ARCHIVIO ISTITUZIONALE
DELLA RICERCA

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Autonomous Underwater Intervention: Experimental Results of the MARIS Project

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:

Availability:
This version is available at: https://hdl.handle.net/11585/643432 since: 2018-09-17

Published:

DOI: http://doi.org/10.1109/JOE.2017.2733878

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/).
When citing, please refer to the published version.

(Article begins on next page)

20 April 2024

This is the final peer-reviewed accepted manuscript of:

*E. Simetti et al., "**Autonomous Underwater Intervention: Experimental Results of the MARIS Project**," in IEEE Journal of Oceanic Engineering, vol. 43, no. 3, pp. 620-639, July 2018*

The final published version is available online at:
https://doi.org/10.1109/JOE.2017.2733878

Rights / License:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/)

When citing, please refer to the published version.

Autonomous Underwater Intervention: Experimental Results of the MARIS Project

Enrico Simetti, Francesco Wanderlingh, Sandro Torelli, Marco Bibuli, Angelo Odetti, Gabriele Bruzzone, Dario Lodi Rizzini, Jacopo Aleotti, Gianluca Palli, Lorenzo Moriello, and Umberto Scarcia

Abstract

Autonomous Underwater Vehicles are frequently used for survey missions and monitoring tasks, however manipulation and intervention tasks are still largely performed with a human in the loop. Employing autonomous vehicles for these tasks has received a growing interest in the last ten years, and few pioneering projects have been funded on this topic. Among these projects, the Italian MARIS project had the goal of developing technologies and methodologies for the use of autonomous Underwater Vehicle Manipulator Systems in underwater manipulation and transportation tasks. This work presents the developed control framework, the mechatronic integration, and the project's final experimental results on floating underwater intervention.

Index Terms

underwater vehicle manipulator system; underwater gripper; underwater vision; floating underwater control; task priority control; underwater intervention.

E. Simetti, F. Wanderlingh, S. Torelli are with DIBRIS, University of Genova and with the Interuniversity Research Center on Integrated Systems for the Marine Environment (ISME), Via Opera Pia 13, 16145, Genova, Italy, e-mail: enrico.simetti@unige.it

G. Palli, L. Moriello, U. Scarcia are with DEI, Università di Bologna, Viale Risorgimento 2, Bologna, Italy and with the Interuniversity Research Center on Integrated Systems for the Marine Environment (ISME), Via Opera Pia 13, 16145, Genova, Italy

D. Lodi Rizzini, J. Aleotti are with the Robotics and Intelligent Machines Laboratory, Department of Engineering and Architecture, University of Parma, 43124, Italy, e-mail: {dlr;kallasi;aleotti}@ce.unipr.it.

M. Bibuli, A. Odetti, G. Bruzzone are with the National Research Council - Institute of Studies on Intelligent Systems for Automation CNR-ISSIA, Via De Marini 6, 16149 Genova, Italy, Email: marco.bibuli@ge.issia.cnr.it

I. INTRODUCTION

Autonomous underwater vehicles (AUVs) have important applications in ocean related fields, such as monitoring environmental parameters [1], detection of new zones to be exploited for resource gathering [2], monitoring and exploration of archaeological sites [3], and security applications [4] to name but a few.

In the above applications, AUVs perform survey missions. A recent trend is the use of AUVs for performing inspection, repair and maintenance operations. The autonomous UVMS (Underwater Vehicle Manipulator System) solution is very appealing for its cost and human safety aspects compared to current technology. Indeed, in the last decade Chevron has initiated a research program on the use of Intervention AUVs (I-AUVs) to develop a permanent system to increase the frequency of inspections, reducing the overall cost of field maintenance [5].

Research on improving autonomy for underwater manipulation tasks can be dated back to 1990s, in particular to the works on compliant underwater manipulators [6] and coordinated vehicle/arm control for teleoperation [7]. A first milestone was achieved with the AMADEUS project [8] which demonstrated dual arm autonomous manipulation in water tank experiments.

Those efforts were still exploring basic control problems. The first mechatronic assembly of an UVMS was carried within the pioneering project UNION [9]. Another milestone was achieved within the ALIVE project, with the demonstration of autonomous docking on a ROV friendly panel [10] and fixed-base manipulation of a valve. Shortly after ALIVE, the SAUVIM project [11], [12] has instead shown for the first time an UVMS performing autonomous floating manipulation tasks.

Recently, the Spanish project RAUVI [13] studied the problem of finding and recovering a flight data recorder placed at an unknown position at the bottom of a water tank. The recovery, carried out with a special hook, was demonstrated using a decoupled control of the base, which was hovering, and the manipulator [14].

The TRIDENT project [15] followed the trend of the SAUVIM and RAUVI projects. A major difference with those projects is that a *coordinated control* of the UVMS was implemented, exploiting a task priority framework together with a dynamic programming approach. The control framework developed within the TRIDENT project explicitly dealt with the activation and deactivation of scalar control objectives, to allow safety tasks to be at higher priority

and deactivate them whenever not necessary. The project ended with a demonstration of an autonomous underwater recovery of a blackbox mockup in a harbor environment [16].

Partially overlapping in time with the TRIDENT project, the PANDORA project focused on the execution of floating valve-turning operations on a panel, combining learning by demonstration and a task priority control [17], [18]. However, the original task priority framework [19] was adopted, which does not have the ability to activate/deactivate control tasks without discontinuity in the control variables. For that reason, the end-effector position control task was placed at the highest priority, while safety tasks only *attempted* at lower priority. Ad-hoc modifications, outside of the task priority paradigm, were introduced during the experimental trials to cope with the priority inversion problem, resulting in the loss of the task priority properties.

Finally, in the MERBOTS project [20] the authors have used the motion planning ROS package *MoveIt!* to compute reference trajectories for the UVMS. However, instead of integrating motion planning with control, they completely replaced the latter by computing trajectories in the configuration space. For that reason, the resulting behavior requires that each degree of freedom reaches its configuration waypoint, creating many stop and go motions. Furthermore, the architecture does not allow for multi rate control of the arm and the vehicle, does not exploit the better performances of the arm joints w.r.t. the vehicle thrusters, and nonactuated degrees of freedom are not properly taken into account, as the authors themselves underline.

This work presents the major outcomes of the Italian MARIS project [21], which was coordinated by the Genova node of the Italian Research Center on Integrated System for the Marine Environment (ISME), with the participation of Bologna, Cassino, Salento and Pisa ISME nodes, CNR-ISSIA (Genova node) and University of Parma. The goal of the project was the development and integration of technologies and methodologies to automate underwater manipulation and transportation activities.

In particular, this work reports the most important novel results of the project, which are the task priority control framework and the MARIS gripper, both experimentally proved. The special focus on the three test campaigns and the knowledge gained while moving from theory to practice represents another important contribution of this work. Insights on the developed artificial vision techniques and integration efforts are also given.

In comparison with the latest underwater intervention projects (RAUVI, TRIDENT, PANDORA, MERBOTS), the following innovative aspects of the work presented in this paper can

be emphasized:

- The control framework developed within the project has extended the original task priority framework [19] to allow control tasks activation and deactivation without discontinuities in the control variables. With this possibility, inequality control objectives such as joint limits can be taken into account only when they are close to be violated, without overconstraining the system. This allows safety tasks, which are mostly of inequality type, to be put at the highest priority, overcoming the problems outlined in [18]. With respect to TRIDENT, the framework allows the activation and deactivation of control tasks of any dimension, not just scalar ones.
- The new pseudo inversion scheme allows using smaller damping values, compared to the dynamic programming technique that was used in the TRIDENT project [22].
- The developed control framework encompasses two parallel optimizations that take into account the differences between the vehicle and arm degrees of freedom. The control of the arm is *optimal* w.r.t. the current *measured* vehicle velocities, increasing the performance of floating intervention activities, overcoming the shortcomings underlined in [20] where the motion planning could not differentiate between the two subsystems. In addition, the proposed framework allows a seamless coordinated control of the arm and vehicle, without *imposing* sequential vehicle-arm motions as in [14]. However, if needed, such a behavior can still be reproduced.
- Furthermore, the parallel optimizations allow an easy implementation of multi-rate control of the arm and the vehicle, a feature that was lacking in [20], [18].
- The aforementioned features have been implemented maintaining an invariant and uniform algorithmic structure, without requiring ad-hoc modifications outside of the task priority paradigm as in [18].

Concerning the MARIS gripper, the major improvements include:

- Dimensions and weight have been significantly reduced compared to TRIDENT [23] (Fig. 2);
- Structural elements have been manufactured in Aluminium T7075 instead of 3D printing;
- An optoelectronic F/T (force/torque) sensor has been integrated into the wrist interface;
- A camera characterized by a resolution of 1024x768 pixel and 30 fps has been placed in the gripper palm to allow close reconstruction of the environment and the objects during

grasping activities;

- A pair of high-power LED capable of 3000 lumens each have been integrated in the gripper palm to allow operation also in dark environments like the underwater one;
- The motors have been encapsulated inside independent sealed tubes, allowing fast motor substitution without affecting the sealing of the other mechanical and electric parts.

The major improvement and contributions of the vision system consist of the following points:

- The stereo camera rig has customizable baseline and varifocal lenses that can be adapted to specific vision tasks; the vision system also achieves a trade-off between on-board processing performance (about 10 Hz algorithm execution, compared to the approximately 2 Hz of the TRIDENT vision system), and balance of power consumption and thermal dissipation;
- The chosen target objects to be grasped are cylindrical and patternless, often only partially visible during the execution of manipulation tasks, whereas previous works exploited full visibility or specific object patterns; detection and pose estimation is based on robust features like color, shape and dimension and the approach can be extended to more general objects with distinctive color and regular shape;
- The detection algorithm manages partial occlusion by the AUV manipulator occurring while approaching large objects, whereas in TRIDENT the control was required to avoid occlusions;
- The image processing pipeline has been tested in both night and daylight conditions.

The article is structured as follows. Section II describes the mechatronics of the MARIS system, including the vehicle, manipulator and vision subsystems and the newly developed MARIS gripper. Section III describes the control architecture. Section IV reports the vision techniques used for estimating the position of the target object to be grasped, while Section V describes the software architecture of the MARIS UVMS. Section VI reports the results on floating underwater manipulation. Finally, some conclusions are given in Section VII.

II. Mechatronics of the MARIS Underwater Vehicle Manipulator System

This section presents the mechatronics of the MARIS UVMS. A brief description of the vehicle and manipulator subsystems is given, since the vehicle was not developed within MARIS and the arm is a commercial product. Then, the developed vision system and the gripper are described.

A. UVMS

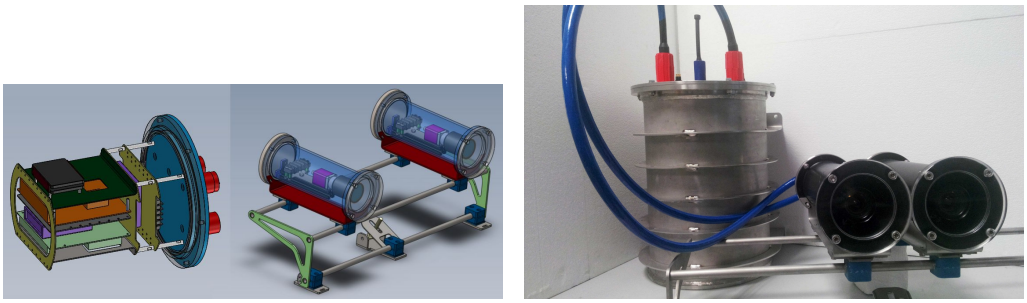
The mobile robotic platform employed in the MARIS project as underwater floating carrier is the R2 ROV/AUV developed by CNR-ISSIA, an evolution of the former Romeo ROV. It is an open-frame fully actuated robotic platform designed to have a compact size, comparable to small/medium class ROV/AUV; its dimensions are: 1.3 m length, 0.9 m width, 1.0 m height. The total weight (in air) of the platform can range from 350 up to 500 kg, depending on the specific sensor package and mission payload. The control in the 6 degrees of freedom is provided by a redundant actuation framework composed by 4 vertical and 4 horizontal thrusters. As proprioceptive sensors, the R2 ROV/AUV is equipped with a Microstrain 3DM-GX3-35 GPS and AHRS (Attitude Heading Reference System) unit and within the MARIS project, a Teledyne Explorer DVL (Doppler Velocity Logger) device was integrated, to track the sea bottom and to determine the vehicle velocity.

The MARIS manipulator is based on the Underwater Modular Manipulator (UMA) developed within the TRIDENT project [15] and now commercialized by Graal Tech srl. Since the manipulator itself is not a product of the MARIS project, it will not be described here. The interested reader can find more details in [23]. As main difference w.r.t. the TRIDENT project, where UMA was originally conceived, the MARIS manipulator's third link has a 'T' shape. This particular choice was made to fold the arm as much as possible during the deployment and navigation of the UVMS.

B. Vision System

The computer vision system provides information required for the autonomous execution of manipulation and grasping tasks of target objects. For the MARIS project, the goal was to detect cylindrical target objects and provide their 3D pose w.r.t. the vehicle to the control system using stereo vision.

In general, image processing applications are computationally demanding and require high performance CPUs with significant power consumption. The developed vision system achieves a trade-off between on-board processing performance (about 10 Hz algorithm execution), and balance of power consumption and thermal dissipation. The adopted hardware solution is a system comprising two x86 CPUs and a microcontroller. The main computational unit performs the artificial vision calculations. while the auxiliary CPU provides other safety and monitoring



(a) (b)

Fig. 1. (a) 3D CAD model of the vision system: internal view of the vision canister (left) and stereo rig (right). (b) Underwater vision system.

services [24], [25]. The microcontroller is responsible for synchronized camera triggering and temperature monitoring inside the canister.

