldsymbol{p} \\ d\boldsymbol{\lambda} \\ d\boldsymbol{\mu} \\ dz \end{bmatrix} = \boldsymbol{0} \tag{3.41}$$

where the submatrices are defined below (corresponding dimensions in parentheses):

$$\boldsymbol{F}_{\boldsymbol{x}\ (1\times n_x)} = \left(\nabla_{\boldsymbol{x}}\ z\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T \tag{3.42}$$

$$\boldsymbol{F}_{\boldsymbol{p}\ (1\times n_p)} = \left(\nabla_{\boldsymbol{p}}\ z\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T \tag{3.43}$$

$$\boldsymbol{F}_{\boldsymbol{xx}\ (n_x\times n_x)} = \nabla_{\boldsymbol{xx}}\ z\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{i=1}^{n_h} \lambda_i^* \nabla_{\boldsymbol{xx}}\ h_i\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{j=1}^{m_J} \mu_j^* \nabla_{\boldsymbol{xx}}\ g_j\left(\boldsymbol{x}^*, \boldsymbol{p}\right) \tag{3.44}$$

$$\boldsymbol{F}_{\boldsymbol{xp}\ (n_x\times n_p)} = \nabla_{\boldsymbol{xp}}\ z\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{i=1}^{n_h} \lambda_i^* \nabla_{\boldsymbol{xp}}\ h_i\left(\boldsymbol{x}^*, \boldsymbol{p}\right) + \sum_{j=1}^{m_J} \mu_j^* \nabla_{\boldsymbol{xp}}\ g_j\left(\boldsymbol{x}^*, \boldsymbol{p}\right) \tag{3.45}$$

$$\boldsymbol{H}_{\boldsymbol{x}\ (n_h\times n_x)} = \left(\nabla_{\boldsymbol{x}}\ \boldsymbol{h}\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T \tag{3.46}$$

$$\boldsymbol{H}_{\boldsymbol{p}\ (n_h\times n_p)} = \left(\nabla_{\boldsymbol{p}}\ \boldsymbol{h}\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T \tag{3.47}$$

$$\boldsymbol{G}_{\boldsymbol{x}\ (m_J\times n_x)} = \left(\nabla_{\boldsymbol{x}}\ \boldsymbol{g}\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T \tag{3.48}$$

$$\boldsymbol{G}_{\boldsymbol{p}\ (m_J\times n_p)} = \left(\nabla_{\boldsymbol{p}}\ \boldsymbol{g}\left(\boldsymbol{x}^*, \boldsymbol{p}\right)\right)^T \tag{3.49}$$

Vector (3.42) is the gradient of the objective function with respect to $\boldsymbol{x}$, vector (3.43) is the gradient on the objective function with respect to $\boldsymbol{p}$, submatrix (3.44) is the Hessian of the Lagrangian $\left(z\left(\boldsymbol{x}, \boldsymbol{p}\right) + \boldsymbol{\lambda}^T \cdot \boldsymbol{h}\left(\boldsymbol{x}, \boldsymbol{p}\right) + \boldsymbol{\mu}^T \cdot \boldsymbol{g}\left(\boldsymbol{x}, \boldsymbol{p}\right)\right)$ with respect to $\boldsymbol{x}$, submatrix (3.45) is the Hessian of the Lagrangian with respect to $\boldsymbol{x}$ and $\boldsymbol{p}$, submatrix (3.46) is the Jacobian of $\boldsymbol{h}\left(\boldsymbol{x}, \boldsymbol{p}\right)$ with respect to $\boldsymbol{x}$, submatrix (3.47) is the Jacobian of $\boldsymbol{h}\left(\boldsymbol{x}, \boldsymbol{p}\right)$ with respect to $\boldsymbol{p}$, submatrix (3.48) is the Jacobian of $\boldsymbol{g}\left(\boldsymbol{x}, \boldsymbol{p}\right)$ with respect to $\boldsymbol{x}$ for binding constraints, and submatrix (3.49) is the Jacobian of $\boldsymbol{g}\left(\boldsymbol{x}, \boldsymbol{p}\right)$ with respect to $\boldsymbol{p}$ for binding constraints.

To compute all sensitivities with respect to the components of the parameter vector $\boldsymbol{p}$, the Eq.3.41 can be written as:

$$\boldsymbol{U} \cdot \left[d\boldsymbol{x} \quad d\boldsymbol{\lambda} \quad d\boldsymbol{\mu} \quad dz\right]^T = \boldsymbol{V} \cdot d\boldsymbol{p} \tag{3.50}$$

where the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are given by:

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{F}_x & \boldsymbol{0} & \boldsymbol{0} & -1 \\ \boldsymbol{F}_{xx} & \boldsymbol{H}_x^T & \boldsymbol{G}_x^T & \boldsymbol{0} \\ \boldsymbol{H}_x & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{G}_x & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \tag{3.51}$$

$$\boldsymbol{V} = -\begin{bmatrix} \boldsymbol{F}_p \\ \boldsymbol{F}_{xp} \\ \boldsymbol{H}_p \\ \boldsymbol{G}_p \end{bmatrix} \tag{3.52}$$

If the optimal solution $\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, z^*$ is a non-degenerated regular point, then the matrix $\boldsymbol{U}$ is invertible, and consequently the solution of system (3.50) is unique and becomes:

$$\begin{bmatrix} d\boldsymbol{x} & d\boldsymbol{\lambda} & d\boldsymbol{\mu} & dz \end{bmatrix}^T = \begin{bmatrix} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{p}} & \frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{p}} & \frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{p}} & \frac{\partial z}{\partial \boldsymbol{p}} \end{bmatrix}^T d\boldsymbol{p} = \boldsymbol{U}^{-1}\boldsymbol{V} \cdot d\boldsymbol{p} \tag{3.53}$$

from which the matrix of all partial derivatives with respect to parameters results:

$$\boldsymbol{S} = \begin{bmatrix} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{p}} & \frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{p}} & \frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{p}} & \frac{\partial z}{\partial \boldsymbol{p}} \end{bmatrix}^T = \boldsymbol{U}^{-1}\boldsymbol{V} \tag{3.54}$$

Expression (3.54) allows one to derive sensitivities of the decision variables, the Lagrange multipliers and the objective function with respect to all parameters. Therefore, the expected change of the optimal state $\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, z^*$ after an infinitesimal perturbation $d\boldsymbol{p}$ of the parameters may be derived to a first order approximation, through the above sensitivity matrix.

Two particular cases are the following:

Case 1: $n_h + m_J = n_x$, i.e. when the number of active constraints (equalities plus inequalities) coincides with the number of decision variables, and the matrix $\boldsymbol{Q}_x = \begin{bmatrix} \boldsymbol{H}_x \\ \boldsymbol{G}_x \end{bmatrix}$ is invertible.

Case 2: $\boldsymbol{F}_{xx}$ is positive definite (invertible) and $\boldsymbol{Q}_x$ is full row rank (a typical assumption).

Note that in these two cases, the invertibility of $\boldsymbol{U}$ is guaranteed, and that formula (3.54) is valid for all cases in which $\boldsymbol{U}$ is invertible.

In the first case, if we denote $\boldsymbol{Q}_x = \begin{bmatrix} \boldsymbol{H}_x \\ \boldsymbol{G}_x \end{bmatrix}$, $\boldsymbol{Q}_p = \begin{bmatrix} \boldsymbol{H}_p \\ \boldsymbol{G}_p \end{bmatrix}$ and $\boldsymbol{n} = \begin{bmatrix} \lambda \\ \mu \end{bmatrix}$, matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ can be written as follows:

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{F}_x & \boldsymbol{0} & -1 \\ \boldsymbol{F}_{xx} & \boldsymbol{Q}_x^T & \boldsymbol{0} \\ \boldsymbol{Q}_x & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \tag{3.55}$$

$$\boldsymbol{V} = -\begin{bmatrix} \boldsymbol{F}_p \\ \boldsymbol{F}_{xp} \\ \boldsymbol{Q}_p \end{bmatrix} \tag{3.56}$$

and then, since $\boldsymbol{Q}_x^{-1}$ exists, $\boldsymbol{U}^{-1}$ also exists and can be written as:

$$\boldsymbol{U}^{-1} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{Q}_x^{-1} \\ \boldsymbol{0} & \left(\boldsymbol{Q}_x^T\right)^{-1} & -\left(\boldsymbol{Q}_x^T\right)^{-1} \boldsymbol{F}_{xx} \boldsymbol{Q}_x^{-1} \\ -1 & \boldsymbol{0} & \boldsymbol{F}_x \boldsymbol{Q}_x^{-1} \end{bmatrix} \tag{3.57}$$

leading to

$$\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{p}} = -\boldsymbol{Q}_x^{-1} \boldsymbol{Q}_p \tag{3.58}$$

$$\frac{\partial \boldsymbol{n}}{\partial \boldsymbol{p}} = \left(\boldsymbol{Q}_x^T\right)^{-1} \left(\boldsymbol{F}_{xx} \boldsymbol{Q}_x^{-1} \boldsymbol{Q}_p - \boldsymbol{F}_{xp}\right) \tag{3.59}$$

$$\frac{\partial z}{\partial \boldsymbol{p}} = \boldsymbol{F}_p - \boldsymbol{F}_x \boldsymbol{Q}_x^{-1} \boldsymbol{Q}_p \tag{3.60}$$

which are very neat formulas for the sensitivities.

In the second case where $\boldsymbol{F}_{xx}$ is invertible and $\boldsymbol{Q}_x$ is full row rank, then $\boldsymbol{B} = -\boldsymbol{Q}_x \boldsymbol{F}_{xx}^{-1} \boldsymbol{Q}_x^T$ is also invertible and we have:

$$\boldsymbol{U}^{-1} = \begin{bmatrix} \boldsymbol{0} & \left(\boldsymbol{I} + \boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1}\boldsymbol{Q}_x\right)\boldsymbol{F}_{xx}^{-1} & -\boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1} \\ \boldsymbol{0} & -\boldsymbol{B}^{-1}\boldsymbol{Q}_x\boldsymbol{F}_{xx}^{-1} & \boldsymbol{B}^{-1} \\ -1 & \boldsymbol{F}_x\left(\boldsymbol{I} + \boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1}\boldsymbol{Q}_x\right)\boldsymbol{F}_{xx}^{-1} & -\boldsymbol{F}_x\boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1} \end{bmatrix} \tag{3.61}$$

from which we get the alternative closed formulas:

$$\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{p}} = -\left(\boldsymbol{I} + \boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1}\boldsymbol{Q}_x\right)\boldsymbol{F}_{xx}^{-1}\boldsymbol{F}_{xp} + \boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1}\boldsymbol{Q}_p \tag{3.62}$$

$$\frac{\partial \boldsymbol{n}}{\partial \boldsymbol{p}} = \boldsymbol{B}^{-1}\boldsymbol{Q}_x\boldsymbol{F}_{xx}^{-1}\boldsymbol{F}_{xp} - \boldsymbol{B}^{-1}\boldsymbol{Q}_p \tag{3.63}$$

$$\frac{\partial z}{\partial \boldsymbol{p}} = \boldsymbol{F}_p - \boldsymbol{F}_x\left(\boldsymbol{I} + \boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1}\boldsymbol{Q}_x\right)\boldsymbol{F}_{xx}^{-1}\boldsymbol{F}_{xp} + \boldsymbol{F}_x\boldsymbol{F}_{xx}^{-1}\boldsymbol{Q}_x^T\boldsymbol{B}^{-1}\boldsymbol{Q}_p \tag{3.64}$$

### 3.4.2 Sensitivity Analysis in Optimization Problem

Towards addressing the fact that the procedure described in §3.3 demands high computational effort, in this subsection we will present an algorithm that incorporates the aforementioned methodology in a general on-line motion planning scheme for UVMS expected to interact with the environment.

As already mentioned, when the UVMS approaches the interaction spot, a path planning algorithm in the operational space is used to generate in real time a sequence of end-effector poses navigating it towards the target point. The proposed iterative algorithm receives as input the above sequence of end effector poses and produces a sequence of UVMS pose configurations that leads in a smooth way to the final optimal configuration for efficient interaction with the environment.

The initial optimal pose configuration is obtained via the nonlinear programming routine *fmincon*. The optimization problem to be solved has already been formulated (3.34-3.35). At this point, instead of solving iteratively the optimization problem using the *fmincon* routine and considering the new desired e-e poses, we apply the

aforementioned method of sensitivity analysis. Under the assumption that the optimal solution produced by the *fmincon* is a non-degenerate regular point, all the sensitivities wrt to the end-effector position and orientation are calculated. Afterwards, we compute the expected new optimal state via the corresponding sensitivities considering an infinitesimal perturbation of e-e desired pose as it is demanded by the task space planner. Then, this optimal state is used to calculate over again all the sensitivities and derive subsequently the next optimal state. Now, we must take into account that the proposed methodology has local validity and that the new optimal points, computed via the corresponding sensitivities, are derived to a first order approximation. Therefore, the sequential approximations will diverge from the actual optimal points over time. For this reason, we must set some conditions to be checked during the iterative procedure. For instance, the iterative process must terminate when an inactive constraint is activated or an approximated optimal state passes outside the feasible region. At this point, the *fmincon* routine is called again to derive the optimal UVMS pose configuration corresponding to the desired e-e pose. After that the iterative perturbation approach described above begins again. The proposed motion planning algorithm continues until the e-e is located at the target point ready to interact with the environment. A flowchart of this algorithm is depicted in Fig. 3.2.