The imaging subsystem is based on two AVT Mako G125C GigE cameras connected to the main CPU through a standard Ethernet link with support to PoE (Power over Ethernet). Accurate pose estimation requires full camera resolution (1292×964 pixels), while image processing at a lower resolution is supported directly on board by sensor level image binning.

For higher flexibility, varifocal lenses with a focal length between 4.4 mm and 11.0 mm have been chosen. Moreover, cameras are housed in separate canisters arranged in a rig allowing changes in baseline and pitch configurations (see Figure 1).

C. MARIS Gripper

The gripper designed for the MARIS project represents the evolution of previous devices developed for the TRIDENT project and described in [23], [26], [27]. Figure 2 shows a comparison between the current gripper and its predecessor.

The main functional specification of the MARIS gripper are: *i*) ability to grasp cylindrical objects with diameter from 5 to 200 mm; *ii*) operating depth 50 m; *iii*) irreversible actuators; *iv*) precision, parallel and power grasps; *v*) transmission system compliance to adapt to object shape irregularities and uncertain dimensions; *vi*) an embedded wrist F/T sensor.

According to these design specifications, the size and shape of objects to be grasped is ample. However, since internal manipulation capability is not required, a solution composed by a mechanism with three fingers capable of a large workspace has been adopted.

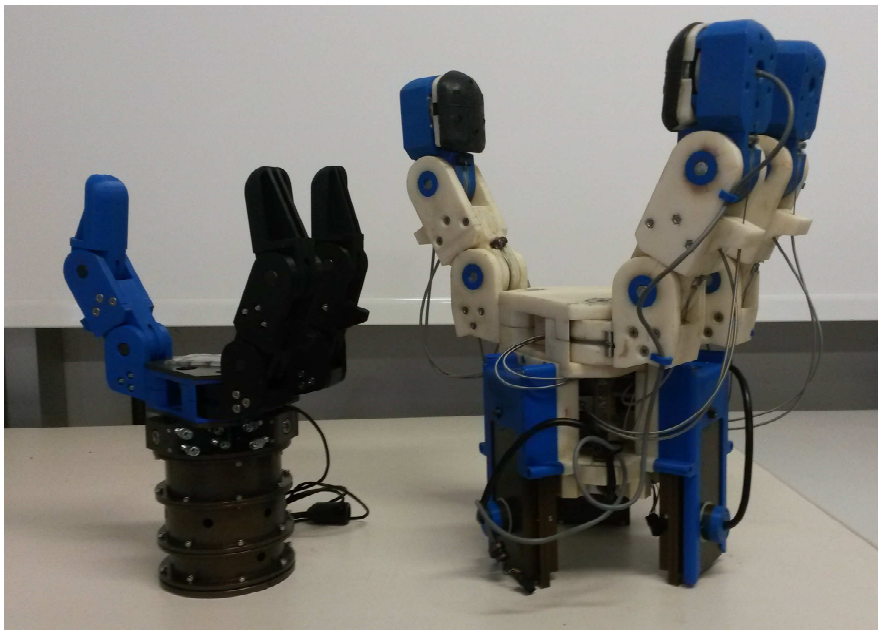


Fig. 2. The TRIDENT (right) and the MARIS (left) grippers.

In particular, the finger structural elements are manufactured in ABS plastic to reduce the weight, to increase the buoyancy and to prevent damages to the other gripper components in case of accidental collision with heavy objects or with the environment. Differently from the TRIDENT gripper, anodized aluminium has been adopted for structural parts, such as the wrist flange and the palm structure, to prevent corrosion.

The overall dimensions have been substantially reduced, as can be seen in Fig. 2. The weight is about 4.5 daN in air, while in water it is about 1 daN, which can be easily compensated by adding proper floats.

A camera characterized by a resolution of 1024x768 pixel and 30 fps and a couple of high-power LEDs capable of 3000 lumens each have been integrated in the gripper palm to allow object detection at close range during grasping activities. Finally, an optoelectronic F/T sensor [28]–[30] has been added in the wrist interface.

Further details on the kinematics, actuation and the F/T sensors are reported in the following subsections.

1) Kinematics: In Fig. 3 a schematic view of the gripper kinematics is reported. The gripper has three fingers: one named T (which can be intended as an opposable thumb), and two identical

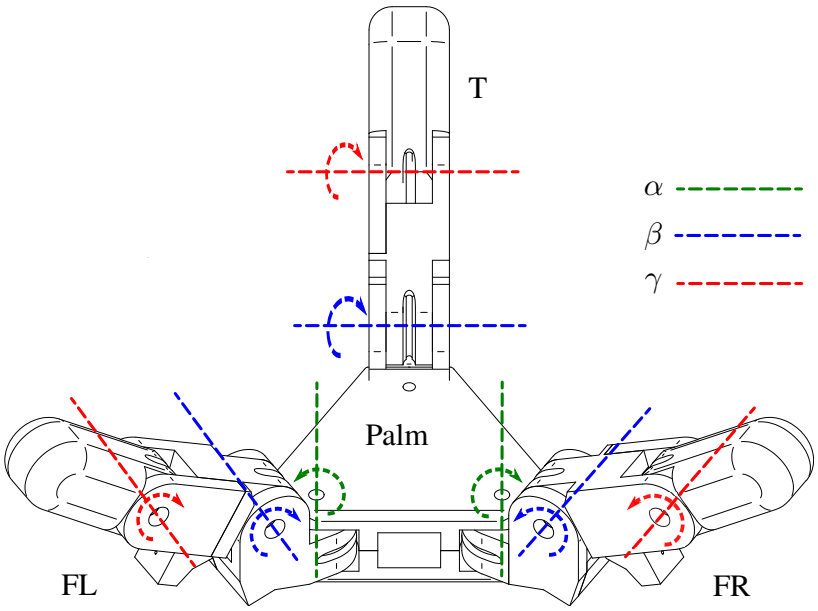
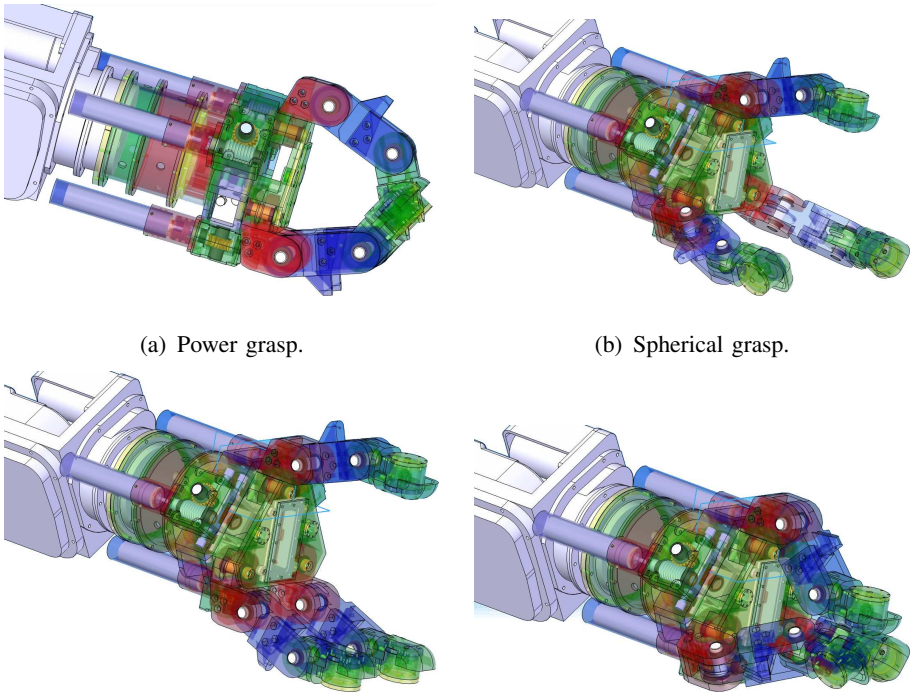


Fig. 3. Kinematic structure of the gripper.

fingers named FR and FL (right and left finger respectively). The thumb has two links only: the proximal link, connected to the palm by a revolute joint (proximal joint) with a rotational axis parallel to the palm plane, and a distal link connected to the proximal link by a revolute joint (distal joint) whose rotational axis is also parallel to the palm plane. The FR and FL fingers differ from the thumb by the connection of the finger to the palm: in this case, an additional joint (palm joint) with rotational axis perpendicular to the palm plane is introduced between the palm and the proximal link, allowing the rotation of the whole finger with respect to the palm axis. This arrangement allows performing both parallel and precision grasps, by means of opposition of the fingertips.

In total, the gripper has 8 joints, each one driven by an independent closed-loop cable actuation. Only 3 motors are used for the actuation, and couplings among the joints are implemented by connecting in parallel the cable driving system of the three joint groups (i.e. distal, proximal and palm joints) to the same motor. This solution represents a good trade-off between functionality and complexity, since it enables several grasp configurations on a great variety of objects, both in force and in form closure. Some significant finger postures and grasp configurations are reported in Fig. 4.



(a) Power grasp. (b) Spherical grasp.

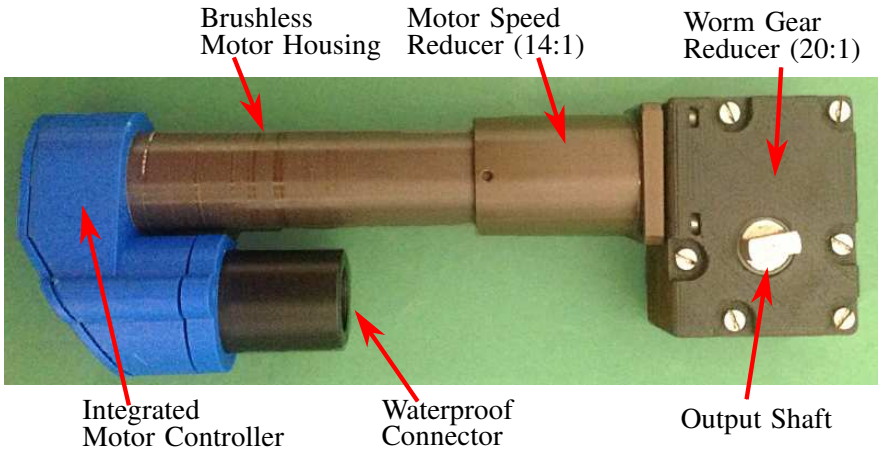
(c) Parallel grasp. (d) Tripod precision grasp.

Fig. 4. CAD view of the gripper design and kinematic configurations.

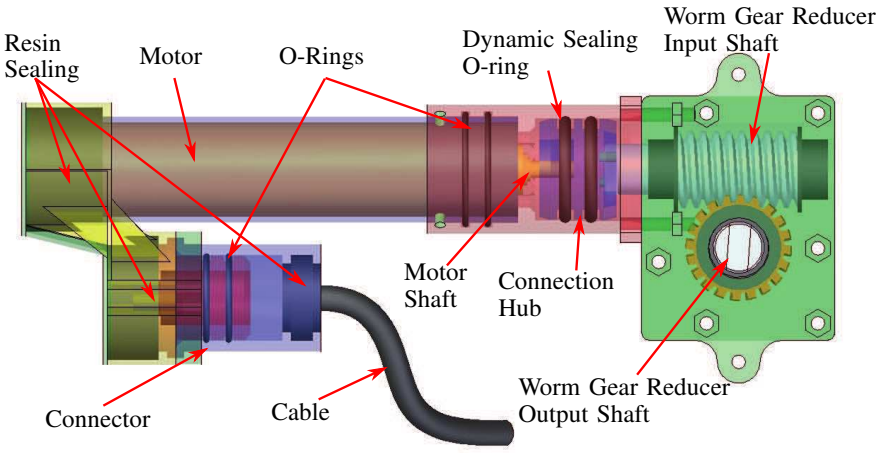
2) Actuation: While in the TRIDENT project the motors were enclosed into sealed boxes together with the two-stages speed reducers, in the MARIS project the actuator structure has been completely redesigned by encapsulating the motors inside a sealed tube that allows fast substitution of the motors without affecting the sealing of the other mechanical and electrical parts. A detail of the actuation module developed for the underwater gripper is shown in Fig. 5. The actuator housing is sealed in such a way to allow fast motor replacement in case of fault, as can be seen in Fig. 5(b).

In particular, the actuation system of the gripper is based on the Faulhaber 12 W brushless DC motor EN 2250 BX4 CCD with integrated motion controller and CAN interface. The closed-loop cable transmission of the gripper, whose details are visible in Fig. 6, implements a double-acting actuator [31]. This transmission system has been adopted mainly because it allows an optimal distribution of the weight and of the actuators. It also introduces some actuation compliance that is useful for safety reasons, in particular for operating in unstructured environments.

The maximum normal force applicable by each finger in continuous operation is about 150 N,



(a) Detail of the actuation module.



(b) Internal view of the actuation module.

Fig. 5. Detailed view of the gripper actuation module.

which can be considered satisfactory for the typical operations of the MARIS project. Moreover, thanks to the introduction of the worm gear reducer, the actuators are non-backdrivable. This feature allows holding of the desired gripper configuration without further supplying power to the motors even during a grasp.

3) Wrist Force/Torque Sensor: A F/T sensor [28], [29] has been designed and integrated into the gripper wrist interface for controlling the force and the stability of the manipulator interaction, see Fig. 7(a) and 7(b). The sensor detects the deformation of the wrist mechanical structure, allowing the estimation of both the forces and the torques applied to the gripper. In Fig. 8(a) and 8(b) the comparison between forces and torques estimated by the gripper F/T

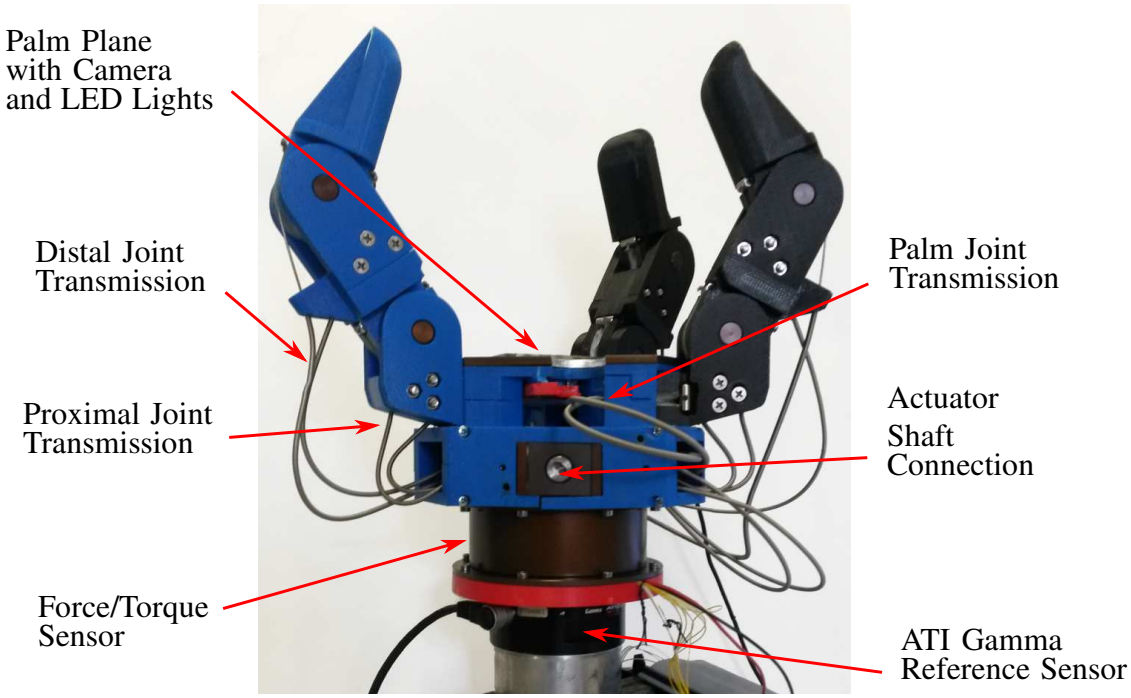


Fig. 6. The gripper mounted on the reference sensor during wrist force/torque sensor calibration: detailed view of the cable transmission system and main components.

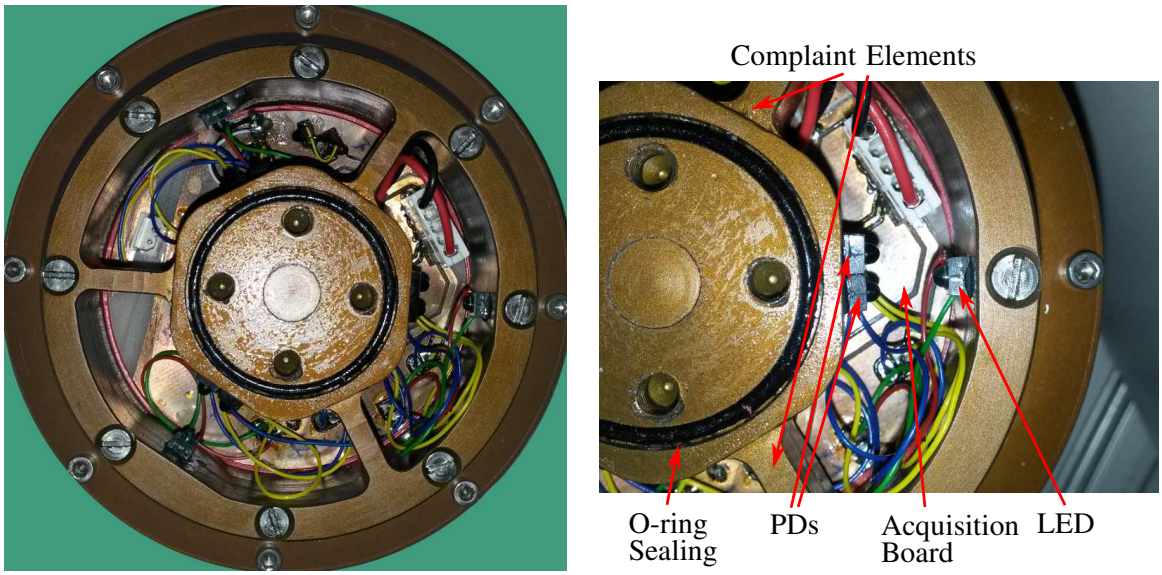
sensor and an ATI Gamma 10-130 F/T reference sensor is reported. The sensor communicates with the control system by means of the same CAN bus used for the arm and gripper motors for a simple integration with the arm/gripper system.

D. Integration of MARIS subsystems

In order to host all the subsystems needed to comply with the MARIS project requirements, the R2 ROV/AUV has been suitably adapted to mount the robotic arm and gripper system underneath the payload sled, the stereo camera system in front of the vehicle, and the related canisters for the subsystems control. The vehicle with the MARIS payload integrated is shown in Figure 9.

III. MARIS Control Framework

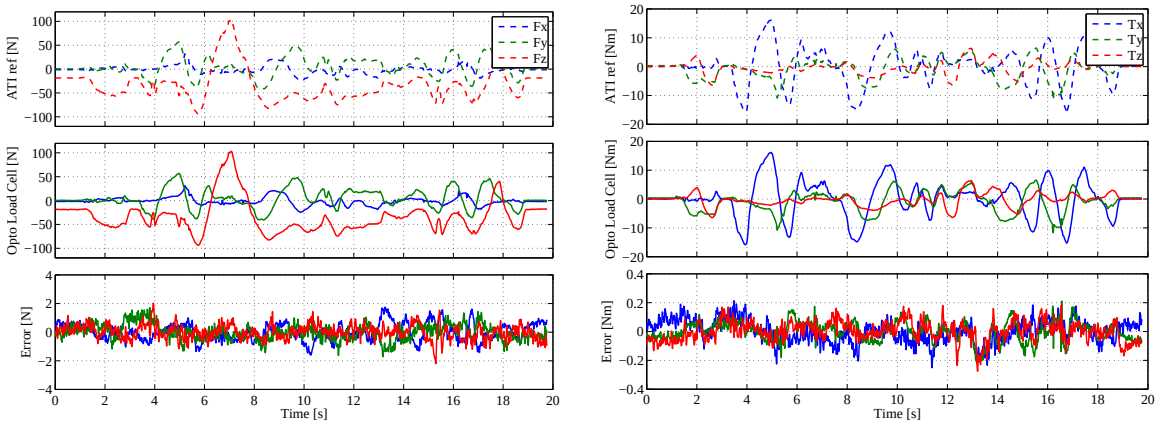
This section presents the control architecture developed for the MARIS system. The architecture treats the vehicle and the manipulators as a unique body and it is based on a modular,



(a) Detailed view of the wrist force/torque interior.

(b) Detailed view of the wrist force/torque sensor measuring elements.

Fig. 7. The wrist force/torque sensor.



(a) Comparison of the estimated force with the ATI reference sensor.

(b) Comparison of the estimated torque with the ATI reference sensor.

Fig. 8. Forces/torques measured by the fingertip sensor.

reconfigurable and hierarchical control architecture composed of four nested control levels (starting from the outermost to the innermost):

- Mission Supervisor
- Kinematic Control Layer (KCL)

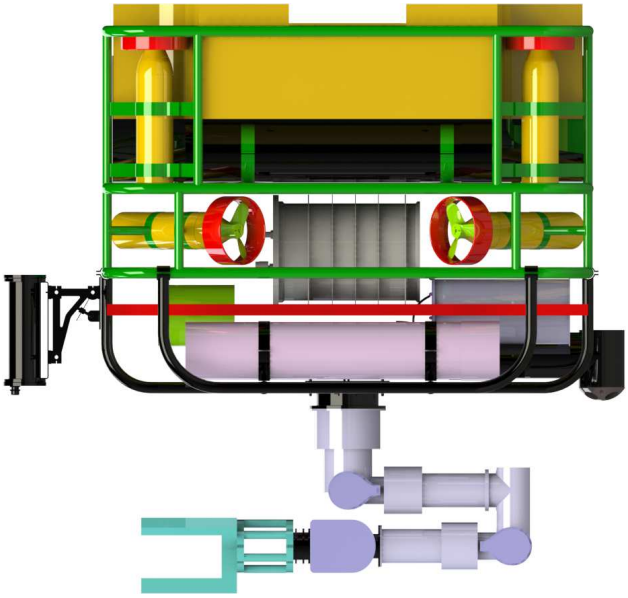


Fig. 9. The R2 ROV/AUV hosting the different MARIS subsystems: vision system (left), DVL (right), manipulator and a sketch of the hand (bottom).

- Dynamic Control Layer (DCL)
- Thruster Allocation

A. *Mission Supervisor*

The mission supervisor has the role of scheduling the *action* to be executed (see the definition in section III-B7). For the purposes of the MARIS experimental trials, the mission was defined as a sequence of three steps:

1) perform a survey until the object is detected (vision system triggers an event when the object is detected for the first time);
2) grasp the pipe (the hand motors current is monitored to detect a successful grasp);
3) move the arm in a predefined position to begin the transportation phase.

After such steps, the intervention was considered successful. Due to the limited size of the pool where the trials were held, there was not enough space for more complex missions, therefore this module was not developed further.

B. The Task Priority Control Approach to Kinematic Control

This section develops the main concepts behind the proposed task priority based kinematic control of the MARIS system. The task priority approach is a flexible, inclusive framework that allows to specify how the system should achieve certain *control objectives*, by defining what is called as a *control task*, which is the minimal building block of the architecture. A different *priority* can be assigned to each task. A hierarchy of prioritized control tasks is called an *action*, which defines a complex emergent behavior. Finally, actions are sequenced to fulfil some given high level goal (*mission*). These concepts are better developed and explained in the following sections.

1) Basic Definitions: The mathematical definitions are limited to what is strictly necessary for the explanation of the main concepts:

- The system configuration vector $\boldsymbol{c} \in \mathbb{R}^n$ of the UVMS as

$$\boldsymbol{c} \triangleq \begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{\eta} \end{bmatrix}, \tag{1}$$

  where $\boldsymbol{q} \in \mathbb{R}^l$ is the arm configuration vector and $\boldsymbol{\eta} \in \mathbb{R}^6$ is the vehicle *generalized coordinate position vector*. From the above definitions it results $n = l + 6$;

- The system velocity vector $\dot{\boldsymbol{y}} \in \mathbb{R}^n$ of the UVMS as

$$\dot{\boldsymbol{y}} \triangleq \begin{bmatrix} \dot{\boldsymbol{q}} \\ \boldsymbol{v} \end{bmatrix}, \tag{2}$$

  where $\dot{\boldsymbol{q}} \in \mathbb{R}^l$ are the joint velocities and $\boldsymbol{v} \in \mathbb{R}^6$ is the stacked vector of the vehicle linear and angular velocity vectors.

2) Control Objectives: With the above definitions, the concept of control objective can be introduced. A configuration dependent scalar variable $x(\boldsymbol{c})$ is said to correspond to a *scalar equality control objective* when it is required to satisfy:

$$x(\boldsymbol{c}) = x_0, \tag{3}$$

or to a *scalar inequality control objective* when it is required to satisfy

$$x(\boldsymbol{c}) \geq x_{min} \qquad \text{or} \qquad x(\boldsymbol{c}) \leq x_{max}, \tag{4}$$

where the min and max subscripts indicate a minimum and maximum value respectively.

The use of scalar objectives allows a simple discussion of the proposed approach. However, it does not constitute a limitation. Indeed, it is possible to define m scalar control objectives, each one corresponding to a single component of a vector $\boldsymbol{h} \in \mathbb{R}^m$. Then, the approach allows the definition of requirements for the whole vector, where some components are required to stay within some ranges, and others are required to attain a given value.

3) Reactive Control Tasks: Control objectives define what the system needs to do, but not how the system can accomplish them. This is where the concept of *reactive control task* comes into play. *A task is defined as tracking a given feedback reference rate $\dot{\tilde{x}}$, capable of driving the associated variable x toward the corresponding objective.*

For equality control objectives, a feedback reference rate $\dot{\tilde{x}}$ that drives x toward x_0 is

$$\dot{\tilde{x}} \triangleq \gamma(x_0 - x),\ \gamma > 0, \tag{5}$$

where γ is a positive gain to control the convergence speed. For inequality control objectives, a suitable feedback reference rate $\dot{\tilde{x}}$ is any rate that drives x toward any arbitrary point inside the region where the inequality is satisfied. For instance, consider an inequality objective of the type $x \leq x_{max}$ and consider a point x^* such that $x^* \leq x_{max}$, then a suitable feedback reference rate that drives x toward its corresponding objective is

$$\dot{\tilde{x}} \triangleq \gamma(x^* - x),\ \gamma > 0. \tag{6}$$

Note that the above are just simple examples of proportional control laws to achieve the desired objectives. More complex generation of reference feedback rates could be implemented if required.