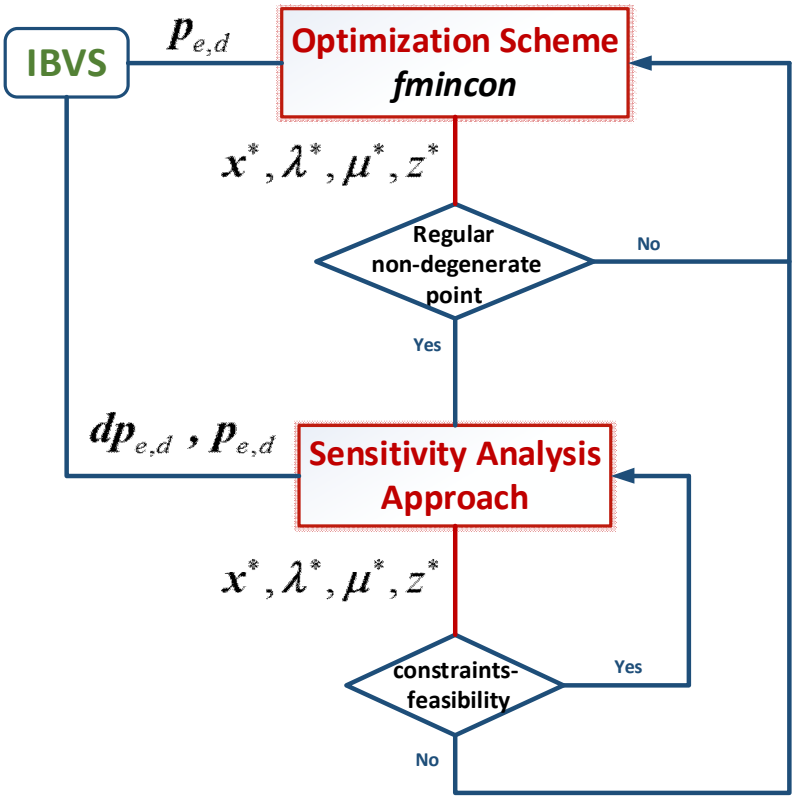



**Figure 3.2:** Flowchart of the proposed motion planning algorithm.

Next, we proceed to present the mathematical formulation of the proposed motion planning scheme. First of all, we must re-formulate the optimization problem defined in (3.34-3.35) in order to be described by the standardized formulation of problem ($P_3$). The design vector $\boldsymbol{x} \in \mathbb{R}^{n_m+8}$ is already defined by Eq.3.32. The parameter vector[7] of the problem is defined as follows:

$$\boldsymbol{p} = \left[ \boldsymbol{p}_{e1,d}^T \quad \boldsymbol{p}_{e2,d}^T \right]^T \in \mathbb{R}^6 \tag{3.65}$$

where the vectors $\boldsymbol{p}_{e1,d} \in \mathbb{R}^3$ and $\boldsymbol{p}_{e2,d} \in \mathbb{R}^3$ denote the desired end-effector position and orientation respectively, expressed in the inertial frame.

The objective function, as defined in Eq.3.33, depends on the design vector, while it does not depend on the parameters. All the equality constraints (3.35a-3.35b) as well as the inequality constraints (3.35c-3.35f) are stacked in the following systems of equations and inequalities:

$$\begin{aligned} \boldsymbol{h}(\boldsymbol{x},\boldsymbol{p}) &= \left[ h_1(\boldsymbol{x},\boldsymbol{p}),\ \ldots,h_{n_h}(\boldsymbol{x},\boldsymbol{p}) \right]^T = \boldsymbol{0} \\ \boldsymbol{g}(\boldsymbol{x},\boldsymbol{p}) &= \left[ g_1(\boldsymbol{x},\boldsymbol{p}),\ \ldots,g_{n_g}(\boldsymbol{x},\boldsymbol{p}) \right]^T \le \boldsymbol{0} \end{aligned} \tag{3.66}$$

Therefore, the optimization problem is defined as:

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathbb{R}^{n_m+8}}{\text{minimize}} \quad z(\boldsymbol{x}) \\ &\text{subject to:} \quad h_i(\boldsymbol{x},\boldsymbol{p}) = 0\ , \qquad i = 1,2,\ldots,n_h \\ &\qquad\qquad\quad\ g_j(\boldsymbol{x},\boldsymbol{p}) \le 0\ , \qquad j = 1,2,\ldots,n_g \end{aligned} \tag{$P_4$}$$

Differentiating the KKT conditions, we have:

$$\left( \nabla_x\, z(\boldsymbol{x}^*) \right)^T d\boldsymbol{x} - dz = 0 \tag{3.67}$$

[7] The term $\boldsymbol{p}$ is also used in Chapter 2. Henceforth, $\boldsymbol{p} \in \mathbb{R}^6$ will express the parameter vector.

$$\left(\sum_{i=1}^{n_h}\lambda_i^*\nabla_{xx}\ h_i\left(\boldsymbol{x}^*,\boldsymbol{p}\right)+\sum_{j=1}^{m_J}\mu_j^*\nabla_{xx}\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)d\boldsymbol{x}$$
$$+\left(\sum_{j=1}^{m_J}\mu_j^*\nabla_{xp}\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)d\boldsymbol{p}$$
$$+\nabla_{\boldsymbol{x}}\ \boldsymbol{h}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)d\boldsymbol{\lambda}+\nabla_{\boldsymbol{x}}\ \boldsymbol{g}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)d\boldsymbol{\mu}=\boldsymbol{0} \tag{3.68}$$

$$\left(\nabla_{\boldsymbol{x}}\ \boldsymbol{h}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T d\boldsymbol{x}+\left(\nabla_{\boldsymbol{p}}\ \boldsymbol{h}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T d\boldsymbol{p}=\boldsymbol{0} \tag{3.69}$$

$$\left(\nabla_{\boldsymbol{x}}\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T d\boldsymbol{x}+\left(\nabla_{\boldsymbol{p}}\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T d\boldsymbol{p}=0 \quad \text{if } \mu_j^*\neq 0\ ,\ j\in J \tag{3.70}$$

The linear system of (3.67-3.70) is written in matrix form as follows:

$$\begin{bmatrix}\boldsymbol{F}_{\boldsymbol{x}} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & -1\\ \boldsymbol{F}_{\boldsymbol{xx}} & \boldsymbol{F}_{\boldsymbol{xp}} & \boldsymbol{H}_{\boldsymbol{x}}^T & \boldsymbol{G}_{\boldsymbol{x}}^T & \boldsymbol{0}\\ \boldsymbol{H}_{\boldsymbol{x}} & \boldsymbol{H}_{\boldsymbol{p}} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{G}_{\boldsymbol{x}} & \boldsymbol{G}_{\boldsymbol{p}} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0}\end{bmatrix}\cdot\begin{bmatrix}d\boldsymbol{x}\\ d\boldsymbol{p}\\ d\boldsymbol{\lambda}\\ d\boldsymbol{\mu}\\ dz\end{bmatrix}=\boldsymbol{0} \tag{3.71}$$

where the submatrices are defined below (corresponding dimensions in parentheses):

$$\boldsymbol{F}_{\boldsymbol{x}\ (1\times 12)}=\left(\nabla_{\boldsymbol{x}}\ z\left(\boldsymbol{x}^*\right)\right)^T \tag{3.72}$$

$$\boldsymbol{F}_{\boldsymbol{xx}\ (12\times 12)}=\sum_{i=1}^{n_h}\lambda_i^*\nabla_{xx}\ h_i\left(\boldsymbol{x}^*,\boldsymbol{p}\right)+\sum_{j=1}^{m_J}\mu_j^*\nabla_{xx}\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right) \tag{3.73}$$

$$\boldsymbol{F}_{\boldsymbol{xp}\ (12\times 6)}=\sum_{j=1}^{m_J}\mu_j^*\nabla_{xp}\ g_j\left(\boldsymbol{x}^*,\boldsymbol{p}\right) \tag{3.74}$$

$$\boldsymbol{H}_{\boldsymbol{x}\ (n_h\times 12)}=\left(\nabla_{\boldsymbol{x}}\ \boldsymbol{h}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T \tag{3.75}$$

$$\boldsymbol{H}_{\boldsymbol{p}\ (n_h\times 6)}=\left(\nabla_{\boldsymbol{p}}\ \boldsymbol{h}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T \tag{3.76}$$

$$\boldsymbol{G}_{\boldsymbol{x}\ \left(m_J\times 12\right)}=\left(\nabla_{\boldsymbol{x}}\ \boldsymbol{g}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T \tag{3.77}$$

$$\boldsymbol{G}_{\boldsymbol{p}\ \left(m_J\times 6\right)}=\left(\nabla_{\boldsymbol{p}}\ \boldsymbol{g}\left(\boldsymbol{x}^*,\boldsymbol{p}\right)\right)^T \tag{3.78}$$

To compute all sensitivities with respect to the components of the parameter vector $\boldsymbol{p}$, the Eq. 3.71 can be written as:

$$\boldsymbol{U} \cdot \left[ d\boldsymbol{x} \quad d\boldsymbol{\lambda} \quad d\boldsymbol{\mu} \quad dz \right]^T = \boldsymbol{V} \cdot d\boldsymbol{p} \tag{3.79}$$

where the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are given by

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{F}_x & \boldsymbol{0} & \boldsymbol{0} & -1 \\ \boldsymbol{F}_{xx} & \boldsymbol{H}_x^T & \boldsymbol{G}_x^T & \boldsymbol{0} \\ \boldsymbol{H}_x & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{G}_x & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \tag{3.80}$$

$$\boldsymbol{V} = -\begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{F}_{xp} \\ \boldsymbol{H}_p \\ \boldsymbol{G}_p \end{bmatrix} \tag{3.81}$$

Then, if the optimal solution $\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, z^*$ is a non-degenerated regular point, the sensitivity matrix with respect to parameters is derived as follows:

$$\boldsymbol{S} = \begin{bmatrix} \dfrac{\partial \boldsymbol{x}}{\partial \boldsymbol{p}} & \dfrac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{p}} & \dfrac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{p}} & \dfrac{\partial z}{\partial \boldsymbol{p}} \end{bmatrix}^T = \boldsymbol{U}^{-1}\boldsymbol{V} \tag{3.82}$$

# Chapter 4

# Simulations

In this Chapter, the efficacy of the developed motion planning algorithms and optimization scheme as part of the overall interaction control scheme will be demonstrated through a series of simulation studies in MATLAB, where various underwater scenarios will be considered. The autonomous UVMS used for the simulation studies is presented in §4.1, while the simulation results of the proposed algorithms are provided in §4.2 and §4.3.

## 4.1 Simulation Model

As already mentioned, this work has been inspired by the PANDORA research project. The autonomous UVMS, the motion planning algorithms are designed for, is composed of the Girona500 AUV [66-67] and the ARM 5E Micro manipulator [68]. The vehicle-manipulator system is depicted in Fig. 4.1. As it is obvious the manipulator arm is mounted on the front area of the lower torpedo-shaped hull.

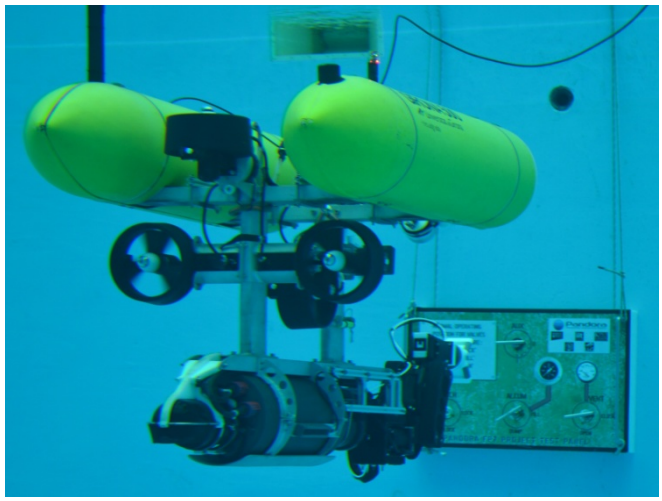


**Figure 4.1:** The GIRONA-UVMS composed of Girona500 AUV and ARM 5E Micro manipulator.

## GIRONA 500 AUV

The Girona500 is a compact, lightweight and hovering capable AUV developed at the Underwater Robotics Lab of the University of Girona (Fig. 4.2). The main characteristic of the vehicle is that it can be reconfigured for different tasks, ranging from the classical sonar and video imaging surveys to the challenging autonomous intervention tasks, by equipping mission-specific payloads, reconfiguring the propulsion system and adjusting the vehicle buoyancy.

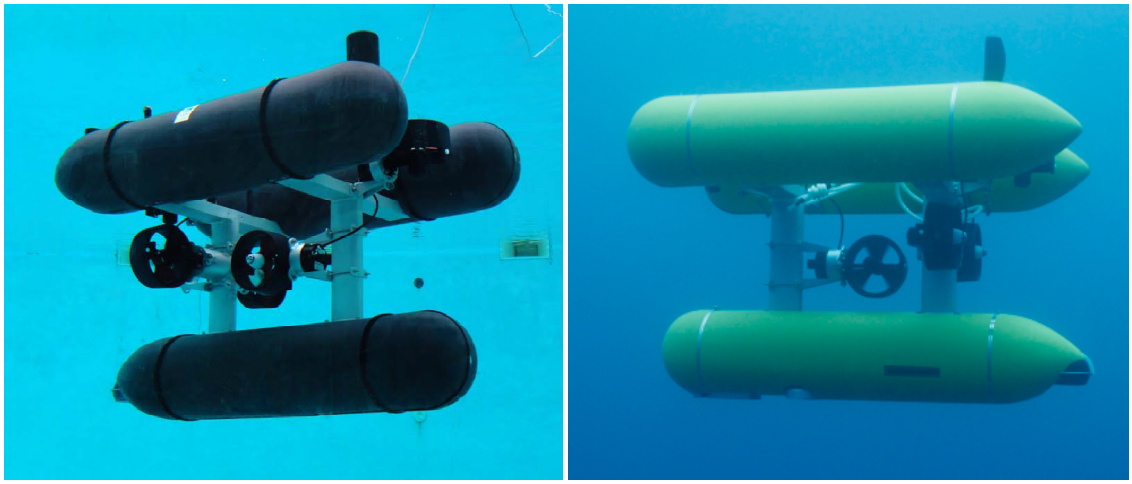


**Figure 4.2:** The Girona500 AUV at the water tank (left) and the sea (right).

The Girona500 is designed for a maximum operating depth of up to 500m. The vehicle is composed of an aluminum frame which supports three torpedo-shaped hulls of 0.3m in diameter and 1.5m in length as well as other elements like the thrusters. The overall dimensions of the vehicle are 1m in height, 1m in width, 1.5m in length and a weight of less than 200 kg. This design offers a good hydrodynamic performance and a large space for housing the equipments while maintaining a compact size which allows operating the vehicle from small boats. The two upper hulls, which contain the flotation foam and the electronics housing, are positively buoyant, while the lower one contains the more heavy elements such as the batteries and the payload. This particular arrangement of the components separates the centre of gravity from the centre of buoyancy by about 11 cm, which is significantly more than found in a typical torpedo shape design. This provides the vehicle with passive stability in pitch and roll, making it suitable for tasks that will benefit from a steady platform such as interventions or imaging surveys.