The link between the system velocity vector $\dot{\boldsymbol{y}}$ and the considered control objective x is given by the Jacobian relationship

$$\dot{x} = \boldsymbol{g}^T(\boldsymbol{c})\dot{\boldsymbol{y}}, \tag{7}$$

where $\boldsymbol{g} \in \mathbb{R}^n$ is a vector.

4) Flexibility through Task Activation and Deactivation: An important feature of this architecture is the ability of activating and deactivating control tasks. To this purpose, an activation function is always associated to each control objective and its corresponding reactive control task. The activation function takes the following form

$$a(x) \triangleq a^i(x), \tag{8}$$

where $a^i(x) \in [0,1]$ is a function of the control objective variable $x(\boldsymbol{c})$, and its purpose is to deactivate the task whenever the inequality objective is satisfied, to avoid over-constraining the system.

The relationship between the value of the activation function and the control task is as follows:

- If $a(x) = 1$, the control task is called *active* and the corresponding actual $\dot{x}$ should therefore track $\dot{\bar{x}}$ as close as possible;
- If $a(x) = 0$, the control task is named *inactive* and the actual $\dot{x}$ should be unconstrained;
- If $0 < a(x) < 1$ the control task is named *in transition* and the actual $\dot{x}$ should smoothly evolve between the two previous cases.

Note that for equality control objectives it clearly holds that $a^i(x) \equiv 1$.

5) Non-reactive Control Tasks: Not all the control tasks are associated with a control objective. Indeed, some of them can be directly defined in a specific task velocity space. For example, the goal of minimizing vehicle motions is directly expressed in the velocity space of the vehicle, without the definition of a variable $x(\boldsymbol{c})$. For this reason, non-reactive tasks are always characterized by having $a(x) = a^i(x) \equiv 1$. From now on, the distinction between reactive and non-reactive control tasks will be dropped, and the generic term control task will be used, unless otherwise specified.

6) Task Priority Inverse Kinematics: Once the control objectives have been specified, their respective *priority* must be set. If multiple scalar control tasks are assigned to the same priority level k, they lead to a "multidimensional" task of the type

$$\dot{\boldsymbol{x}}_k = \boldsymbol{J}_k(\boldsymbol{c})\dot{\boldsymbol{y}}, \tag{9}$$

where now $\dot{\boldsymbol{x}}_k \in \mathbb{R}^m$ is the task velocity vector and $\boldsymbol{J}_k \in \mathbb{R}^{m \times n}$ is the Jacobian matrix of the task. At the same time, the task reference vector $\dot{\bar{\boldsymbol{x}}}_k$ is defined as the stacking of the reference rates of each scalar control task, while their activation functions (8) are organized in a diagonal matrix $\boldsymbol{A}_k$.

With the above definitions, the control problem becomes tracking the given reference velocities, following the required priority order, and taking into account their corresponding activation values. The solution of this problem can be found solving the following sequence of minimization problems:

$$S_k \triangleq \left\{ \arg\text{R-}\min_{\dot{\boldsymbol{y}} \in S_{k-1}} \left\| \boldsymbol{A}_k(\dot{\bar{\boldsymbol{x}}}_k - \boldsymbol{J}_k\dot{\boldsymbol{y}}) \right\|^2 \right\}, \quad k = 1, 2, \ldots, N, \tag{10}$$

where S_{k-1} is the manifold of solutions of all the previous tasks in the hierarchy, and $S_0 \triangleq \mathbb{R}^n$.

The solution of each problem in (10) is not straightforward. It is well known that minimization problems can be invariant to weights (such is the role of $\boldsymbol{A}_k$) [32], [33]. For this reason, the notation $\mathrm{R\text{-}min}$ has been used to highlight the fact that a special regularized pseudo inverse solution of that problem is employed, as defined in [22].

The methodology, termed *iCAT task priority framework*, results in the following Task Priority Inverse Kinematics (TPIK) algorithm:

$$\boldsymbol{\rho}_0 = \mathbf{0}, \quad \boldsymbol{Q}_0 = \boldsymbol{I}, \tag{11}$$

then for $k = 1, \ldots, N$

$$\begin{aligned} \boldsymbol{W}_k &= \boldsymbol{J}_k \boldsymbol{Q}_{k-1} (\boldsymbol{J}_k \boldsymbol{Q}_{k-1})^{\#, \boldsymbol{A}_k, \boldsymbol{Q}_{k-1}}, \\ \boldsymbol{Q}_k &= \boldsymbol{Q}_{k-1} (\boldsymbol{I} - (\boldsymbol{J}_k \boldsymbol{Q}_{k-1})^{\#, \boldsymbol{A}_k, \boldsymbol{I}} \boldsymbol{J}_k \boldsymbol{Q}_{k-1}), \\ \boldsymbol{\rho}_k &= \boldsymbol{\rho}_{k-1} + \mathrm{Sat}\left(\boldsymbol{Q}_{k-1} (\boldsymbol{J}_k \boldsymbol{Q}_{k-1})^{\#, \boldsymbol{A}_k, \boldsymbol{I}} \boldsymbol{W}_k \left(\dot{\boldsymbol{x}}_k - \boldsymbol{J}_k \boldsymbol{\rho}_{k-1} \right) \right), \end{aligned} \tag{12}$$

where the operator $\boldsymbol{X}^{\#, \boldsymbol{A}, \boldsymbol{Q}}$ is defined as in [22] and where the function Sat$(\cdot)$ implements the saturation proposed in [34].

7) Actions: An *action* is a prioritized list of control objectives and associated reactive tasks, with the possible addition of any non-reactive tasks, to be *concurrently* managed. For example, the grasping action implemented in the field trials that will be discussed later in the paper comprises the following list of control objectives (from highest to lowest priority):

1) Arm joint limits avoidance;
2) Arm manipulability;
3) Arm elbow-camera occlusion avoidance;
4) Camera centering;
5) End-effector linear position control;
6) End-effector angular position control;
7) Arm preferred shape.

Whenever the control system needs to transition from an action to another, the control objectives that are not anymore relevant need to be deactivated, while the new ones need to be activated. To this purpose, the activation function (8) is modified to become

$$a(x, \boldsymbol{p}) = a^i(x) a^p(\boldsymbol{p}), \tag{13}$$

where $a^p(\boldsymbol{p}) \in [0,1]$ is a continuous sigmoidal function of a vector of parameters $\boldsymbol{p}$ external to the control task itself. In particular, the $a^p(\boldsymbol{p})$ can be conveniently parametrized by the previous and current action being executed, and the time elapsed in the current action, to obtain the desired activation/deactivation smooth transition.

8) Control of Underactuated Vehicles: The proposed architecture easily deals with underactuated vehicles. To that aim, a non-reactive control task defined as follows is placed *at the highest priority*:

- The reference rate $\dot{\tilde{\boldsymbol{x}}}$ is composed by the vector of the *measured* vehicle velocities.
- The Jacobian is simply $\begin{bmatrix}\mathbf{0}_{6\times l} & \boldsymbol{I}_{6\times 6}\end{bmatrix}$.
- The activation matrix $\boldsymbol{A}$ diagonal elements are equal to one if they correspond to a non-actuated degrees of freedom, zero otherwise.

If such a task is placed at the *top of the hierarchy*, then:

- The velocity components of $\boldsymbol{\rho}_1$ will be constrained to be equal to the measured ones for non-actuated degrees of freedom.
- The matrix $\boldsymbol{Q}_1$ will prevent all the lower priority tasks from interferring with the highest priority one, effectively inhibiting them from changing the values of $\boldsymbol{\rho}_k$ in the components corresponding to non-actuated degrees of freedom.
- Each lower priority task k will take into account the current velocities of the non-actuated degrees of freedom, during the computation of their task references $\left(\dot{\tilde{\boldsymbol{x}}}_k - \boldsymbol{J}_k\boldsymbol{\rho}_{k-1}\right)$. Therefore, the final control vector is the *optimal* one considering the current *measured* velocities in the non-actuated degrees of freedom.

Before forwarding the reference vector to the DCL for its tracking, the values corresponding to non-actuated degrees of freedom are discarded.

9) Compensation of Vehicle Velocity Tracking Inaccuracies and Multi-rate Control: In the previous sections, the control of the UVMS has been presented in a whole-body manner, jointly considering the vehicle velocity and arm joint velocities as control variables in the stacked vector $\dot{\boldsymbol{y}}$. However, nonlinear properties of thrusters and their dynamic performances [35]–[37] are well known, and are considerably worse than arm motor ones. Such considerations lead to the conclusion that the vehicle velocity is tracked with far higher inaccuracy than the joint velocities.

To improve the overall control performances of the UVMS during floating operations, the idea is to have two parallel optimizations (12):

- The first one, called TPIK 1, exploits the mechanism presented in section III-B8 to deal with non-actuated vehicle degrees of freedom. Of the whole vector $\dot{\tilde{\boldsymbol{y}}}$, only the vehicle reference velocities $\bar{\boldsymbol{v}}$ are maintained, while the joint reference velocities are discarded;
- The second one, called TPIK 2, considers the vehicle as *fully non-actuated*. Therefore, the control task shown in section III-B8 is used with an identity activation matrix, and initialized with the vehicle measured velocities in all degrees of freedom. De-facto only the arm joint velocities are subject to optimization in TPIK 2. Of the whole vector $\dot{\tilde{\boldsymbol{y}}}$, only the joint reference velocities $\dot{\tilde{\boldsymbol{q}}}$ are maintained, which represent the *optimal* joint velocities in correspondence of the actual *measured* vehicle velocities.

The outputs of the two parallel optimization are then forwarded to the respective DCLs. With the proposed scheme, the arm and vehicle are controlled in a coordinated manner, however the arm joint velocities are always tuned to the current vehicle velocity. Therefore, the optimality of the arm motion is not affected by the inaccuracy of the vehicle velocity tracking.

Finally, the two parallel optimizations allow to easily tackle multi-rate control. TPIK 1, which generates the reference velocities for the vehicle, can run at the frequency allowed by the vehicle DCL. Conversely, TPIK 2, which generates the joint reference velocities, can run at the arm control frequency. Of course, the vehicle velocity feedback should be updated at the same frequency to maintain the optimality.

C. Dynamic Control Layer

The Dynamic Control Layer is responsible for arm and vehicle reference velocity tracking. For what concerns the MARIS vehicle's velocities regulation, it is composed by a set of PI (Proportional-Integral) regulators, with a feedforward input signal, that independently control the motion along each degree of freedom. For what concerns the manipulator, the DCL simply consists of the Faulhaber motor controllers embedded in the arm joints and their independent PI regulators.

D. Thruster Allocation

The R2 vehicle has a redundant thrust allocation scheme composed by 4 vertical and 4 horizontal thrusters. The vertical motion is achieved by applying the same thrust reference to all the vertical thrusters in order to command the desired heave velocity.

A more complex scheme is instead applied for the horizontal motion: thrusts have to be combined in order to properly generate surge, sway and yaw velocities at the same time. To achieve this goal a priority based thrust allocation scheme is employed in order to: i) firstly satisfy the torque request, i.e. always allowing the vehicle to turn and orientate itself tracking the desired references; ii) with the ”remaining” maneuverability space the module satisfies at best as possible (i.e. until thrusters saturation is reached) the linear velocity requests.

IV. MARIS Vision System

The goal of the MARIS vision system is the detection of target objects and the estimation of their 3D position and orientation. In particular, patternless cylindrical pipes have been chosen as target objects due to their relevance for grasping and transportation tasks of the underwater offshore industry. The detection of such objects must address several issues.

In underwater conditions, there are few reliable features that can be exploited in target detection. Color is one of the few distinctive features for submerged targets [38], [39]. Color restoration has been performed according to grey world hypothesis [40], which assumes that the average surface reflectance in a scene is achromatic. After this operation, a raw region of interest (ROI) can be identified based on color. Detection is enabled by searching in the ROI straight regular contours, which are distinctive of human made artifacts.

Moreover, long cylindrical objects like pipes are only partially observable during the manipulation, since the object is either partially occluded by the robot manipulator or quickly goes out of the camera field of view. Since the manipulator configuration is known, its pose is used to mask the projection of the robotic arm in the image planes of left and right cameras [41]. This arm removal prevents potential errors in both ROI search and line contour detection. Furthermore, the line contours are tracked in the image to make line estimation more robust.

Partial visibility highlights other issues in pose estimation. Manipulation and grasping tasks require accurate 3D pose estimation of target objects. Standard dense stereo processing does not

meet such a requirement due to the problematic matching of homologous points and keypoint features in underwater environment [39].

The adopted 3D pose estimation algorithm exploits the a priori information about object geometry and the strongest features of cylinder objects, namely their contours. In particular, the line contours in the image plane correspond to planes tangent to the targets in 3D space. The symmetry axis of the cylinder can be found through the intersection of these planes [39], [41].

However, the long line contours are not sufficient to correctly assess the reference frame on the target object due to the intrinsic symmetry of the cylinder. Therefore, additional data must be provided: the terminal parts of the object to place the reference frame origin, and the orientation of the frame around the symmetry axis.

The terminal of the cylinder is represented in the image by a short line similarly to cylinder long edges. The corresponding plane is used to find the reference frame origin, once properly translated to the center of the cylinder. Since the cylinder terminal may be intermittently observed and wrong pose estimation may occur, a tracking algorithm updates the value of the target reference frame and filters wrong detection outcomes.

The orientation of the reference frame around the cylinder symmetry axis is given by the vehicle inertial sensor. In particular, the z axis is aligned with the gravity direction. Although the object is axially symmetric and can be grasped from all the directions, the assessment of orientation is required for a consistent control of grasping operations.

The methodology applied to pipes could be extended to a more general class of targets under given hypotheses. In particular, the color of targets must be robustly detectable on the seabed under different lighting conditions, and their shape must be regular for extraction of stable and geometrically defined contours. Moreover, the object geometry must be known a priori in order to match the edges detected in the two stereo images with object parts in the 3D space. Object symmetry, if present, must be addressed to disambiguate among different candidate poses, as well as the conditions for pose estimation when targets are only partially visible. The proposed approach enables sufficiently accurate and robust perception for object manipulation tasks, even in the challenging and heterogeneous conditions of underwater environments. Other more general approaches have not proven to be sufficiently robust and reliable under different working conditions [39].

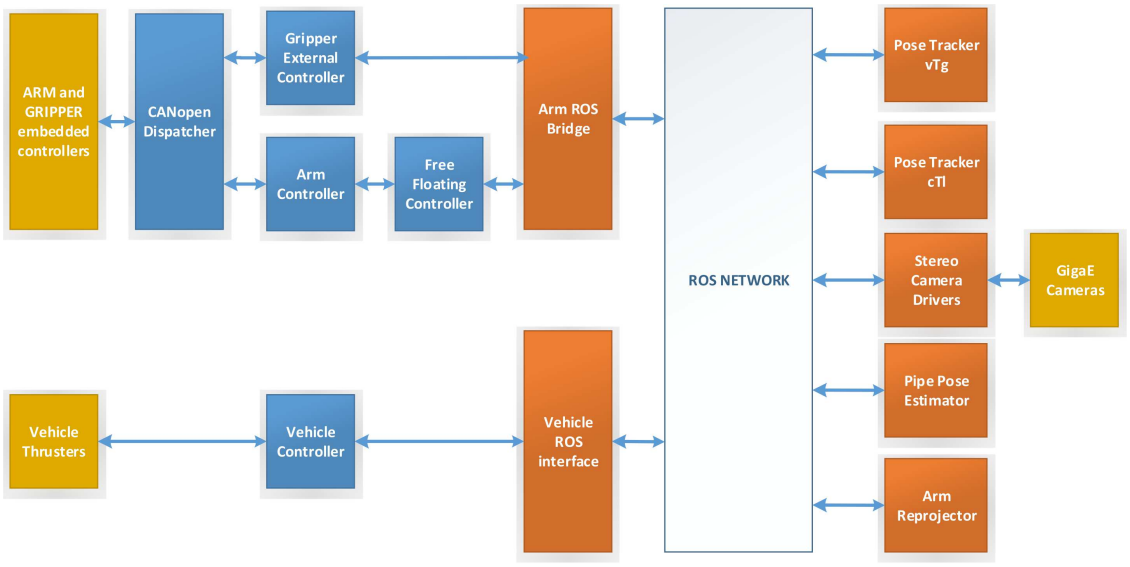


Fig. 10. The MARIS software architecture. Blocks colouring scheme is as follows: blue (RT processes), orange (ROS nodes), yellow (hardware).

V. MARIS Software Architecture

The MARIS software architecture is composed by a set of processes running on different CPUs and microcontrollers. To simplify the integration of all these processes, the project has adopted ROS as a baseline mechanism for interprocess communication (IPC), at least for the interfaces between subsystems. An exception to this rule is represented by real-time (RT) processes, since they rely on different IPC mechanisms such as shared memories, mailboxes and semaphores to maintain their RT status. In the following, we shall briefly analyze the set of processes that compose the MARIS architecture.

Starting from the top-left of Fig. 10, the lowest level is represented by *Gripper External Controller* and *Arm Controller*, which implement the CANopen protocol to interface themselves with the low-level embedded controllers of both the arm and the gripper. Since both controllers need to access the same CAN bus line, a *CANopen dispatcher* process is in charge of managing the CAN driver, receiving and dispatching messages to and from the CAN bus. Since all the above controllers are running as RT processes under RTAI, an *Arm ROS bridge* process is dedicated to managing the transfer of information between the RT processes and the ROS network.

A similar hierarchy is implemented for the control of the vehicle. The *Vehicle Controller*

implements the navigation system and the vehicle DCL, interacting with the hardware and implementing the thruster allocation. In particular, the navigation system is composed by two Kalman filters devoted to the estimation of yaw rate and linear velocities respectively, while other two Kalman filters are employed for the angular pose and linear position estimation. A ROS node called *Vehicle ROS Interface* is dedicated to bridge all the required data between the ROS network and the vehicle RT control processes.

The vision processes are natively written as ROS nodes, so in this case there is no need for a ROS bridge. The vision PC hosts the *Stereo Camera Drivers* processes, which acquire and publish the images from the cameras. The *Arm Reprojector* projects the robotic arm mask in images allowing the *Pipe Pose Estimator* to detect the pipe in the image and to estimate its pose without interference from the arm shape. The *Pose Tracker vTg* and *Pose Tracker cTl* track the desired grasp position (frame $\langle g \rangle$ w.r.t. vehicle frame $\langle v \rangle$) and of one end of the pipe (frame $\langle l \rangle$ w.r.t. the camera frame $\langle c \rangle$), which are used by the *Freefloating Controller* to drive the whole system on top of the pipe itself to perform the grasp.

Finally, the *Freefloating Controller* process is in charge of controlling the whole UVMS. The controller implements the task priority based kinematic inversion scheme presented in section III-B. The outputs of this controller are reference velocities for both the arm and the vehicle, which are sent to the *Arm Controller* and *Vehicle Controller*.

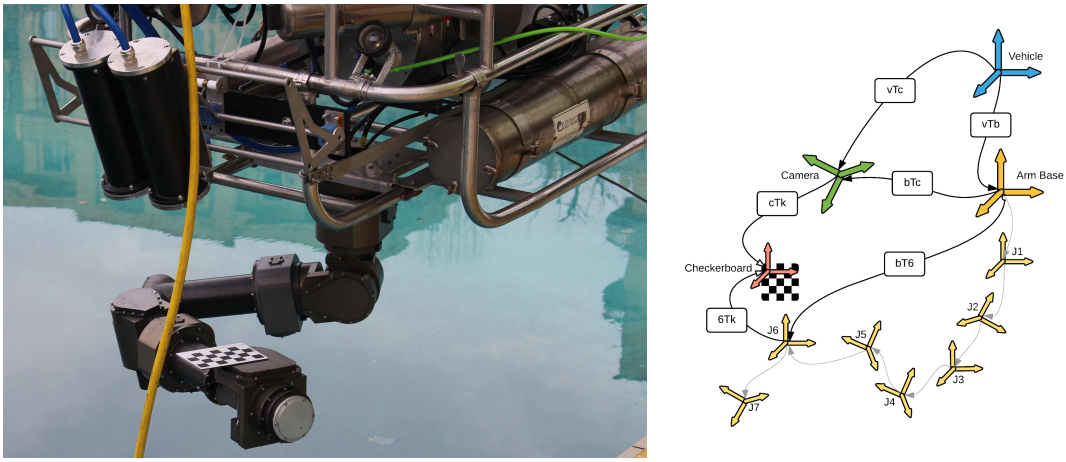
VI. Floating Underwater Manipulation Experimental Results

In this section, some of the experimental results of the MARIS project on floating underwater manipulation are presented. Three separate test campaigns were carried out, during October 2015, December 2015 and April 2016, all of them held in a public pool. In these experiments, a colored pipe was placed inside the pool, floating a few centimeters from the floor of the pool for safety reasons, to avoid collisions with the floor during early testing and tuning of the control parameters. The vehicle was umbilically connected to a ground station on the sideline of the pool, for supervision and quick access to the telemetry for debugging purposes. The results are summarized, in terms of grasp success rate, in Table I. Before commenting each of the test campaigns, the theory-to-practice transition is discussed.

TABLE I

MARIS EXPERIMENTAL RESULTS

Test Campaign	Attempts	Successful Grasps	Success Rate	DVL integrated
October 2015	16	4	25%	NO
December 2015	13	4	30.7%	NO
April 2016	17	12	70.5%	YES



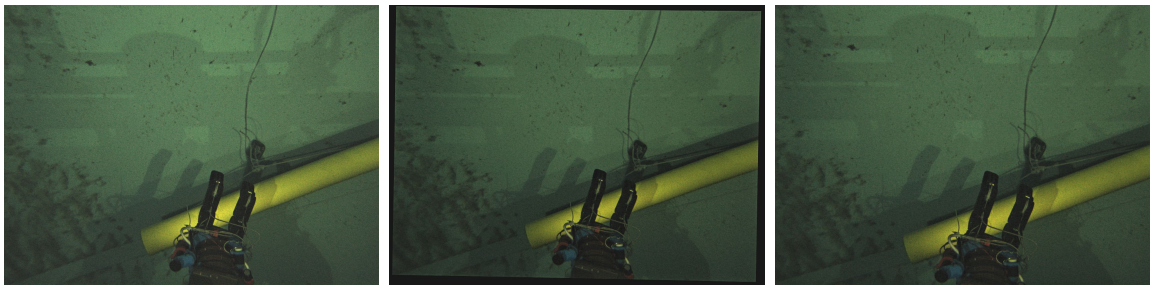
(a) (b)

Fig. 11. (a) Checkerboard attached to the manipulator wrist and used for intrinsic, relative extrinsic and system extrinsic parameters of stereo camera; (b) reference frames of AUV system involved in estimation of system extrinsic parameters.

A. From Theory to Practice

Before the execution of the actual experiments on floating manipulation, a few preliminary integration and tuning steps were necessary. Indeed, the deployment of the proposed stereo vision system required camera calibration and parameter tuning of the object detection algorithm. Finally, the development and integration of the custom MARIS sled required the tuning of the vehicle dynamic response.

1) Camera Calibration: Calibration of the vision system is a fundamental step of the integration phase to enable reliable execution of manipulation and grasping tasks. The output of the calibration phase includes different sets of parameters: the *intrinsic* parameters, the *relative extrinsic* and the *system extrinsic*. Intrinsic parameters are related to camera perspective geometry



(a) (b) (c)

Fig. 12. Rectification of underwater left frame using intrinsic parameters: (a) original image, rectified images with intrinsic parameters from (b) in-air calibration and (c) in-water calibration.

and optical distortion, whereas relative extrinsic ones represent the relative pose between left and right cameras in a stereo rig. Intrinsic and relative extrinsic calibration affect the accuracy of the target object pose w.r.t. the camera. System extrinsic parameters encode the relative pose between the stereo vision and the manipulator reference frames, and their accurate estimation relies on the accuracy of the arm kinematics.

For a pinhole camera model, the estimation of intrinsic and relative extrinsic parameters is a standard procedure, usually performed by observing a checkerboard at slightly different poses. An underwater camera is more accurately modeled as an axial camera [42] due to light refraction through different media (water, plexiglass canister surface, air inside the canister). However, since stereo processing with axial camera is unpractical, it is common practice in underwater computer vision to adopt a pinhole model with the parameters obtained by an in-water calibration using a checkerboard. A less accurate alternative is to calibrate the stereo system before immersion and to optimize parameter values (in particular radial distortion) using ad hoc correction techniques [24]. Other approaches are based on local approximation of axial cameras with the pinhole model [43].

The intrinsic and relative extrinsic calibration parameters used in the experiments reported in this paper have been estimated according to the difficult and time-demanding in-water procedure of the pinhole model. The checkerboard has been attached to the robotic arm as shown in Figure 11(a) and moved in front of the camera until convergence to stable parameter values. The standard stereo calibration tool provided by the ROS framework has been used to compute the required parameters.

Figure 12 shows the rectification of a frame acquired in underwater (Figure 12(a)) using parameters of in-air (Figure 12(b)) and in-water calibration (Figure 12(c)). Several details like the size of the pipe after rectification illustrate the impact of accurate calibration on the images used to estimate target object pose.

Another calibration is needed to assess the position and orientation of the stereo camera w.r.t. the vehicle and the robotic arm frames, and it is represented by six system extrinsic parameters. Such a reference frame transformation is required to plan the robot motion and to grasp the target object. System extrinsic calibration has been performed using the same checkerboard rigidly attached to the manipulator (Figure 11(a)). Since the size of checkerboard squares is known, the checkerboard pose w.r.t. the left camera frame can be estimated by standard software tools. The position and orientation of checkerboard frame w.r.t. the closest reference frame attached to the manipulator, in particular the frame related to joint $J6$, has been manually assessed. Figure 11(b) illustrates the relevant frames of MARIS system: stereo camera (c), vehicle (v), manipulator base (b) and joints ($1 \dots 6$), and the checkerboard (k). Thus, the pose of stereo camera w.r.t. the manipulator base has been computed using the transformation of the checkerboard w.r.t. the camera (cTk), of the checkerboard w.r.t. manipulator joint 6 ($6Tk$) and of the joint 6 w.r.t. the manipulator base ($bT6$).