In its standard configuration, the vehicle is equipped with typical navigation sensors (DVL, AHRS, pressure gauge and USBL) and basic survey equipment (profiler sonar, side scan sonar, video camera and sound velocity sensor). In addition to these sensors, almost half the volume of the lower hull is reserved for payload equipment that can be configured according to the requirements of a particular mission (Fig. 4.3).

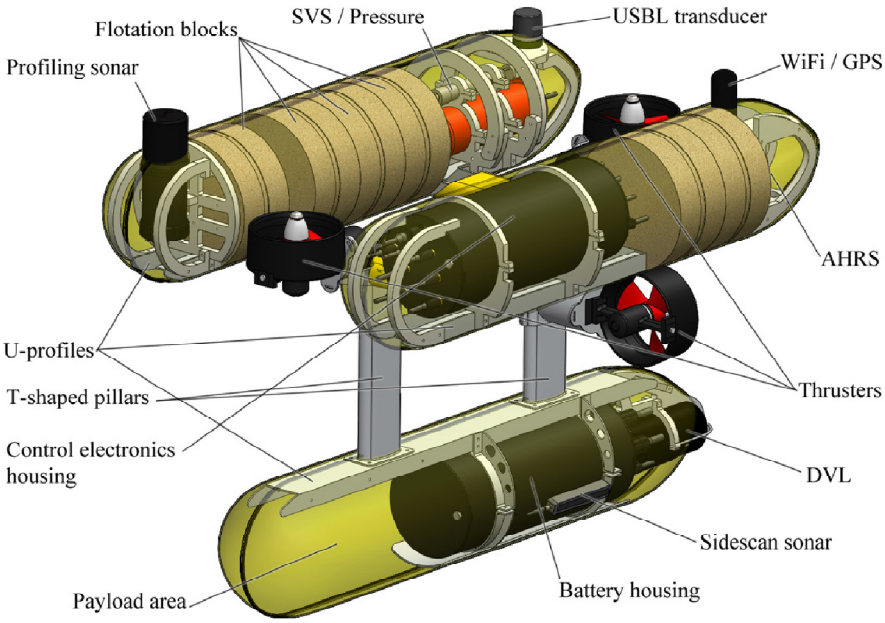



**Figure 4.3:** The Girona 500 AUV internals.

As already mentioned, unlike other similar vehicles, the Girona500 also has the capacity to modify its propulsion system from the redundant vectored thrust typical of intervention ROVs, to more lightweight and efficient arrangements, preferred for long endurance tasks. On its minimal set-up, the Girona500 is equipped with three thrusters, two to actuate the surge and yaw and one to actuate the heave (Fig. 4.4a). The basic layout has four thrusters (Fig. 4.3), two vertical to actuate the heave and pitch and two horizontal for the yaw and surge. In the presence of currents, or when the task at hand demands the capacity of executing lateral movements, there is the possibility to mount bow and stern thrusters (Fig. 4.4c). The lateral motion can also be achieved by installing a single thruster in the middle of the two pillars, although at the cost of losing the redundancy (Fig. 4.4b). It is possible to reconfigure the vehicle to operate with up to eight thrusters to achieve a fully actuated vehicle (Fig. 4.4d). This configuration is only employed in tasks, such as a free-floating manipulation, where a high lifting thrust or a precise control is required. Finally, more flotation modules can be incorporated to adjust the buoyancy with each particular configuration.

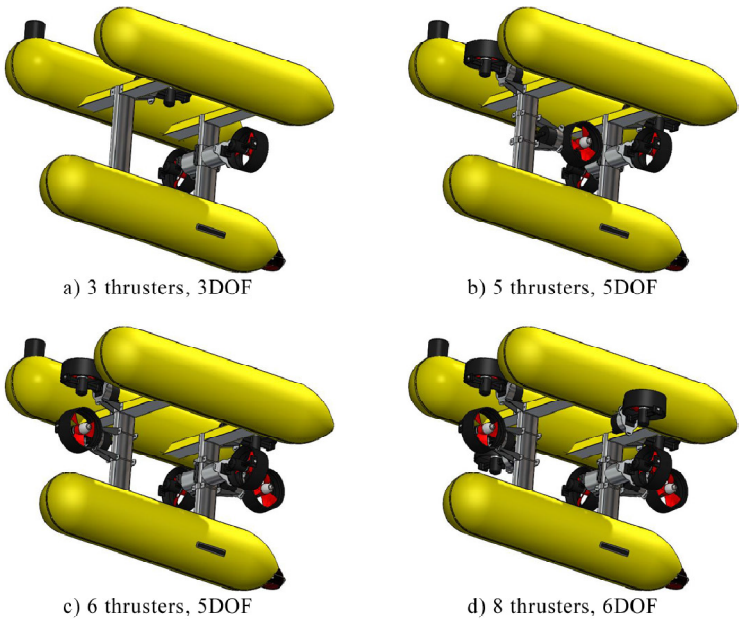



**Figure 4.4:** Some thruster configurations for the Girona500 propulsion system.

## ARM 5E Micro

The ARM 5E Micro is a robotic manipulator arm composed of four revolute joints (Fig. 4.5). It is actuated by 24-30V brushless DC motors, while an actuated robot gripper allows for grasping small objects. The total weight in the air is about 10 kg, whereas in fresh water it decreases to 2.75 kg approximately. The arm is capable of lifting 10 kg at full reach, and can descend up to 300m in water.

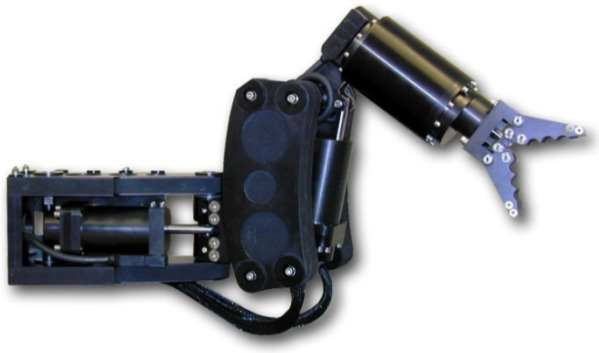


**Figure 4.5:** The ARM 5E Micro.

Adopting the Denavit-Hartenberg notation, the reference frames at the joints are attached as depicted in Fig. 4.6.

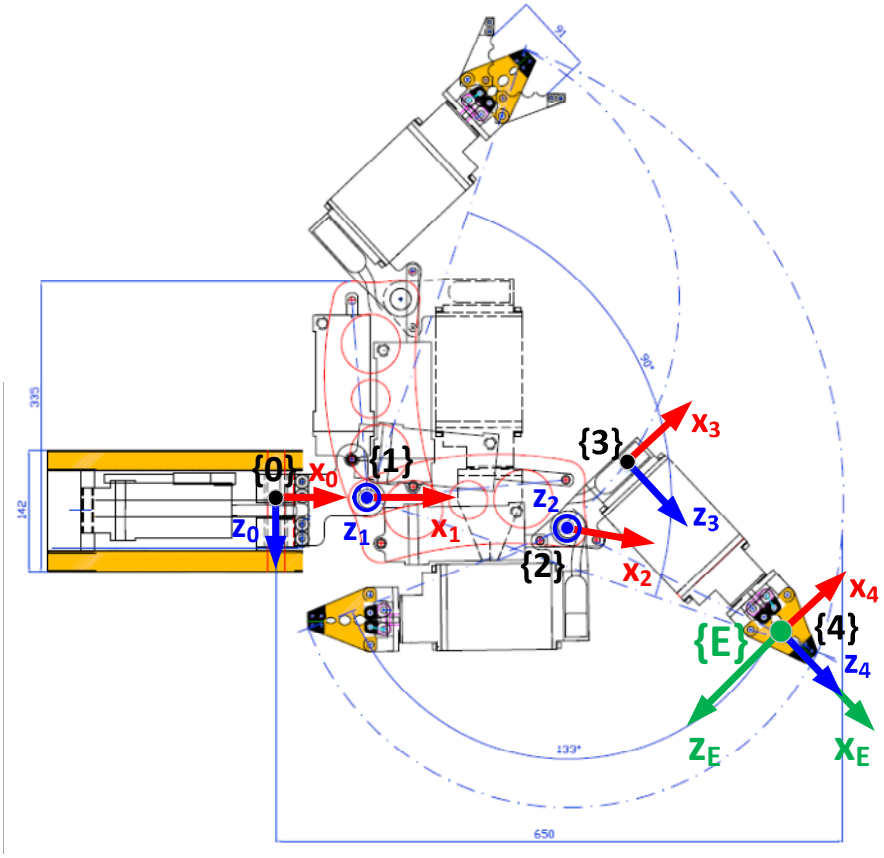



**Figure 4.6:** Schematic diagram of ARM 5E Micro with reference frames attached.

Consequently, the following D-H parameters (Table 4.1) for the manipulator arm are derived. We can notice that frame {4} does not coincide with the end-effector frame {*E*} and a proper rotation is needed. Regarding the mechanical limitations of the joints, they are presented in Table 4.2.

**Table 4.1:** D-H parameters of ARM 5E Micro

| **D-H** | | | | |
|---|---|---|---|---|
| | **$d_i$ (m)** | **$q_i$** | **$a_i$ (m)** | **alpha$_i$ (rad)** |
| **1** | 0 | $q_1$ | 0,1 | -pi/2 |
| **2** | 0 | $q_2$ | 0,26 | 0 |
| **3** | 0 | $q_3$ | 0,09 | pi/2 |
| **4** | 0,29 | $q_4$ | 0 | 0 |
| **E** | $Rot_y$(-pi/2) | | | |

**Table 4.2:** Joint limits of ARM 5E Micro

| Lower Limit (rad) | Joint | Upper Limit (rad) |
|---|---|---|
| -0,52 | slew - $q_1$ | 1,46 |
| -0,1471 | elevation - $q_2$ | 1,314 |
| -1,297 | elbow - $q_3$ | 0,73 |
| -3,14 | jaw rotate - $q_4$ | 3,14 |

**GIRONA–UVMS**

The origin of the vehicle-fixed frame {*V*} is chosen to coincide with the vehicle's center of gravity ($CG_V$), i.e. ${}^V\boldsymbol{r}_{Gv} = {}^V\begin{bmatrix} x_{Gv} & y_{Gv} & z_{Gv} \end{bmatrix}^T = \mathbf{0}_{3\times1}$, while the position vector of the vehicle's center of buoyancy ($CB_V$) with respect to {*V*} is ${}^V\boldsymbol{r}_{Bv} = {}^V\begin{bmatrix} x_{Bv} & y_{Bv} & z_{Bv} \end{bmatrix}^T =$ $= \begin{bmatrix} 0 & 0 & -0.11 \end{bmatrix}^T m$.

We assume that the center of gravity of each link ($CG_i$) coincides with the corresponding center of buoyancy ($CB_i$). We also assume that they are located at the geometrical center of the link. Thus, it turns out:

$$\lambda_{Gi} = \lambda_{Bi} = \frac{1}{2} \Rightarrow \begin{cases} \boldsymbol{p}_{Gi} = \boldsymbol{p}_{Bi} \\ {}^V\boldsymbol{r}_{Gi} = {}^V\boldsymbol{r}_{Bi} \end{cases}, i = 1,\ldots,4 \qquad (4.1)$$

The constant transformation matrix describing the position and orientation of manipulator base frame {0} wrt to the vehicle-fixed frame {*V*} is:

$${}^V\boldsymbol{T}_0 = \begin{bmatrix} 1 & 0 & 0 & 0.75 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0.45 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (4.2)$$

On the basis of the D-H parameters, we compute the homogeneous transformation matrices ${}^{i-1}\boldsymbol{T}_i(q_i)$, $i = 1,\ldots,4$, as follows:

$$ {}^{i-1}\boldsymbol{T}_i = \begin{bmatrix} cq_i & -sq_i c\alpha_i & sq_i s\alpha_i & \mathrm{a}_i \cdot cq_i \\ sq_i & cq_i c\alpha_i & -cq_i s\alpha_i & \mathrm{a}_i \cdot sq_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.3} $$

The propulsion system of Girona500 AUV is reconfigurable. Here, it is considered that the vehicle is equipped with five thrusters (Fig. 4.7), two vertical to actuate the heave and pitch, two horizontal for the yaw and surge and a single thruster in the middle of the two pillars for the sway motion. So, with this configuration a 5-dof actuation is achieved with roll-dof non-actuated.