2) Tuning Vision System Parameters: Beside calibration parameters, there are other variables to be set in order to configure the vision system. The algorithm described in section IV requires knowledge of the cylindrical target object size and color. The length and radius of the pipes used in the experiments are about 1 m and 5 cm, respectively. Although different colors have been used in other experimental sessions, yellow pipes have been mainly used as target. The object detection algorithm finds an initial ROI through selection in HSV color space (e.g. hue $22 - 35$, minimum saturation 63, minimum value 30). Color criterion has proven reasonably robust to changing light conditions. The parameters that required more frequent adaptation are camera exposure time and gain. Tests have taken place in shallow water (from 3 m to 4 m depth) where the observation of target object is more sensitive to the amount of natural light. Once such camera parameters have been set, the vision system properly detects the target.

Figure 13 shows a representative sample of operating conditions during experiments. Figures 13(a)-(b) show pictures of the yellow pipe in clear water respectively in daylight and night conditions. The contribution of AUV headlights to the scene luminance is important only during

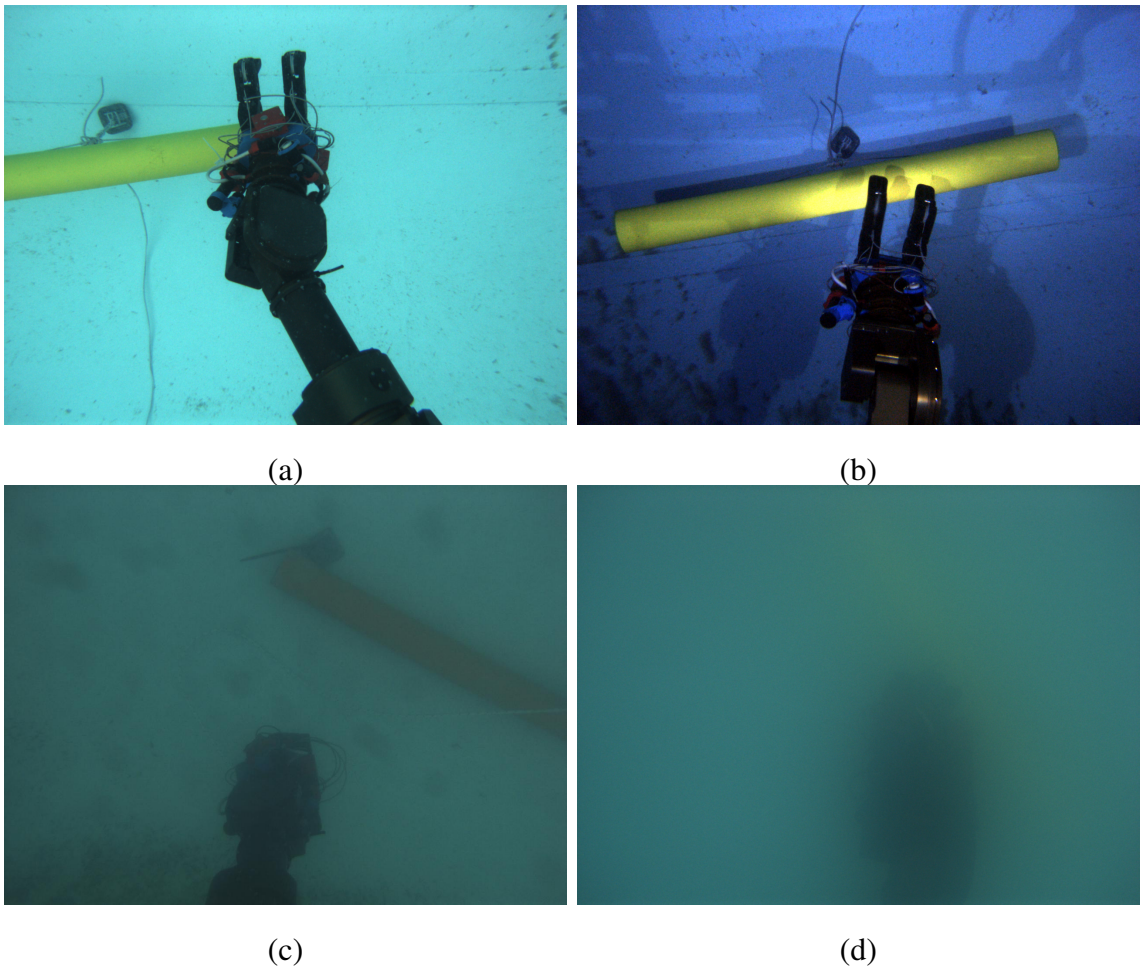


Fig. 13. Example of AUV camera views of cylindrical target objects in different light and water conditions: (a) day-light clear water; (b) night-vision clear waters and artificial illumination by the AUV; (c) slighly turbid water; (d) very turbid water.

night-time experiments, whereas in presence of natural light it is negligible. In the latter case, the value of camera exposure is smaller than in the night-condition case, but the algorithm parameters have not been changed. Grasping experiments have been successfully executed in both conditions. Figures 13(c)-(d) illustrate operations in turbid waters. In the first of these cases, the vision system provides intermittent detection. With extremely turbid waters like in Figure 13(d) the detection algorithm is completely unable to operate.

3) Tuning of Dynamic Responses: After the integration of the MARIS payload, the vehicle DCL gains had to be adjusted to the new dynamic model parameters. A preliminary fine-tuning phase has been carried out during the field trials, where an on-line observation of the controller

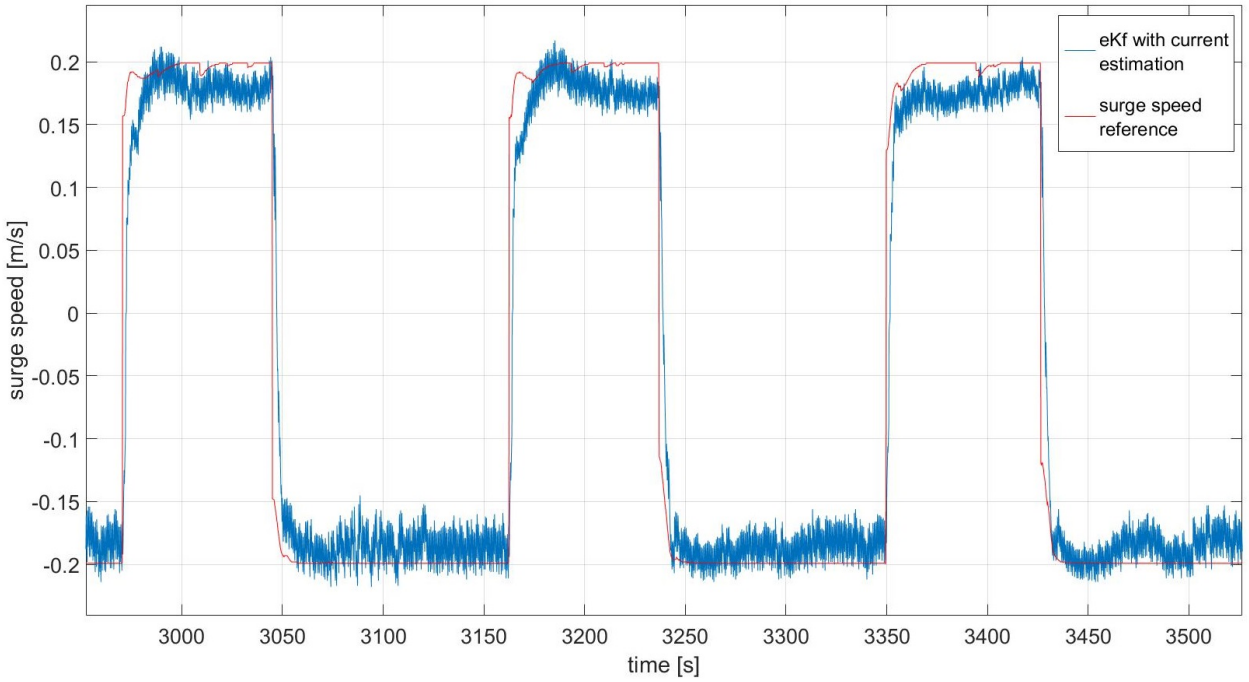


Fig. 14. Surge speed controller response with corrected (sea current estimation) feedback signal.

response has been used to correct the gain values to achieve the desired motion response. The response of the controller is reported in Figure 14.

4) Control Tasks Implementation: Another step during the transition from theory to practice is the implementation of the control tasks. To give some practical insights on this point, let us consider the grasping action presented in section III-B7, and let us focus on the arm manipulability control objective. In this case, the scalar variable $x(\boldsymbol{c})$ that represents the control objective is

$$x(\boldsymbol{c}) = \mu(\boldsymbol{q}) \triangleq \sqrt{\det\left[\boldsymbol{J}_{ee}(\boldsymbol{q})\boldsymbol{J}_{ee}^{T}(\boldsymbol{q})\right]}. \tag{14}$$

The manipulability measure $\mu(\boldsymbol{q})$ [44] is a continuous quantity that represents the distance from singular postures of the end-effector Jacobian matrix $\boldsymbol{J}_{ee}(\boldsymbol{q})$. Therefore, to avoid incurring into the problems related to the kinematic singularities, the manipulability measure should be maintained above a minimum threshold:

$$\mu(\boldsymbol{q}) > \mu_{min}. \tag{15}$$

Once the control objective has been defined, three quantities must be defined: The activation

function, the task Jacobian and the reference rate. From a practical point of view, this can be done as follows:

- The activation function $a^i(x) = a^i(\mu)$ is defined as follows:

$$a^i(\mu) \triangleq \begin{cases} 1, & \mu(\boldsymbol{q}) < \mu_{min} \\ s(\mu), & \mu_{min} \leq \mu(\boldsymbol{q}) \leq \mu_{min} + \Delta \\ 0, & \mu(\boldsymbol{q}) > \mu_{min} + \Delta \end{cases} \tag{16}$$

  where $s(\mu)$ is any sigmoid function joining the two extrema with continuity. The value Δ represents the width of the transition zone between the complete activation and deactivation of the control task. A deeper discussion on its choice and influence is given in [22].

- The feedback reference rate (6) becomes

$$\dot{\bar{x}}_\mu(\boldsymbol{q}) \triangleq \gamma(\mu_{min} + \Delta - \mu(\boldsymbol{q})),\ \gamma > 0, \tag{17}$$

  where the term $\mu_{min} + \Delta$ represents an arbitrary point inside the region where the inequality is satisfied, as mentioned in (6).

- Finally, the task Jacobian can be partitioned as

$$\boldsymbol{J}_\mu(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{J}_\mu^a(\boldsymbol{q}) & \mathbf{0}_{1\times 6} \end{bmatrix}, \tag{18}$$

  where $\boldsymbol{J}_\mu^a(\boldsymbol{q}) \in \mathbb{R}^{1\times l}$ is the Jacobian of the manipulability measure w.r.t. the joint velocities only, evaluated in real time via the procedure developed in [45].

B. October 2015 Campaign Results

The first test campaign was carried out during October 2015. Unfortunately, the visibility conditions in the pool were pretty bad, as can be seen in Fig. 13(d).

Notwithstanding the poor environmental conditions, tests were performed to assess the correct integration of all the software components, and to stabilize the feedback coming from the vision system. Initial tests were carried out while operating out of water. Figure 15 shows the sequence of a grasping trial. Successively, some tests were performed in water, but given the bad visibility, the vehicle base was not commanded by the control algorithm. Instead, the vehicle was manually moved on top of the pipe and only the manipulator was controlled to perform the grasp. As expected, the bad visibility conditions led to the unsatisfactory results reported in Table I.

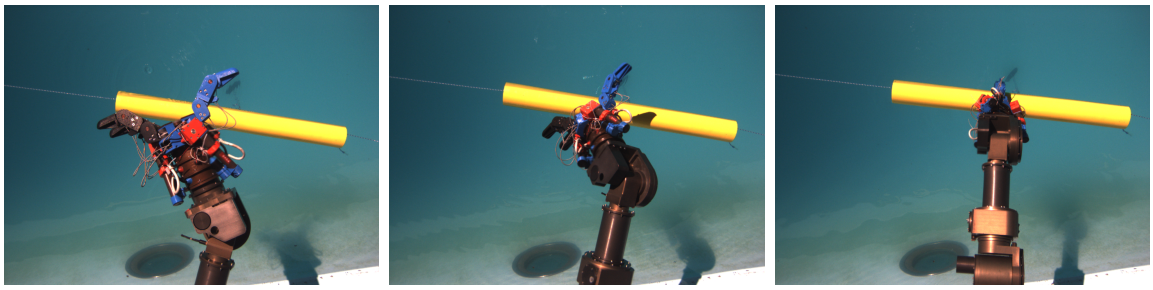


Fig. 15. Sequence of a successfully grasping performed out of water.

C. December 2015 Campaign Results

A second set of trials was scheduled for December 2015, shortly after the October ones. To avoid the problems of the first campaign, the pool was cleaned a week in advance, and the visibility conditions were much better.

The campaign was mostly dedicated to the tuning of the vision algorithm. Figure 16 shows one of the trials, performed late in the afternoon with low light conditions. In particular, the figure highlights the grasping sequence as seen from the left and right cameras of the vision system. The images show the role of the *Arm Reprojector* process, which allowed the *Pipe Pose Estimator* to have a very robust output by eliminating the interference created by the arm occlusions on the pipe.

A few tests were performed with the coordinated control of both the vehicle and the manipulator. Videos recorded from the on-board cameras are available at the following links: https://youtu.be/b7lytrMOMeQ and https://youtu.be/p0ZG-m4ZJl4. However, the success rate was still low, due to the quite significant difference between the commanded and actual vehicle velocities, especially exacerbated due to the poor performance of the thrusters near zero velocity. The inaccuracy of the vehicle reflected into an end-effector velocity not corresponding to the desired one and therefore it often resulted in a failure of convergence towards the desired grasping pose. Note that since the DVL was not yet integrated, it was impossible to apply the compensation technique proposed in section III-B9.