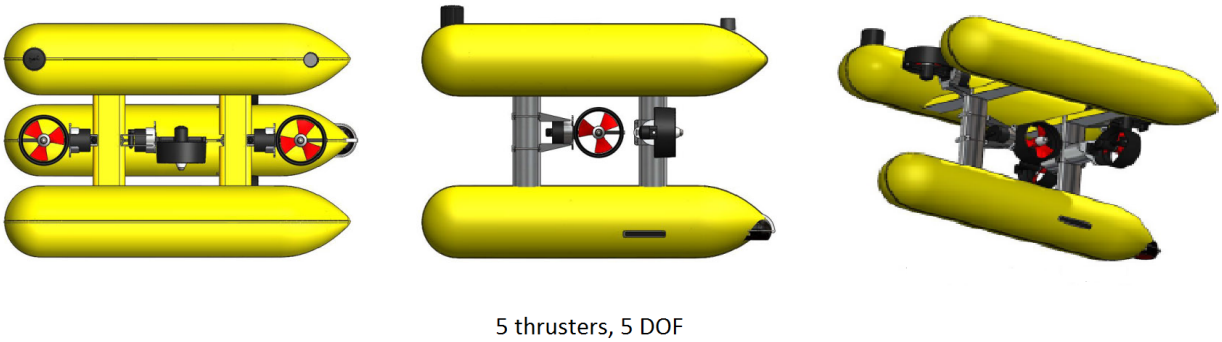



**Figure 4.7:** 5-thruster configuration for the Girona500 propulsion system.

Let us denote the vector of the control inputs of the UVMS actuators by:

$$ \boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_v{}^T & \boldsymbol{u}_m{}^T \end{bmatrix}^T \in \mathbb{R}^9 \quad \begin{matrix} \nearrow \ \boldsymbol{u}_v \in \mathbb{R}^5 \\ \searrow \ \boldsymbol{u}_m \in \mathbb{R}^4 \end{matrix} $$

The relationship between the force/moment acting on the vehicle $\boldsymbol{\tau}_v$ and the control input of the thrusters $\boldsymbol{u}_v \in \mathbb{R}^5$ is highly nonlinear. A common simplification is to consider a linear relationship between $\boldsymbol{\tau}_v$ and $\boldsymbol{u}_v$:

$$ \boldsymbol{\tau}_v = \boldsymbol{S}_v \cdot \boldsymbol{u}_v \tag{4.4} $$

where $\boldsymbol{S}_v \in \mathbb{R}^{6\times5}$ is the following Thruster Configuration Matrix (TCM) of the vehicle. As it is obvious the roll DOF is non-actuated.

$$\boldsymbol{S}_v = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & dv & -dv & 0 \\ -dh & dh & 0 & 0 & 0 \end{bmatrix} \tag{4.5}$$

where $dv = 0.559m$ and $dh = 0.259m$ are the perpendicular distances from the center of the vehicle to the axes of the two vertical thrusters and the two horizontal thrusters, respectively. It can also be recognized that not all the directions are independently actuated.

The relationship between the generalized forces $\boldsymbol{\tau} \in \mathbb{R}^{10}$ and the control input $\boldsymbol{u} \in \mathbb{R}^{9}$ is described by the following Thruster Configuration Matrix, $\boldsymbol{S} \in \mathbb{R}^{10\times 9}$, for the whole UVMS :

$$\boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{\tau}_v \\ \boldsymbol{\tau}_m \end{bmatrix} = \begin{bmatrix} \boldsymbol{S}_v & \boldsymbol{0}_{6\times 4} \\ \boldsymbol{0}_{4\times 5} & \boldsymbol{I}_4 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{u}_v \\ \boldsymbol{u}_m \end{bmatrix} \Rightarrow$$

$$\boldsymbol{\tau} = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & dv & -dv & 0 & 0 & 0 & 0 & 0 \\ -dh & dh & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{u}_v \\ \boldsymbol{u}_m \end{bmatrix} = \boldsymbol{S} \cdot \boldsymbol{u} \tag{4.6}$$

Notice that, while for the AUV $p_v = 5$ control inputs are assumed, for the manipulator it is supposed that $n_m = 4$ joint motors are available.

Considering that the five thrusters are the same, the control input vector $\boldsymbol{u} \in \mathbb{R}^{9}$ has the following limits:

$$\boldsymbol{u}_{\min} \le \boldsymbol{u} \le \boldsymbol{u}_{\max} \quad \Rightarrow \quad \begin{bmatrix} \boldsymbol{u}_{\min\_th} \\ \boldsymbol{u}_{\min\_m} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{u}_{v} \\ \boldsymbol{u}_{m} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{u}_{\max\_th} \\ \boldsymbol{u}_{\max\_m} \end{bmatrix} \quad \Rightarrow$$

$$\begin{bmatrix} u_{\min\_th} \\ u_{\min\_th} \\ u_{\min\_th} \\ u_{\min\_th} \\ u_{\min\_th} \\ u_{\min\_m\,1} \\ u_{\min\_m\,2} \\ u_{\min\_m\,3} \\ u_{\min\_m\,4} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{u}_{v} \\ \boldsymbol{u}_{m} \end{bmatrix} \le \begin{bmatrix} u_{\max\_th} \\ u_{\max\_th} \\ u_{\max\_th} \\ u_{\max\_th} \\ u_{\max\_th} \\ u_{\max\_m\,1} \\ u_{\max\_m\,2} \\ u_{\max\_m\,3} \\ u_{\max\_m\,4} \end{bmatrix} \tag{4.7}$$

Thus, it turns out that the limits for the generalized forces $\tau \in \mathbb{R}^{10}$ are given by:

$$\boldsymbol{\tau}_{\min} \le \boldsymbol{\tau} \le \boldsymbol{\tau}_{\max} \quad \Rightarrow \quad \begin{bmatrix} \boldsymbol{\tau}_{\min\_v} \\ \boldsymbol{\tau}_{\min\_m} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{\tau}_{v} \\ \boldsymbol{\tau}_{m} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{\tau}_{\max\_v} \\ \boldsymbol{\tau}_{\max\_m} \end{bmatrix} \quad \Rightarrow$$

$$\begin{bmatrix} -2u_{\max\_th} \\ u_{\min\_th} \\ -2u_{\max\_th} \\ 0 \\ dv \cdot \left( u_{\min\_th} - u_{\max\_th} \right) \\ dh \cdot \left( u_{\min\_th} - u_{\max\_th} \right) \\ u_{\min\_m\,1} \\ u_{\min\_m\,2} \\ u_{\min\_m\,3} \\ u_{\min\_m\,4} \end{bmatrix} \le \begin{bmatrix} \boldsymbol{\tau}_{v} \\ \boldsymbol{\tau}_{m} \end{bmatrix} = \boldsymbol{S} \cdot \begin{bmatrix} \boldsymbol{u}_{v} \\ \boldsymbol{u}_{m} \end{bmatrix} \le \begin{bmatrix} -2u_{\min\_th} \\ u_{\max\_th} \\ -2u_{\min\_th} \\ 0 \\ dv \cdot \left( u_{\max\_th} - u_{\min\_th} \right) \\ dh \cdot \left( u_{\max\_th} - u_{\min\_th} \right) \\ u_{\max\_m\,1} \\ u_{\max\_m\,2} \\ u_{\max\_m\,3} \\ u_{\max\_m\,4} \end{bmatrix} \tag{4.8}$$

## 4.2 Simulation Results: 1st Approach of Solution

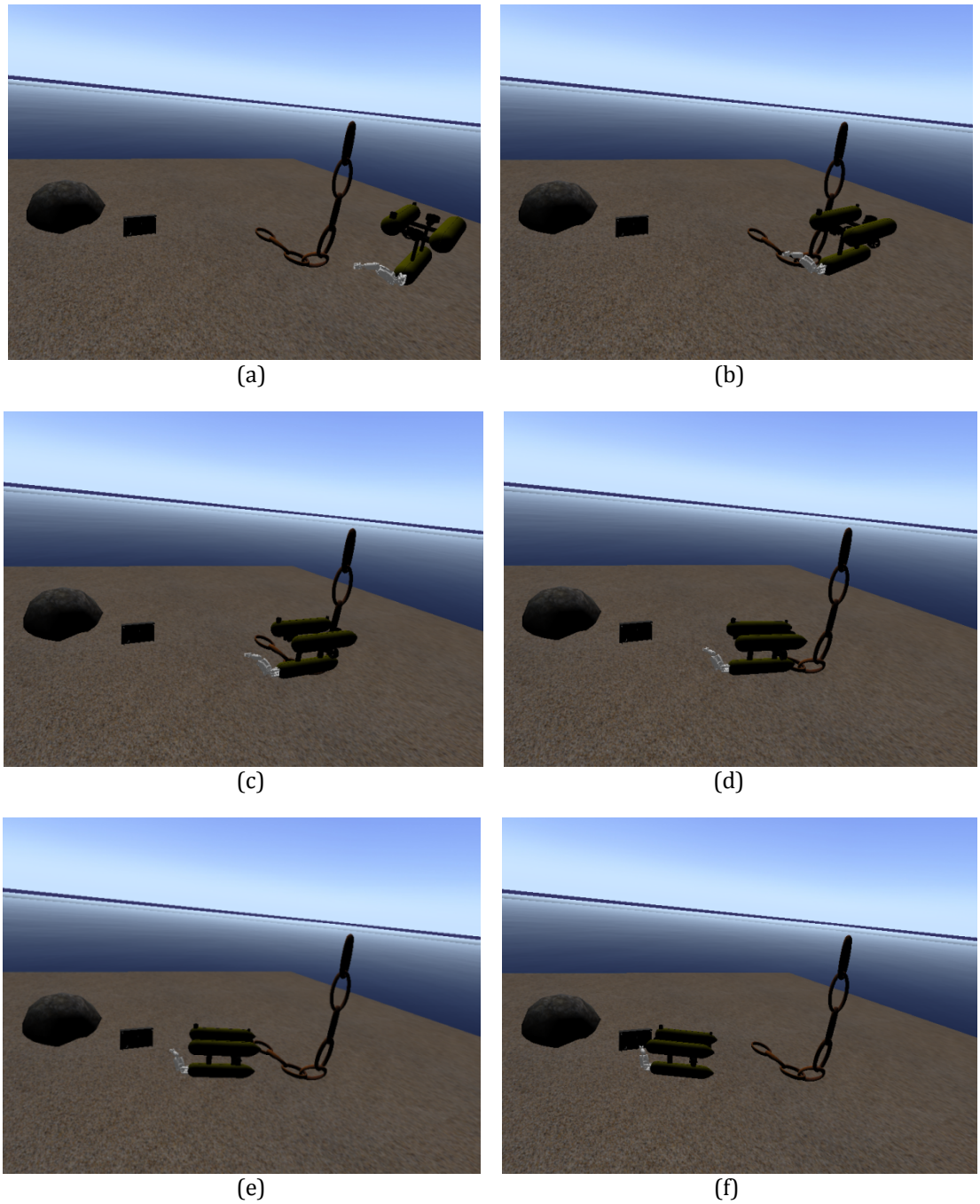
In this subsection, the efficacy of the first proposed motion planning scheme, as it was presented in §3.3, will be demonstrated through a series of simulations in MATLAB. Three case studies are considered as will be presented below. An image-based controller is used as part of the overall motion planning scheme. The image-based controller's goal is to locate the correct valve head of a subsea valve panel, identify the state of the valve and produce a sequence of end-effector poses that leads the gripper to the target point with appropriate orientation.

This sequence of desired e-e poses constitutes the input of the optimization scheme. The nonlinear constrained optimization problem is solved iteratively using *fmincon*. Due to the nonlinear complicated nature of the constraints, the *active-set* algorithm is chosen to solve the problem. It is important to mention that the optimal UVMS pose configuration corresponding to each desired e-e pose, constitutes the initial estimation for the next optimization process. Moreover, the gradients of the objective function and the nonlinear constraints have been analytically calculated and are supplied to the *active-set* algorithm in order to speed up the computation procedure. The adopting UVMS model used for the simulations has been described in §4.1.

### 4.2.1 Case Study 1

In this first test, the initial desired end-effector pose wrt to the inertial frame {*I*}, when the UVMS enters the "*reach-to-grasp*" phase, is assumed to be $\boldsymbol{p}_{e,d\_i} = \begin{bmatrix} -4.3 & 2.5 & -1.1 & 0 & -pi/8 & -pi/5 \end{bmatrix}^T \text{m, rad}$. We also assume that the final desired e-e pose, i.e. the location of the correct valve head, coincides with {*I*}. Apart from the constraints to be satisfied, as they were defined in §3.3, we also demand that the vehicle's roll and pitch angles wrt the inertial frame are kept to zero. Furthermore, we consider that we want to maximize both the force and moment that the end-effector could apply. Specifically, we want to maximize the magnitude of force along the x-axis of the e-e frame as well as the magnitude of moment about the same axis. So, we have that $\boldsymbol{e}_f = {}^I\boldsymbol{R}_E \cdot \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$ and $\boldsymbol{e}_\mu = {}^I\boldsymbol{R}_E \cdot \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$. Finally, we select the weighting factor of the objective function to be $w = 0.9$. It must be noticed that the desired path, as it is produced by the image based controller, consists of 100 desired end-effector poses.

A depiction of the generated joint space path appears in Fig. 4.8 where the UVMS navigates towards the correct valve head. For illustrative purposes, the sequence of UVMS optimal postures, as it has been generated by MATLAB, is provided in the UWSim simulator [69] where the snapshots of Fig. 4.8 are taken from. Fig. 4.8a illustrates the initial optimal UVMS pose configuration, while Fig.4.8b–Fig.4.8e show the optimal UVMS pose configurations each 20 iterations of the path planning scheme. Finally, Fig. 4.8f and Fig. 4.9 depict the final UVMS pose configuration when the end-effector is ready to grasp the valve head and apply the required interaction wrench.



(a) (b) (c) (d) (e) (f)

**Figure 4.8:** 1st Solution Process - Case Study 1: UVMS optimal pose configurations during the reach-to-grasp phase.

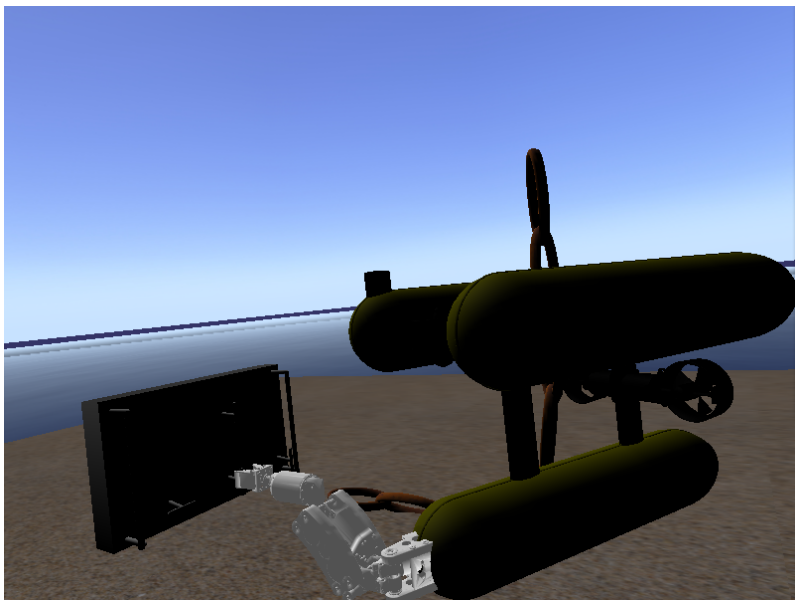


**Figure 4.9:** 1st Solution Process - Case Study 1: UVMS final optimal pose configuration grasping the valve head.

In Fig. 4.10 the generated sequence of the vehicle's position and orientation in terms of Euler angles, both expressed in the inertial frame, is given. It can be observed that the vehicle's roll and pitch angles are kept to zero as it has been demanded. Fig. 4.11 reports the generated sequence of manipulator's joint angular position states. All the joint states satisfy the angular limitations of the joint motors.