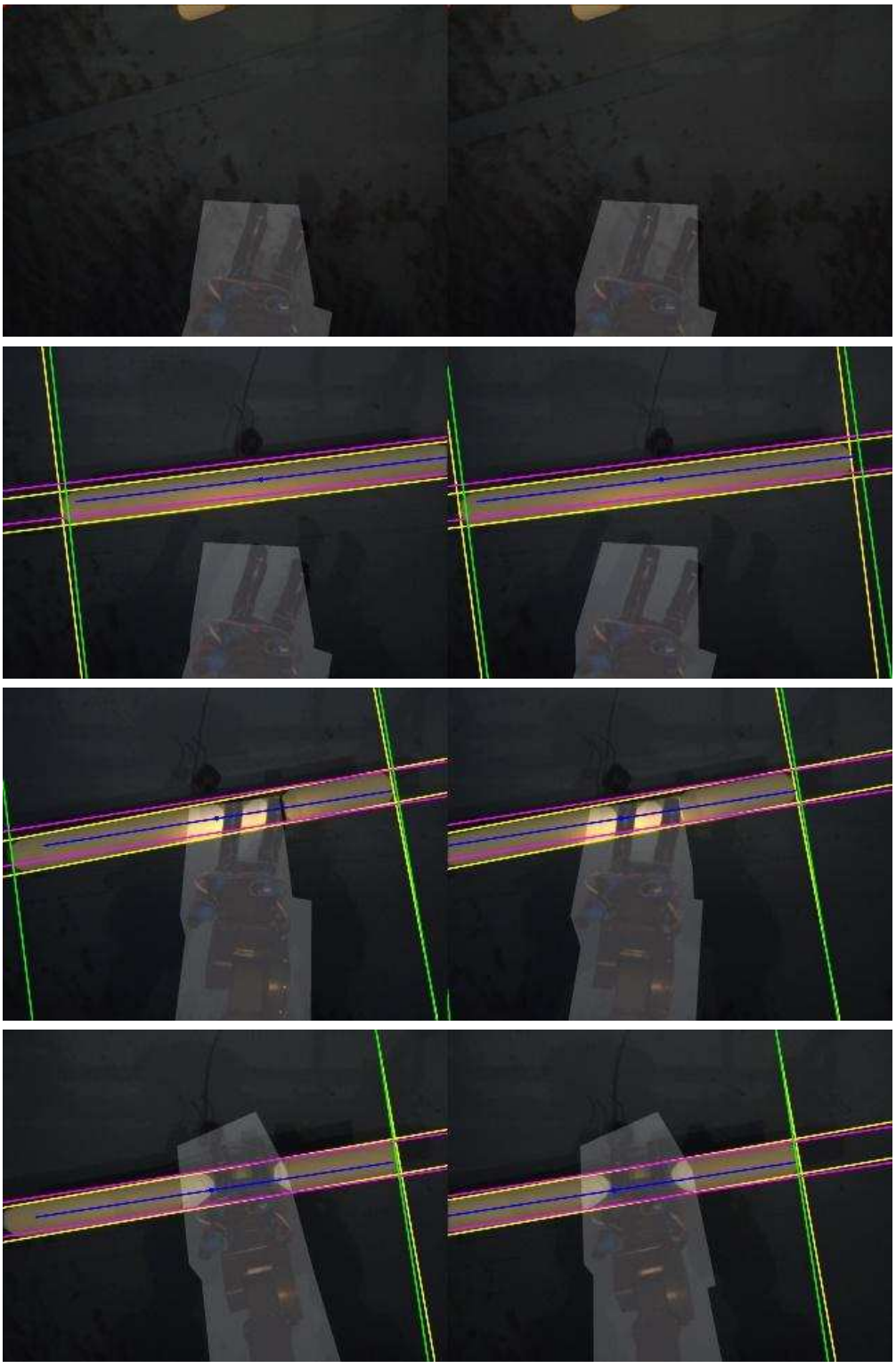
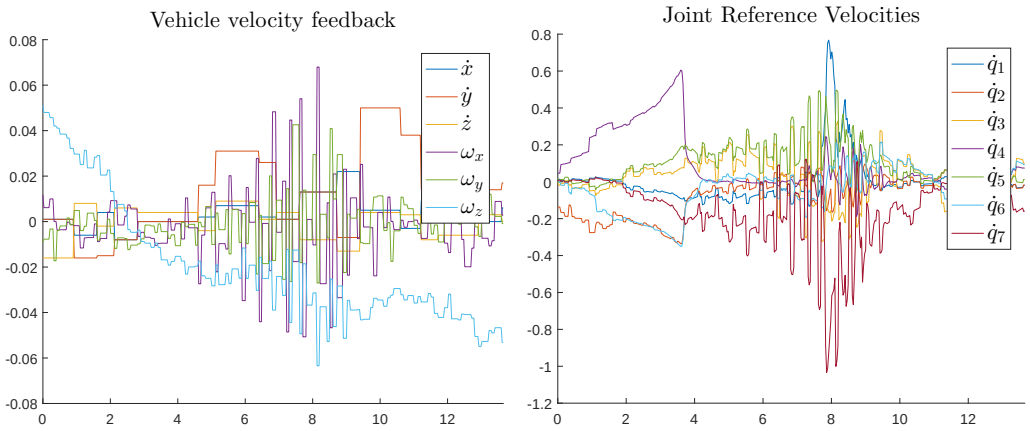


Fig. 16. Successful pipe grasping sequence in night-time conditions observed from the left and right cameras of the vision system. The images show the mask of the robotic arm reprojected in each frame, the yellow lines representing the current measurement of pipe line contours, whereas the purple and green lines are estimated by the edge trackers.



(a) (b)

Fig. 17. April 2016 MARIS test with unfiltered noisy vehicle feedback leads to oscillating behavior of the manipulator: (a) vehicle feedback, (b) joint reference velocities.

D. April 2016 Campaign Results

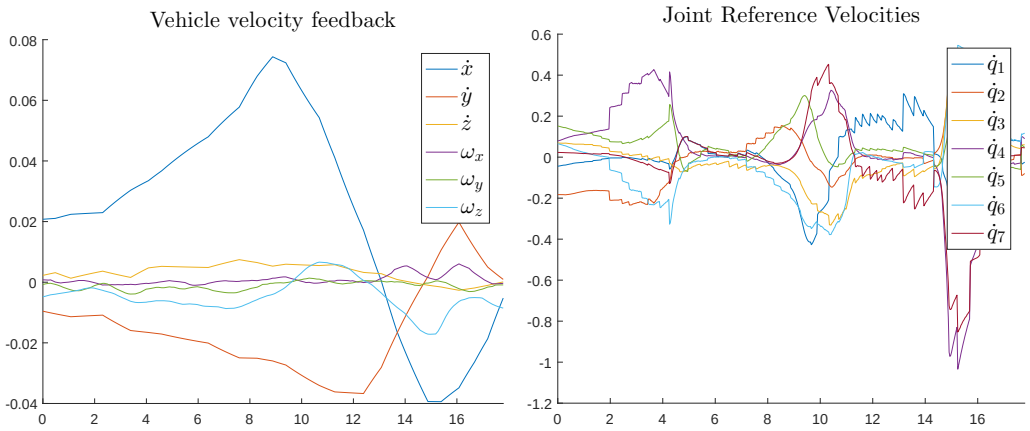
The final test campaign was held in April 2016. The visibility conditions of the pool were similar to those of the December 2015 tests. The major improvement with respect to the previous trials was the integration of the DVL in the R2 ROV, which allowed the use of the compensation technique presented in section III-B9.

An important lesson learnt during these trials was that trying to use the technique proposed in section III-B9 can have a detrimental effect on the performance of the system if the sensor feedback is particularly noisy. In fact, Figure 17 shows the vehicle feedback data from one of the trials and reports the generated joint reference velocities. Indeed, as can be seen from the plots, the manipulator, while trying to compensate what in reality was sensor noise, was inducing oscillations on the vehicle, further exacerbating the issue.

A simple, but necessary fix was to filter the high frequency noise, especially present in the angular rate feedback. With the introduction of a simple first order filter, with cutoff frequency of $\omega = 50$ rad/s, the self-induced oscillations disappeared, as can be seen from Fig. 18.

Thanks to the integration of the DVL, the filter to reduce the noise, and the implementation of the compensation technique, the success rate of the grasping operation increased to 70%, basically doubling w.r.t. the previous test campaign.

Figure 19 shows a successful grasp during the April 2016 trials, recorded by an external



(a) (b)

Fig. 18. April test with vehicle feedback filtered at $\omega = 50$ rad/s leads to much smoother manipulator behavior and increased performances thanks to the vehicle velocity compensation technique: (a) vehicle feedback velocity (filtered), (b) joint reference velocities.

camera placed on the pool floor. The full sequence can be seen at the following link: https://youtu.be/b3jJZUoeFTo.

VII. Conclusions

This work has presented the results of the MARIS project, including the developed control framework, the overall mechatronic integration, and the project's final experiments in underwater floating manipulation.

The system is composed by the CNR-ISSIA R2 ROV, which has been reconfigured and updated to work as an AUV, and a commercial manipulator manufactured by Graal Tech, whose control system has been entirely developed by ISME. Finally, the gripper (ISME Bologna node) and the vision system (University of Parma) have been developed during the course of the project.

The whole system has been tested in water tank experiments, in different light and water conditions. A video of one of these trials, with a successful grasp of the pipe is available at the link https://youtu.be/b3jJZUoeFTo. The final test campaign achieved approximately 70% success rate in grasping the target, with 12 successful grasps in 17 attempts. The results of MARIS project are a considerable improvement in robustness over the previous milestone of the TRIDENT project, both in terms of number of successful attempts as well as in the ability to carry out the task in different light conditions.

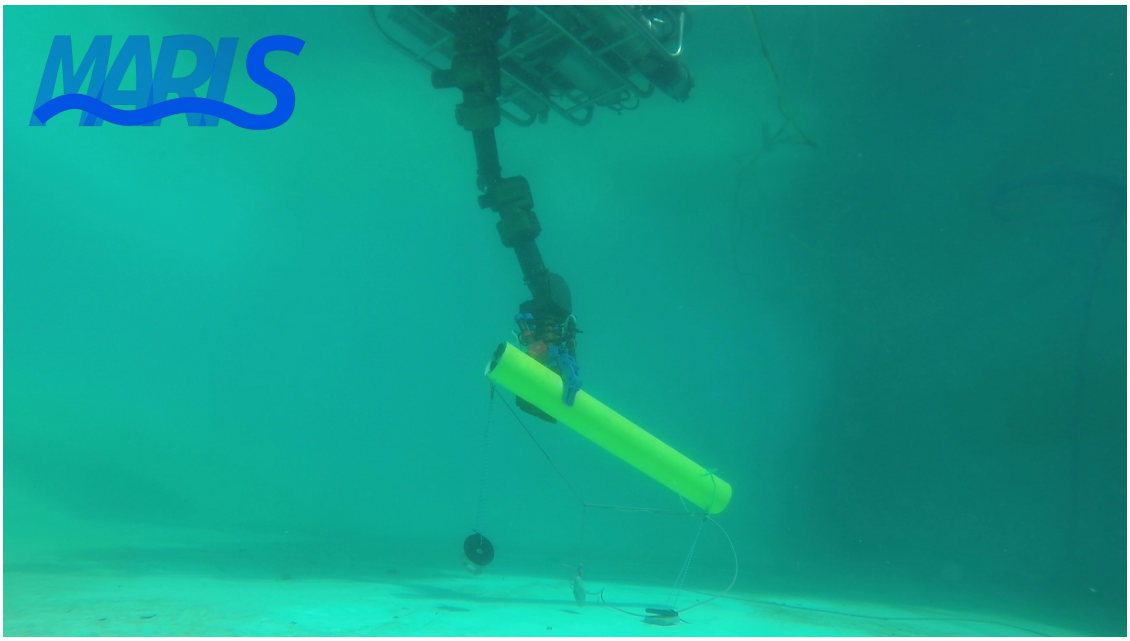


Fig. 19. Another successful grasp with different light conditions. This snapshot is extract from the video whose link is given in Section VII.

Nowadays, some of the authors are involved in an on-going Horizon 2020 project called DexROV [46], which focuses on increasing the autonomy of ROV operations to allow a supervised teleoperation from remote sites. The MARIS control framework is being currently integrated with a cognitive engine to cope with the latency problems of remote teleoperation. Furthermore, the same control framework is also further developed within the ROBUST project [47], where an UVMS system is employed for deep-water mining sites exploration.

As part of future works, the integration between the developed control framework and motion planning will be investigated. The idea is to let the motion planning focus on the generation of Cartesian trajectories for the end-effectors or for the object, without planning in the configuration space as instead it was done in MERBOTS. Thanks to the reactive capabilities of the developed control framework, the planner will not have to deal with low level details, reducing its computational time requirements and simplifying the planning-control integration.

Finally, within the MARIS project, theoretical studies on interaction control [48], multi-vehicle localization [49], communications [50], dual arm UVMS [51] (https://youtu.be/wBiOCVoRM9g) and cooperative underwater manipulation systems [52], [53] (https://youtu.be/9WRRUotcjmM)

have been performed, and represent the next step that has to be demonstrated in field trials.

ACKNOWLEDGMENTS

This work was supported by the MIUR (Italian Ministry of Education, University and Research) within the MARIS project, protocol number 2010FBLHRJ.