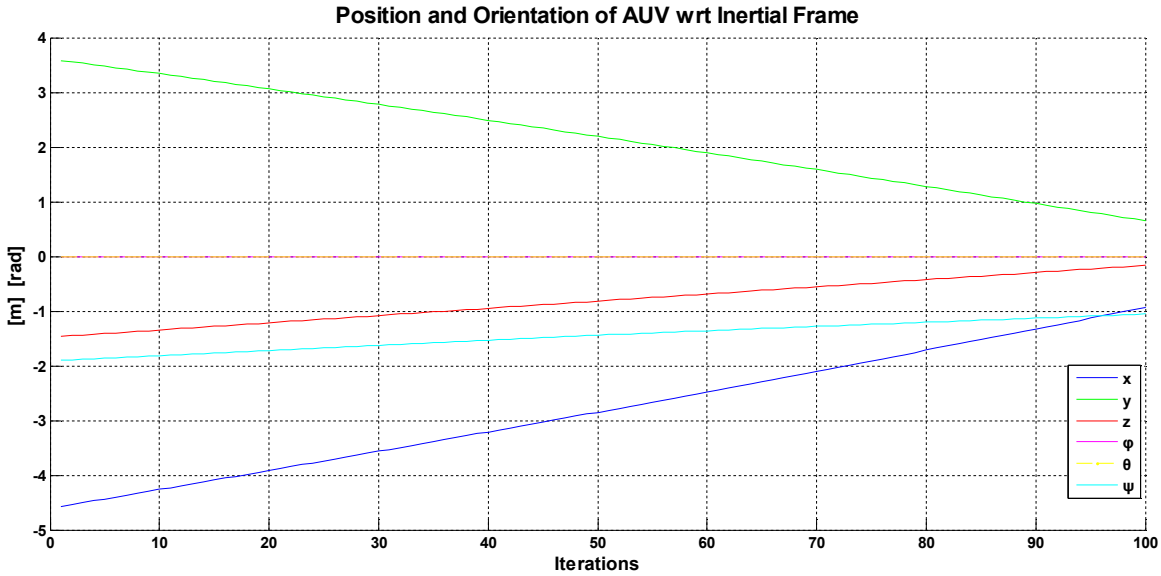



**Figure 4.10:** 1st Solution Process - Case Study 1: AUV optimal position and orientation variables during the reach-to-grasp phase.

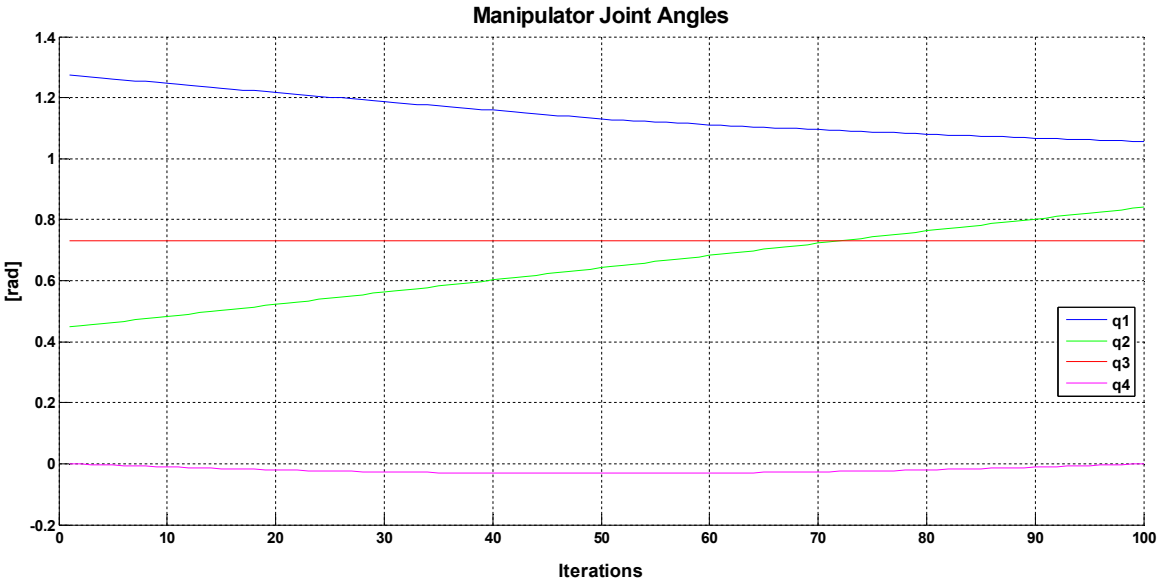
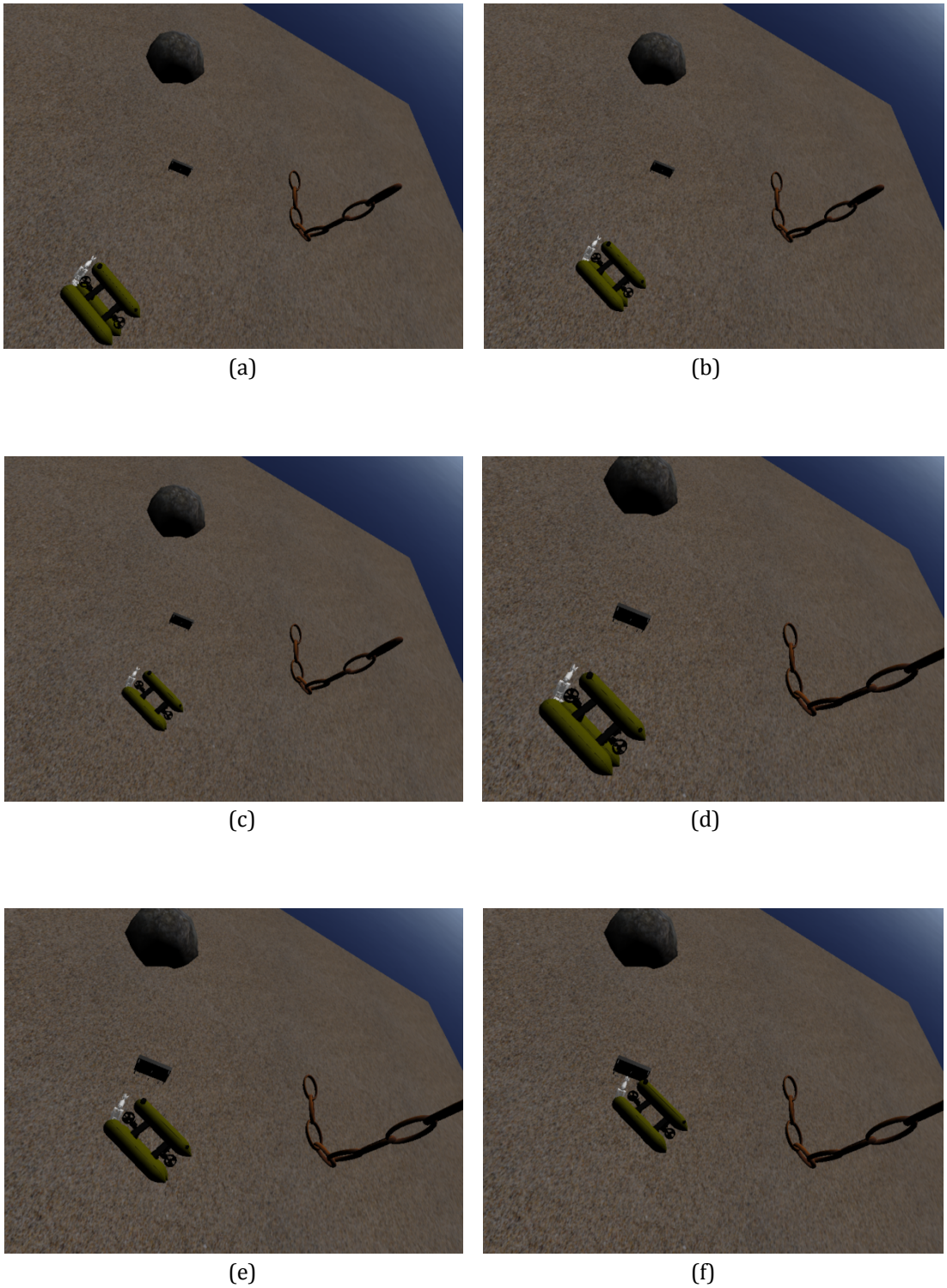



**Figure 4.11:** 1st Solution Process - Case Study 1: Manipulator optimal joint states during the reach-to-grasp phase.

### 4.2.2 Case Study 2

In the second case study, the desired end-effector path starts from the initial desired end-effector pose $\boldsymbol{p}_{e,d\_i} = \begin{bmatrix} -4 & -1.5 & -1 & 0 & -pi/8 & pi/12 \end{bmatrix}^T \text{m}, \text{rad}$ wrt to the inertial frame $\{I\}$ and lasts at the final desired e-e pose $\boldsymbol{p}_{e,d\_f} = \begin{bmatrix} 0 & 0 & 0 & pi/2 & 0 & 0 \end{bmatrix}^T \text{m}, \text{rad}$. As in the first case study, we demand that the vehicle's roll and pitch angles wrt the inertial frame are kept to zero and we want to maximize the magnitude of force along the x-axis of the e-e frame as well as the magnitude of moment about the same axis. The weighting factor of the objective function is selected again to be $w = 0.9$. The desired path, as it is produced by the image based controller, consists of 100 desired end-effector poses.

A depiction of the generated joint space path appears in Fig. 4.12 where the UVMS navigates towards the correct valve head. Fig. 4.12a illustrates the initial optimal UVMS pose configuration, while Fig.4.12b–Fig.4.12e show the optimal UVMS pose configurations each 20 iterations of the path planning scheme. Finally, Fig.4.12f and Fig.4.13 depict the final UVMS pose configuration when the end-effector is ready to grasp the valve head and apply the required interaction wrench.



(a) (b) (c) (d) (e) (f)

**Figure 4.12:** 1st Solution Process - Case Study 2: UVMS optimal pose configurations during the reach-to-grasp phase.

**Figure 4.13:** 1st Solution Process - Case Study 2: UVMS final optimal pose configuration grasping the valve head.

Fig. 4.14 and Fig. 4.15 report the generated sequences of vehicle's position/orientation variables and manipulator's joint angles respectively. We can observe that there is no significant difference in the final UVMS pose configuration of the two case studies, except for the angle of the fourth joint (see Fig. 4.10-4.11 and Fig. 4.14-4.15).

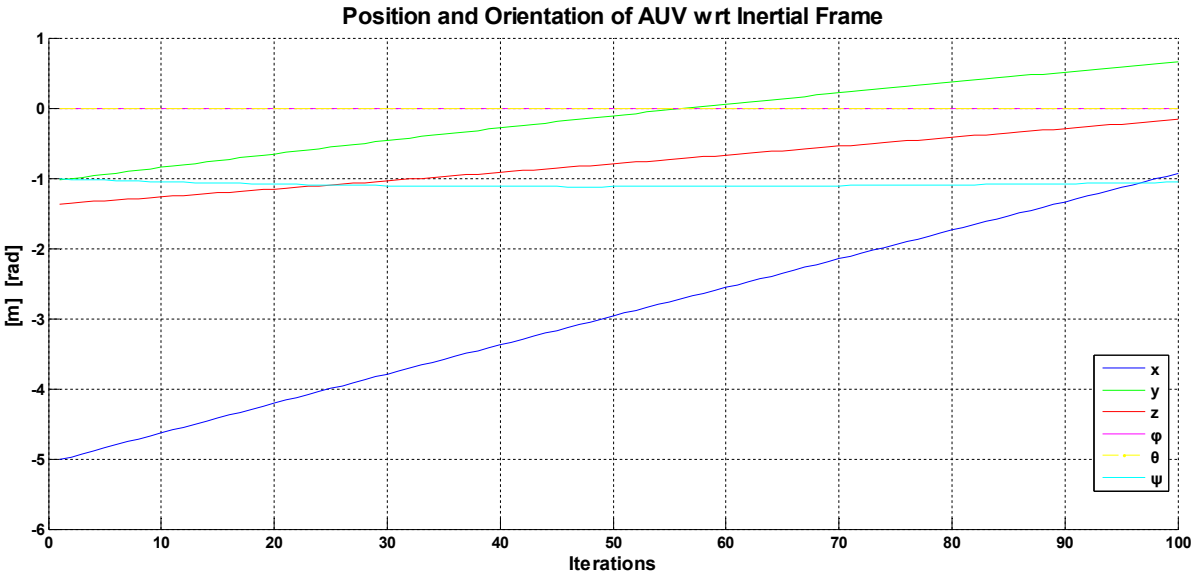



**Figure 4.14:** 1st Solution Process - Case Study 2: AUV optimal position and orientation variables during the reach-to-grasp phase.

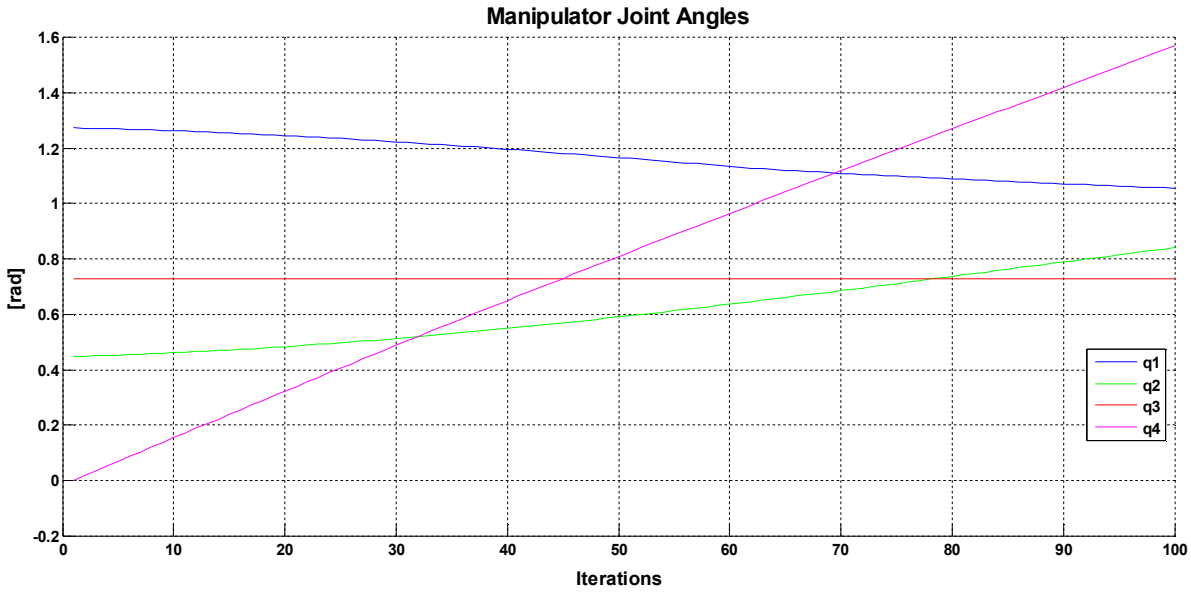
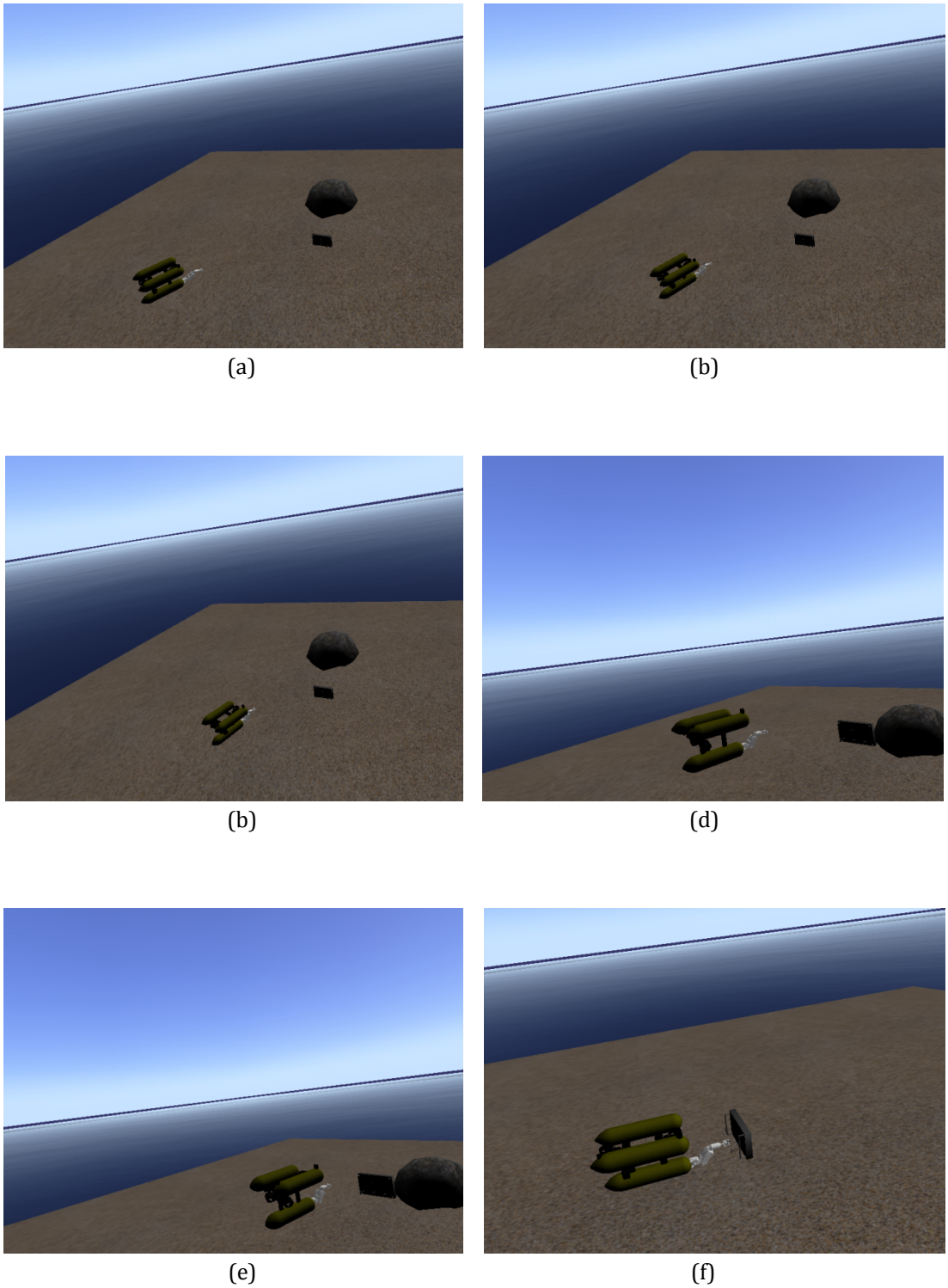



**Figure 4.15:** 1st Solution Process - Case Study 2: Manipulator optimal joint states during the reach-to-grasp phase.

### 4.2.3 Case Study 3

In the third case study, the desired end-effector path starts from the initial desired end-effector pose $\boldsymbol{p}_{e,d\_i} = \left[-4.5 \;\; -3 \;\; -1 \;\; 0 \;\; -pi/10 \;\; pi/6\right]^T \text{m, rad}$ wrt to the inertial frame $\{I\}$ and lasts at the final desired e-e pose which coincides with $\{I\}$. The constraints to be satisfied are the same with the previous case studies (described in §3.3). On the contrary, we want to maximize only the magnitude of force along the x-axis of the e-e frame and so we select the weighting factor of the objective function to be $w = 0$. The desired path of end-effector, as it is produced by the image based controller, consists of 100 desired end-effector poses.

A depiction of the generated joint space path appears in Fig. 4.16 where the UVMS navigates towards the correct valve head. Fig. 4.16a illustrates the initial optimal UVMS pose configuration, while Fig.4.16b–Fig.4.16e show the optimal UVMS pose configurations each 20 iterations of the path planning scheme. Finally, Fig.4.16f and Fig.4.17 depict the final UVMS pose configuration when the end-effector is ready to grasp the valve head and apply the required interaction wrench.



(a) (b)

(b) (d)

(e) (f)

**Figure 4.16:** 1st Solution Process - Case Study 3: UVMS optimal pose configurations during the reach-to-grasp phase.

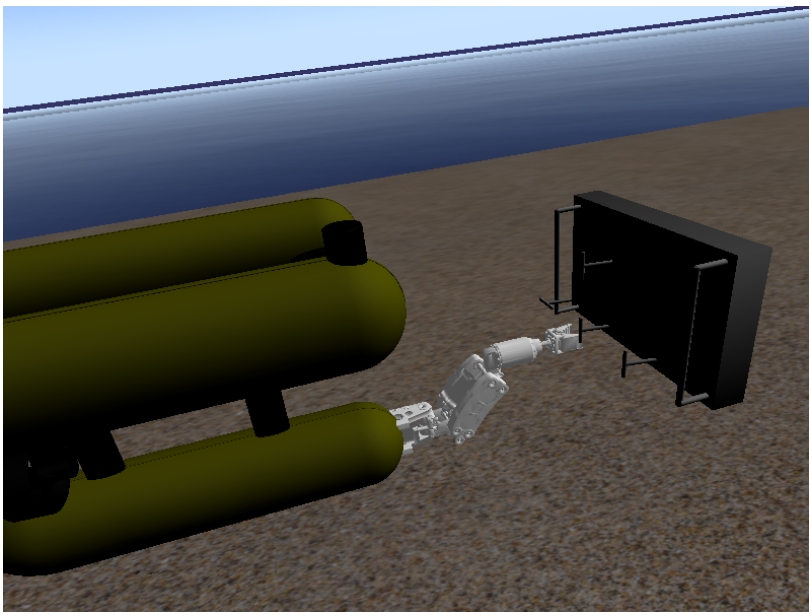


**Figure 4.17:** 1st Solution Process - Case Study 3: UVMS final optimal pose configuration grasping the valve head.

Fig. 4.18 and Fig. 4.19 report the generated sequences of vehicle's position/orientation variables and manipulator's joint angles respectively.

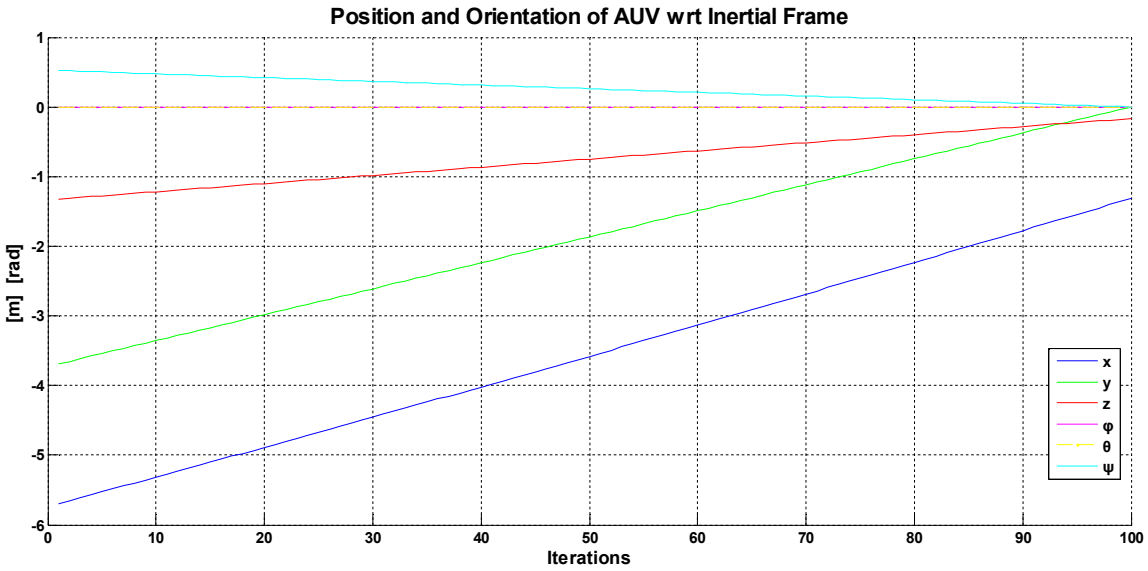



**Figure 4.18:** 1st Solution Process - Case Study 3: AUV optimal position and orientation variables during the reach-to-grasp phase.

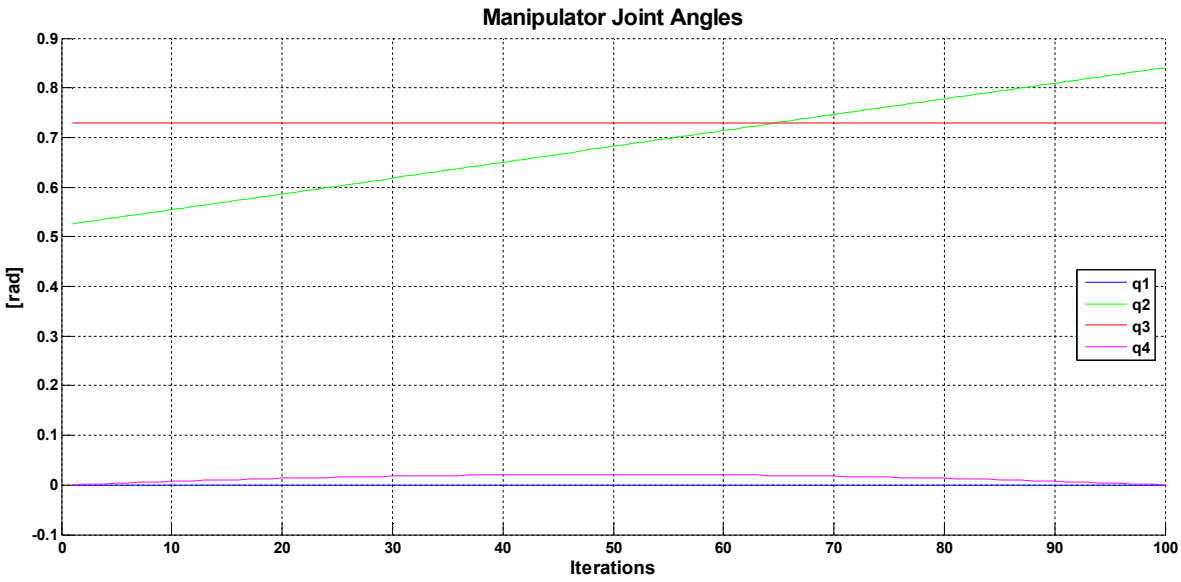
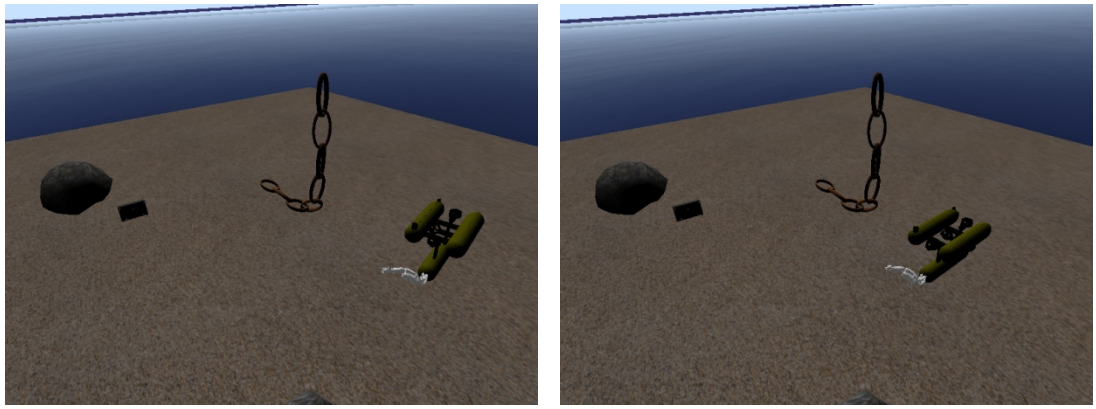



**Figure 4.19:** 1st Solution Process - Case Study 3: Manipulator optimal joint states during the reach-to-grasp phase.

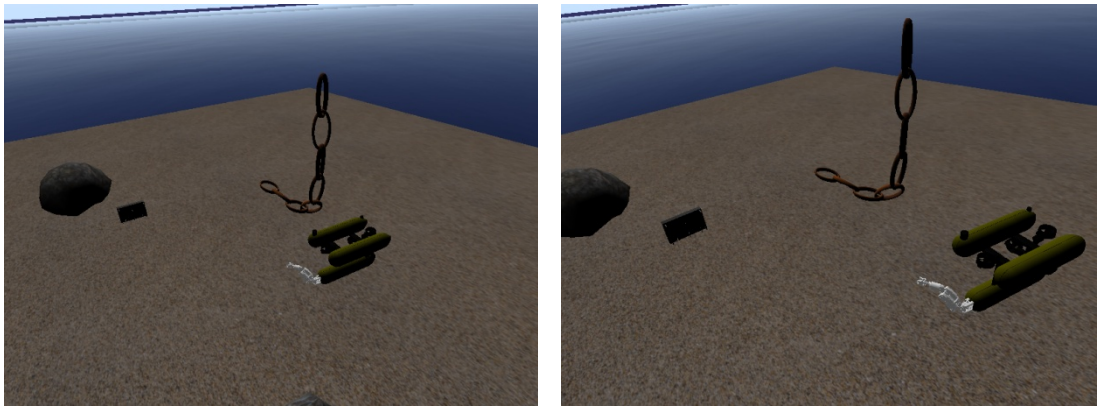
## 4.3 Simulation Results: 2nd Approach of Solution

In this subsection, the performance of the second proposed motion planning scheme, as it was presented in §3.4.2, is verified via MATLAB simulation. The desired end-effector path, generated by the task-space path planner, is the same as in §4.2.1. The proposed iterative algorithm receives as input the above sequence of end effector poses and produces a sequence of UVMS pose configurations that leads in a smooth way to the final optimal configuration for efficient interaction with the environment. The constraints to be satisfied are the same as in the previous simulations. Moreover, as in §4.2.1, we want to maximize the magnitude of force along the x-axis of the e-e frame as well as the magnitude of moment about the same axis and the weighting factor of the objective function is selected to be $w = 0.9$.

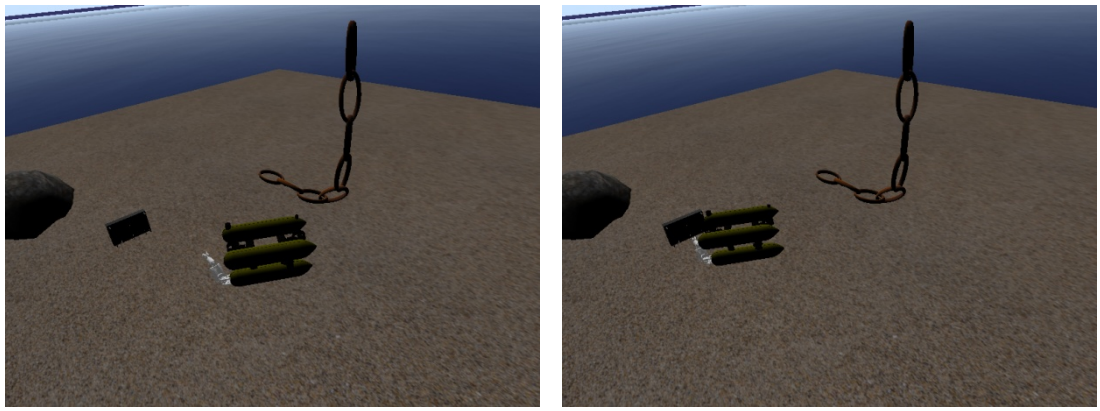
A depiction of the generated joint space path appears in Fig. 4.20 where the UVMS navigates towards the correct valve head. Fig. 4.20a illustrates the initial optimal UVMS pose configuration, while Fig.4.20b–Fig.4.20e show the optimal UVMS pose configurations each 20 iterations of the path planning scheme. Finally, Fig.4.20f and Fig.4.21 depict the final UVMS pose configuration when the end-effector is ready to grasp the valve head and apply the required interaction wrench.



(a) (b)



(b) (d)



(e) (f)

**Figure 4.20:** 2nd Solution Process: UVMS optimal pose configurations during the reach-to-grasp phase.

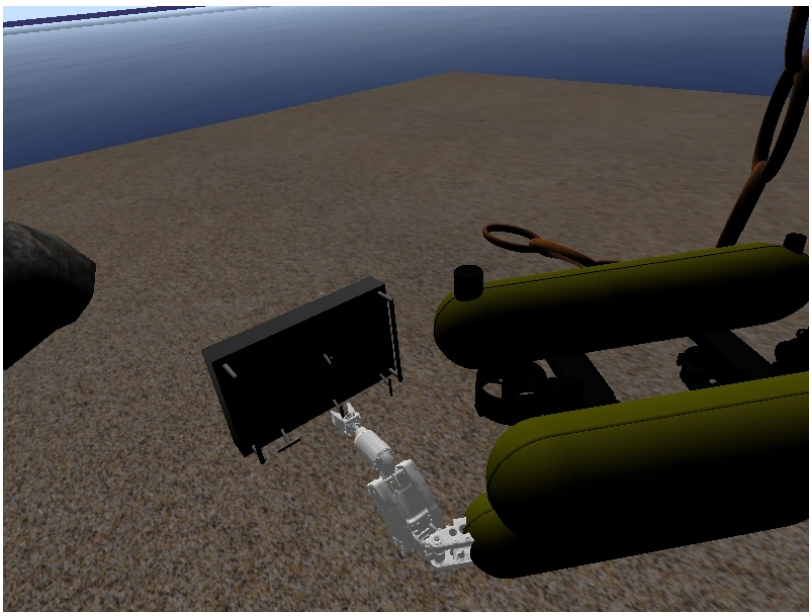


**Figure 4.21:** 2nd Solution Process: UVMS final optimal pose configuration grasping the valve head.

Fig. 4.22 illustrates the generated sequences of vehicle's position and orientation in terms of Euler angles, while Fig. 4.23 depict manipulator's joint angles. It can be observed that there is no significant difference in the final UVMS pose configuration generated by the two proposed motion planning schemes for the same scenario (see Fig. 4.10-4.11 and Fig. 4.22-4.23).

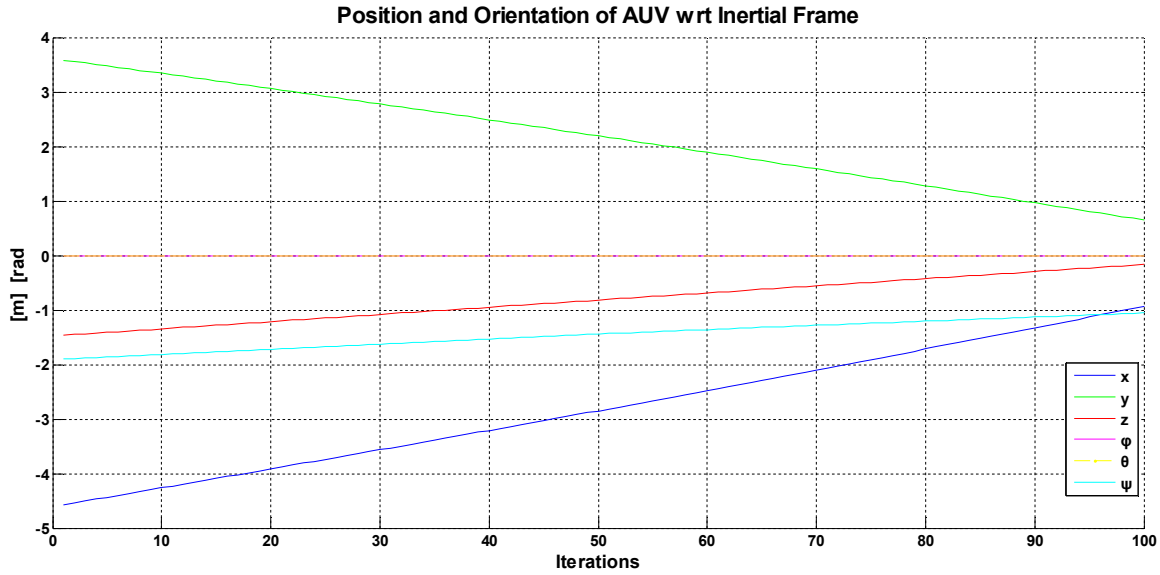



**Figure 4.22:** 2nd Solution Process: AUV optimal position and orientation variables during the reach-to-grasp phase.

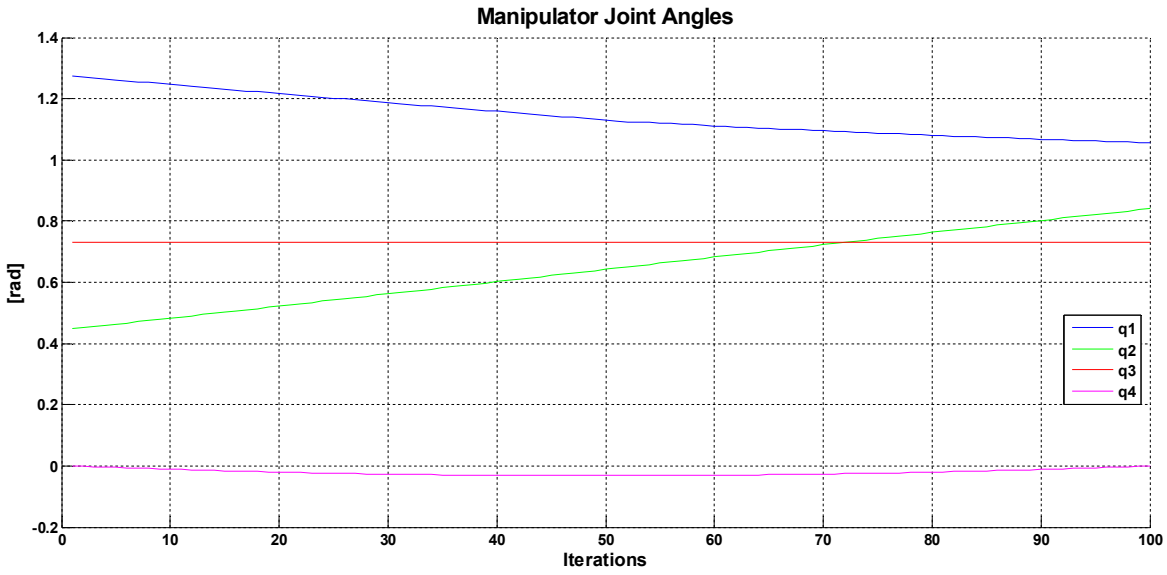



**Figure 4.23:** 2nd Solution Process: Manipulator optimal joint states during the reach-to-grasp phase.

# Chapter 5

# Conclusions & Future Research

## 5.1 Discussion and Conclusions

In this thesis, we presented an optimization scheme that provides an optimal pose configuration of an UVMS for efficient interaction with the environment wrt several intervention requirements and a certain performance criterion (maximization of a meaningfully defined norm of interaction wrench vector), exploiting the redundant dofs of the combined system and ensuring that several constraints are satisfied.

This optimization scheme is incorporated as part of a path planning scheme for UVMS interacting with the environment. In particular, when the UVMS approaches the interaction spot, an image based controller generates in real time a sequence of end-effector poses from the initial pose, when the UVMS enters the "*reach-to-grasp*" phase, to the final desired pose, when the end effector is ready to interact with the environment. Since the control action on the UVMS is carried out in the joint space, a suitable algorithm was proposed to provide in real time a sequence of UVMS pose configurations that leads in a smooth way to the final optimal configuration for efficient interaction wrt the pre-specified performance criterion. This algorithm includes the aforementioned optimization scheme and plays the role of an on-line path planner in the joint space of the UVMS during the "*reach-to-grasp*" phase.

Considering that the nonlinear constraints to be respected make the optimization procedure complicated and computationally intense, a second path planning scheme in the joint space of the UVMS was proposed. This approach speeds up the computation procedure and seems to be more convenient for on-line motion planning schemes. The key idea behind this approach lies in applying sensitivity analysis in an iterative process to derive the sequence of optimal UVMS pose configurations.

Finally, the efficacy of the developed motion planning algorithms and optimization scheme as part of the overall interaction control scheme was demonstrated through a series of simulation studies in MATLAB where various underwater scenarios were considered.

## 5.2 Future Research Directions

As it has been mentioned, the proposed algorithms will be incorporated as part of a two-stage interaction control structure for UVMS performing intervention tasks that involve interaction with the environment. The two-stage interaction control structure is composed of the proposed path planning schemes and a motion control algorithm.

Therefore, it is clear that a real-time, robust, coordinated and adaptive on board nonlinear motion controller for autonomous UVMS has to be designed. This controller will compute the control inputs (driving forces) aimed at tracking the reference path of the system. At the same time, it will compensate for the forces and torques induced by the interaction with the environment and for external disturbances such as the underwater currents. The controller must also allow for an acceptable level of compliance and overcome the issues associated with parameter variations, such as payload variations, model uncertainties, buoyancy variations and internal noises. Moreover, in the case of a force/torque sensor having been mounted on the end-effector of the UVMS, a force control scheme could also be designed.

After the overall interaction control scheme has been designed, its performance will be verified through simulation studies firstly in MATLAB and subsequently in UWSim simulator. Further research work must be related to the experimental testing of the proposed interaction control scheme in order to assess its efficacy under real-life conditions. The experimental trials will be conducted at the Underwater Robotics Lab of the University of Girona. Finally, future research could also be directed towards the incorporation of a more sophisticated optimization algorithm to solve the formulated optimization problem.

# Bibliography


1. K.P. Valavanis, D. Gracanin, M. Matijasevic, R. Kolluru and G. A. Demetriou, "Control Architecture for Autonomous Underwater Vehicles," *IEEE Control Systems,* vol. 17, no. 6, pp. 48-64, Dec 1997.
2. J. Yuh and M. West, "Underwater robotics," *Journal of Advanced Robotics,* vol. 15, no. 5, pp. 609-639, 2001.
3. G. Antonelli, *Underwater Robots. Motion and Force Control of Vehicle-Manipulator Systems*, Series: Springer Tracts in Advanced Robotics, Springer-Verlag, New York, 2006.
4. VideoRay LLC, Official Website. [Online]. Available: http://www.videoray.com
5. Lockheed Martin Corporation, Official Website. [Online]. Available: http://www.lockheedmartin.com
6. G. Marani, S.K. Choi, and J. Yuh, "Underwater autonomous manipulation for intervention missions AUVs," *Ocean Engineering*, vol. 36, no. 1, pp. 15-23, Jan 2009.
7. J. Yuh, *Underwater Robotic Vehicles: Design and Control*, TSI Press, 1995.
8. I. Schjølberg and T.I. Fossen, "Modelling and control of underwater vehicle-manipulator systems," *in Proceedings of the Third Conference on Marine-craft Maneuvering and Control*, Southampton, pp. 45-57, 1994.
9. H. Mahesh, J. Yuh and R. Lakshmi, "A coordinated control of an underwater vehicle and robot manipulator," *Journal of Robotic Systems*, vol. 8, no. 3, pp. 339-370, Jun 1991.
10. H. Schempf and D.R. Yoerger, "Coordinated vehicle/manipulator design and control issues for underwater telemanipulation," *in Proceedings of the IFAC Control Applications in Marine System*, pp. 259-268, Italy, Apr 1992.
11. F. Lizarralde, J. Wen and L. Hsu, "Quaternion-based coordinated control of a subsea mobile manipulator with only position measurements," *in Proceedings of the 34th IEEE Conference on Decision and Control*, vol.4, pp.3996-4001, Dec 1995.
12. T.W. McLain, S.M. Rock and M.J. Lee, "Experiments in the Coordinated Control of an Underwater Arm/Vehicle System," *Autonomous Robots*, vol. 3, no. 2, pp. 213-232, 1996.
13. T.J. Tarn, G.A. Shoults and S.P. Yang, "A dynamic model of an underwater vehicle with a robotic manipulator using Kane 's method," *Autonomous Robots*, vol. 3, no. 2-3, pp. 269-283, 1996.
14. G. Antonelli and S. Chiaverini, "Task-priority redundancy resolution for underwater vehicle-manipulator systems," *in IEEE International Conference on Robotics and Automation*, vol. 1, pp. 768-773, May 1998.
15. 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, Jul 1998.
16. G. Antonelli and S. Chiaverini, "Singularity-free regulation of underwater vehicle-manipulator systems," *in American Control Conference*, vol. 1, pp. 399-403, Jun 1998.
17. O. Olguin Diaz, C. Canudas de Wit and M. Perrier, "A comparative study of neglected dynamics on an underwater vehicle/manipulator system under nonlinear robust control," *in Proceedings of the* OCEANS Conference, vol.2, pp. 936-940, Oct 1998.
18. N. Sarkar, J. Yuh and T.K. Podder, "Adaptive control of underwater vehicle-manipulator systems subject to joint limits," *in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp. 142-147, 1999.
19. C. Canudas de Wit, O. Olguin Diaz and M. Perrier, "Nonlinear control of an underwater vehicle/manipulator with composite dynamics," *IEEE Transactions on Control Systems Technology*, vol.8, no. 6, pp. 948-960, Nov 2000.

20. G.B. Chung, K.S. Eom, B.J. Yi, I.H. Suh, S.R. Oh and Y.J. Cho, "Disturbance observer-based robust control for underwater robotic systems with passive joints," *in Proceedings of the IEEE International Conference on Robotics and Automation*, vol.2, pp.1775-1780, 2000.
21. Y. Cui and N. Sarkar, "A unified force control approach to autonomous underwater manipulation," *Robotica*, vol.19, pp. 255-266, 2001.
22. N. Sarkar and T.K. 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, Apr 2001.
23. J. Yuh, S. Zhao and P.M. Lee, "Application of adaptive disturbance observer control to an underwater manipulator," *in Proceedings of the IEEE International Conference on Robotics and Automation*, vol.4, pp. 3244-3248, Seoul, Korea, 2001.
24. G. Antonelli, F. Caccavale and S. Chiaverini, "Adaptive tracking control of underwater vehicle-manipulator systems based on the virtual decomposition approach," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 594-602, 2004.
25. Y. Sun and C. Cheah, "Adaptive setpoint control of underwater vehicle-manipulator systems," *IEEE Conference on Robotics, Automation and Mechatronics*, vol.1, pp. 434-439, Dec 2004.
26. M. Ishitsuka, S. Sagara and K. Ishii, "Dynamics analysis and resolved acceleration control of an autonomous underwater vehicle equipped with a manipulator," *International Symposium on Underwater Technology*, pp. 277-281, Apr 2004.
27. S. Sagara, M. Tamura, T. Yatoh and K. Shibuya, "Digital RAC for underwater vehicle-manipulator systems considering singular configuration," *Artificial Life and Robotics*, vol. 10, no. 2, pp. 106-111, Nov 2006.
28. C.H. Dos Santos, G. Bittencourt, R. Guenther and E. De Pieri, "A Fuzzy hybrid singularity avoidance for underwater vehicle-manipulator systems," *in 12th IFAC Symposium on Information Control Problems in Manufacturing-MCOM*, pp. 209-214, 2006.
29. M. Ishitsuka and K. Ishii, "Development and control of an underwater manipulator for AUV," *in Symposium on Underwater Technology and Workshop on Scientific Use of Submarine Cables and Related Technologies*, pp. 337-342, Apr 2007.
30. T. Yatoh, S. Sagara and M. Tamura, "Digital type disturbance compensation control of a floating underwater robot with 2 link manipulator," *Artificial Life and Robotics*, vol. 13, pp. 377-381, 2008.
31. J. Han and W. Chung, "Coordinated motion control of underwater vehicle-manipulator system with minimizing restoring moments," *in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3158-3163, Nice, France, Sep 2008.
32. S. Soylu, B.J. Buckham and R.P. Podhorodeski, "Redundancy resolution for underwater mobile manipulators," *Ocean Engineering*, vol. 37, no. 2-3, pp. 325-343, 2010.
33. J. Han, J. Park and W.K. 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.
34. M. Santhakumar and J. Kim, "Modelling, simulation and model reference adaptive control of autonomous underwater vehicle-manipulator systems," *in Proceedings of the 11th International Conference on Control, Automation and Systems*, pp. 643-648, KINTEX, Korea, Oct 2011.
35. M. Santhakumar and J. Kim, "Indirect adaptive control of an autonomous underwater vehicle-manipulator system for underwater manipulation tasks," *Ocean Engineering*, vol. 54, pp. 233-243, Nov 2012.
36. M. Santhakumar, J. Kim and Y. Kim, "A Null Space Control of an Underactuated Underwater Vehicle-Manipulator System under Ocean Currents," *in Proceedings of the Oceans Conference*, pp. 1-5, Yeosu, May 2012.
37. O. Korkmaz, S.K. Ider and M.K. Ozgoren, "Control of an Underactuated Underwater Vehicle Manipulator System in the Presence of Parametric Uncertainty and Disturbance," *in Proceedings of the American Control Conference (ACC)*, pp. 578-584, Washington, Jun 2013.

38. L. LaPierre, P. Fraisse and N. M'Sirdi, "Hybrid position/force control of a ROV with a manipulator," *in Proceedings of the Oceans Conference*, vol. 2, pp. 931–935, 1998.
39. J. Ryu, D. Kwon and P. Lee, "Control of underwater manipulators mounted on a ROV using base force information," *in Proceedings of the IEEE International Conference on Robotics and Automation*, vol.4, pp. 3238-3243, Seoul, Korea, May 2001.
40. J. Kim, W. Chung and J. Yuh, "Dynamic analysis and two-time scale control for underwater vehicle-manipulator systems," *in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol.1, pp. 577-582, Las Vegas, USA, Oct 2003.
41. ECA Robotics, Official Website. [Online]. Available: http://www.eca-robotics.com
42. TRIDENT EU FP7 Project Website. [Online]. Available: http://www.irs.uji.es/trident/index.html
43. J. Evans, P. Redmond, C. Plakas, K. Hamilton and D. Lane, "Autonomous docking for Intervention-AUVs using sonar and video-based real-time 3D pose estimation," *in Proceedings of the Oceans Conference*, vol. 4, pp. 2201-2210, Sep 2003.
44. M. Prats, D. Ribas, N. Palomeras, J. C. García, V. Nannen, J. J. Fernández, J. P. Beltrán, R. Campos, P. Ridao, P. J. Sanz, G. Oliver, M. Carreras, N. Gracias, R. Marín and A. Ortiz, "Reconfigurable AUV for Intervention Missions: A Case Study on Underwater Object Recovery," *Journal of Intelligent Service Robotics*, vol. 5, no. 1, pp. 19-31, Jan 2012.
45. T. Asokan, G. Seet, M. Lau and E. Low, "Optimum positioning of an underwater intervention robot to maximize workspace manipulability," *Mechatronics*, vol.15, pp. 747-766, 2005.
46. P. Sotiropoulos, N. Aspragathos and F. Andritsos, "Determination of the optimum docking position of an underwater unmanned vehicle using a genetic algorithm," *Proceedings of the World Congress on Engineering and Computer Science (WCECS)*, vol. 1, pp. 340-345, Oct 2010.
47. B. Jun, P. Lee and S. Kim, "Manipulability analysis of underwater robotic arms on ROV and application to task-oriented joint configuration," *Journal of Mechanical Science and Technology*, Vol. 22, no. 5 , pp. 887-894, May 2008.
48. PANDORA EU FP7 Project Website.[Online].Available: http://www.persistentautonomy.com
49. C.A. Floudas and P.M. Pardalos, *Encyclopedia of Optimization*, Second Edition, Springer, 2009.
50. S.S. Rao, *Introduction to Optimization, in Engineering Optimization: Theory and Practice*, Fourth Edition, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2009.
51. Johnson E.H., *Tools for structural optimization*, chap.29, pp. 851-863, in Structural Optimization: Status and Promise, M.P. Kamat, Ed., AIAA, Washington, DC, 1993.
52. H.R.E.M. Hornlein and K. Schittkowski, *Software Systems for Structural Optimization*, Birkhauser, Basel, 1993.
53. J.J. More and S.J. Wright, *Optimization Software Guide*, Society of Industrial and Applied Mathematics, Philadelphia, PA, 1993.
54. Mathworks, Official Website. [Online]. Available: http://www.mathworks.com
55. Mathworks, Matlab Optimization Toolbox User's Guide. [Online]. Available: http://www.mathworks.co.uk/access/helpdesk/help/pdf_doc/optim/optim_tb.pdf
56. E. Castillo, A.J. Conejo, C. Castillo and R. Minguez, "Closed formulas in local sensitivity analysis for some classes of linear and non-linear problems," *Journal of the Spanish Society of Statistics and Operations Research*, vol. 15, no. 2, pp. 355-371, Dec 2007.
57. A.V. Fiacco, *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, Academic Press, New York, NY, 1983.
58. G.N. Vanderplaats, *Numerical Optimization Techniques for Engineering Design*, McGraw-Hill, New York, NY, 1984.
59. J.S. Sobieski, J.F. Barthelemy and K.M. Riley, "Sensitivity of Optimal Solutions to Problems Parameters," *Journal of American Institute of Aeronautics and Astronautics (AIAA)*, vol. 20, pp. 1291-1299, 1982.
60. I. Enevoldsen, "Sensitivity Analysis of Reliability-Based Optimal Solutions," *ASCE Journal of Engineering Mechanics*, vol. 120, pp. 198–205, 1994.

61. C. Roos, T. Terlaky and J.P. Vial, *Theory and Algorithms for Linear Optimization: An Interior-Point Approach,* John Wiley & Sons, Chichester, UK, 1997.
62. P. Bjerager and S. Krenk, "Parametric Sensitivity in First-Order Reliability Theory," *ASCE Journal Engineering Mechanics,* vol. 115, pp. 1577-1582, 1989.
63. E. Castillo, A.J. Conejo, C. Castillo, R. Mínguez and D. Ortigosa, "A perturbation approach to sensitivity analysis in mathematical programming," *Journal of Optimization Theory and Applications*, vol. 128, no.1, pp. 49-74, 2006.
64. J.F. Bonnans and A. Shapiro, "Optimization problems with perturbations: a guided tour," *Society for Industrial and Applied Mathematics*, vol. 40, no. 2, pp. 228-264, 1998.
65. J.F. Bonnans and A. Shapiro, *Perturbation analysis of optimization problems*, Springer, New York, 2000.
66. D. Ribas, P. Ridao, L. Magi, N. Palomeras and M. Carreras, "The Girona 500, a multipurpose autonomous underwater vehicle," *IEEE OCEANS*, pp. 1-5, Santander, Spain, Jun 2011.
67. D. Ribas, N. Palomeras, P. Ridao, M. Carreras and A. Mallios, " Girona 500 AUV: From Survey to Intervention," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no.1, pp. 46-53, Feb 2012.
68. ECA Robotics, [Online]. Available: www.eca-robotics.com/ftp/ecatalogue/551/ARM_5E_MICRO.pdf
69. M. Prats, J. Perez, J.J. Fernandez and P.J. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2577-2582, Oct 2012.