The authors would like to thank Prof. Giuseppe Casalino for the ideation of the project, for writing down the proposal and for coordinating the project, CNR-ISSIA Director Massimo Caccia for his initial effort in ideating the project together with Prof. Casalino, Proff. Stefano Caselli and Claudio Melchiorri for coordinating their research units (Parma and Bologna respectively). Furthermore, we would like to thank Andrea Ranieri, Enrica Zereik, Fabio Oleari for their contributions during the experimental trials.

REFERENCES

[1] P. Abreu, G. Antonelli, F. Arrichiello, A. Caffaz, A. Caiti, G. Casalino, N. C. Volpi, I. B. de Jong, D. De Palma, H. Duarte, J. P. Gomes, J. Grimsdale, G. Indiveri, S. Jesus, K. Kebkal, E. Kelholt, A. Pascoal, D. Polani, L. Pollini, E. Simetti, and A. Turetta, "Widely scalable mobile underwater sonar technology: An overview of the H2020 WiMUST project," *Marine Technology Society Journal*, vol. 50, no. 4, pp. 42–53, jul 2016. doi: 10.4031/mtsj.50.4.3

[2] T. Urabe, T. Ura, T. Tsujimoto, and H. Hotta, "Next-generation technology for ocean resources exploration (zipangu-in-the-ocean) project in japan," in *OCEANS 2015 - Genova*, IEEE. IEEE, may 2015, pp. 1–5. doi: 10.1109/oceans-genova.2015.7271762

[3] P. Drap, J. Seinturier, G. Conte, A. Caiti, D. Scaradozzi, S. Zanoli, and P. Gambogi, "Underwater cartography for archaeology in the venus project," *Geomatica: The J. of Geospatial Information Science, Technology, and Practice*, vol. 62, no. 4, pp. 419–428, 2008.

[4] A. Birk, A. Pascoal, G. Antonelli, A. Caiti, G. Casalino, G. Indiveri, and A. Caffaz, "Cooperative cognitive control for autonomous underwater vehicles (CO3AUVs): overview and progresses in the 3rd project year," in *Navigation, Guidance and Control of Underwater Vehicles*, vol. 3, no. 1, 2012, pp. 361–366.

[5] B. Gilmour, G. Niccum, and T. O'Donnell, "Field resident AUV systems – chevron's long-term goal for AUV development," in *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*, IEEE. IEEE, sep 2012, pp. 1–5. doi: 10.1109/AUV.2012.6380718

[6] D. R. Yoerger, H. Schempf, and D. M. DiPietro, "Design and performance evaluation of an actively compliant underwater manipulator for full-ocean depth," *Journal of robotic systems*, vol. 8, no. 3, pp. 371–392, 1991. doi: 10.1002/rob.4620080306

[7] H. Schempf and D. Yoerger, "Coordinated vehicle/manipulator design and control issues for underwater telemanipulation," in *IFAC Control Applications in Marine Systems (CAMS 92)*, Genova, Italy, April 1992.

[8] D. M. Lane, J. B. C. Davies, G. Casalino, G. Bartolini, G. Cannata, G. Veruggio, M. Canals, C. Smith, D. J. O'Brien, M. Pickett, G. Robinson, D. Jones, E. Scott, A. Ferrara, D. Angelleti, M. Coccoli, R. Bono, P. Virgili, R. Pallas, and E. Gracia, "Amadeus: advanced manipulation for deep underwater sampling," *IEEE Robot Autom Mag*, vol. 4, no. 4, pp. 34–45, 1997. doi: 10.1109/100.637804

[9] V. Rigaud, È. Coste-Manière, M.-J. Aldon, P. Probert, M. Perrier, P. Rives, D. Simon, D. Lang, J. Kiener, A. Casal *et al.*, "Union: underwater intelligent operation and navigation," *Robotics & Automation Magazine, IEEE*, vol. 5, no. 1, pp. 25–35, 1998. doi: 10.1109/100.667323

[10] P. Marty *et al.*, "Alive: An autonomous light intervention vehicle," in *Advances In Technology For Underwater Vehicles Conference, Oceanology International*, vol. 2004, 2004.

[11] J. Yuh, S. Choi, C. Ikehara, G. Kim, G. McMurty, M. Ghasemi-Nejhad, N. Sarkar, and K. Sugihara, "Design of a semi-autonomous underwater vehicle for intervention missions (SAUVIM)," in *Underwater Technology, 1998. Proceedings of the 1998 International Symposium on*. Tokyo, Japan: IEEE, 1998, pp. 63–68. doi: 10.1109/UT.1998.670059

[12] G. Marani, S. K. Choi, and J. Yuh, "Underwater autonomous manipulation for intervention missions AUVs," *Ocean Engineering*, vol. 36, pp. 15–23, 2008. doi: 10.1016/j.oceaneng.2008.08.007

[13] P. J. Sanz, M. Prats, P. Ridao, D. Ribas, G. Oliver, and A. Ortiz, "Recent progress in the rauvi project: A reconfigurable autonomous underwater vehicle for intervention," in *Elmar, 2010 Proceedings*. IEEE, 2010, pp. 471–474.

[14] M. Prats, D. Ribas, N. Palomeras, J. C. García, V. Nannen, S. Wirth, J. J. Fernández, J. P. Beltrán, R. Campos, P. Ridao *et al.*, "Reconfigurable auv for intervention missions: a case study on underwater object recovery," *Intelligent Service Robotics*, vol. 5, no. 1, pp. 19–31, 2012. doi: 10.1007/s11370-011-0101-z

[15] P. J. Sanz, P. Ridao, G. Oliver, G. Casalino, Y. Petillot, C. Silvestre, C. Melchiorri, and A. Turetta, "TRIDENT: An european project targeted to increase the autonomy levels for underwater intervention missions," in *Oceans-San Diego, 2013*. IEEE, 2013, pp. 1–10. doi: 10.23919/OCEANS.2013.6741370

[16] E. Simetti, G. Casalino, S. Torelli, A. Sperindé, and A. Turetta, "Floating underwater manipulation: Developed control methodology and experimental validation within the trident project," *Journal of Field Robotics*, vol. 31, no. 3, pp. 364–385, May 2014. doi: 10.1002/rob.21497

[17] A. Carrera, N. Palomeras, N. Hurtos, P. Kormushev, and M. Carreras, "Learning by demonstration applied to underwater intervention," in *CCIA*, 2014, pp. 95–104.

[18] P. Cieslak, P. Ridao, and M. Giergiel, "Autonomous underwater panel operation by GIRONA500 UVMS: A practical approach to autonomous underwater manipulation," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 529–536. doi: 10.1109/icra.2015.7139230

[19] B. Siciliano and J.-J. E. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Proc. Fifth Int Advanced Robotics 'Robots in Unstructured Environments', 91 ICAR. Conf.* Pisa, Italy: IEEE, 1991, pp. 1211–1216. doi: 10.1109/ICAR.1991.240390

[20] D. Youakim, P. Ridao, N. Palomeras, F. Spadafora, D. Ribas, and M. Muzzupappa, "Autonomous underwater free-floating manipulation using moveit!" *IEEE Robotics Automation Magazine*, vol. PP, no. 99, pp. 1–1, 2017. doi: 10.1109/MRA.2016.2636369

[21] G. Casalino, M. Caccia, S. Caselli, C. Melchiorri, G. Antonelli, A. Caiti, G. Indiveri, G. Cannata, E. Simetti, S. Torelli, A. Sperindè, F. Wanderlingh, G. Muscolo, M. Bibuli, G. Bruzzone, E. Zereik, A. Odetti, E. Spirandelli, A. Ranieri, J. Aleotti, D. Lodi Rizzini, F. Oleari, F. Kallasi, G. Palli, U. Scarcia, L. Moriello, and E. Cataldi, "Underwater intervention robotics: An outline of the italian national project MARIS," *Marine Technology Society Journal*, vol. 50, no. 4, pp. 98–107, jul 2016. doi: 10.4031/mtsj.50.4.7

[22] E. Simetti and G. Casalino, "A novel practical technique to integrate inequality control objectives and task transitions in priority based control," *Journal of Intelligent & Robotic Systems*, vol. 84, no. 1, pp. 877–902, apr 2016. doi: 10.1007/s10846-016-0368-6

[23] D. Ribas, P. Ridao, A. Turetta, C. Melchiorri, G. Palli, J. J. Fernandez, and P. J. Sanz, "I-AUV Mechatronics Integration for the TRIDENT FP7 Project," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2583–2592, Oct 2015. doi: 10.1109/TMECH.2015.2395413

[24] F. Oleari, F. Kallasi, D. Lodi Rizzini, J. Aleotti, and S. Caselli, "An underwater stereo vision system: from design to deployment and dataset acquisition," in *Proc. of the IEEE/MTS OCEANS*, 2015, pp. 1–5.

[25] F. Oleari, D. Lodi Rizzini, F. Kallasi, J. Aleotti, and S. Caselli, "Issues in High Performance Vision Systems Design for Underwater Interventions," in *Proc. of the 42nd Annual Conference of IEEE Industrial Electronics Society (IECON)*, 2016, pp. 945–950.

[26] J. Bemfica, C. Melchiorri, L. Moriello, G. Palli, U. Scarcia, and G. Vassura, "Mechatronic design of a three-fingered gripper for underwater applications," in *6th IFAC Symposium on Mechatronic Systems, Hangzhou, China*, 2013, pp. 307–312.

[27] J. Bemfica, C. Melchiorri, L. Moriello, G. Palli, and U. Scarcia, "A three-fingered cable-driven gripper for underwater applications," in *Proc. IEEE Int. Conf. on Robotics and Automation*, May 2014, pp. 2469–2474.

[28] G. Palli, L. Moriello, and C. Melchiorri, "Experimental evaluation of sealing materials in 6-axis force/torque sensors for underwater applications," in *Computer Aided Systems Theory*, R. Moreno-Díaz, F. Pichler, and A. Quesada-Arencibia, Eds. Springer International Publishing, 2015, pp. 841–852.

[29] G. Palli, L. Moriello, U. Scarcia, and C. Melchiorri, "Development of an optoelectronic 6-axis force/torque sensor for robotic applications," *Sensors and Actuators A: Physical*, vol. 220, no. 0, pp. 333–346, 2014.

[30] C. Melchiorri, L. Moriello, G. Palli, and U. Scarcia, "A new force/torque sensor for robotic applications based on optoelectronic components," in *Proc. IEEE Int. Conf. Robotics and Automation*, May 2014, pp. 6408–6413.

[31] C. Melchiorri and M. Kaneko, "Robot hands," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, pp. 345–360.

[32] K. L. Doty, C. Melchiorri, and C. Bonivento, "A theory of generalized inverses applied to robotics," *International Journal of Robotics Research*, vol. 12, no. 1, pp. 1–19, 1993. doi: 10.1177/027836499301200101

[33] N. Mansard, A. Remazeilles, and F. Chaumette, "Continuity of varying-feature-set control laws," *IEEE Trans. on Automatic Control*, vol. 54, no. 11, pp. 2493–2505, 2009. doi: 10.1109/TAC.2009.2031202

[34] G. Antonelli, G. Indiveri, and S. Chiaverini, "Prioritized closed-loop inverse kinematic algorithms for redundant robotic systems with velocity saturations," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 5892–5897. doi: 10.1109/IROS.2009.5354636

[35] L. L. Whitcomb and D. R. Yoerger, "Comparative experiments in the dynamics and model-based control of marine thrusters," in *OCEANS*, vol. 2. IEEE, 1995, pp. 1019–1028. doi: 10.1109/OCEANS.1995.528561

[36] L. L. Whitcomb and D. R. Yoerger, "Preliminary experiments in model-based thruster control for underwater vehicle positioning," *IEEE Journal of Oceanic Engineering*, vol. 24, no. 4, pp. 495–506, 1999. doi: 10.1109/48.809273

[37] R. Bachmayer, L. L. Whitcomb, and M. A. Grosenbaugh, "An accurate four-quadrant nonlinear dynamical model for marine thrusters: Theory and experimental validation," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 146–159, jan 2000. doi: 10.1109/48.820747

[38] S. Bazeille, I. Quidou, and L. Jaulin, "Color-based underwater object recognition using water light attenuation," *Intel Serv Robotics*, vol. 5, pp. 109–118, 2012.

[39] D. Lodi Rizzini, F. Kallasi, F. Oleari, and S. Caselli, "Investigation of vision-based underwater object detection with multiple datasets," *International Journal of Advanced Robotic Systems (IJARS)*, vol. 12, no. 77, pp. 1–13, may 2015.

[40] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin institute*, vol. 310, no. 1, pp. 1–26, 1980.

[41] D. Lodi Rizzini, F. Kallasi, J. Aleotti, F. Oleari, and S. Caselli, "Integration of a Stereo Vision System into an Autonomous Underwater Vehicle for Pipe Manipulation Tasks," *Computers and Electrical Engineering (CAEE)*, vol. 58, pp. 560–571, feb 2017. doi: 10.1016/j.compeleceng.2016.08.023

[42] A. Agrawal, S. Ramalingam, Y. Taguchi, and V. Chari, "A theory of multi-layer flat refractive geometry," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012. ISSN 1063-6919 pp. 3346–3353. doi: 10.1109/CVPR.2012.6248073

[43] T. Dolereit, U. von Lukas, and A. Kuijper, "Underwater stereo calibration utilizing virtual object points," in *OCEANS*, May 2015, pp. 1–7.

[44] T. Yoshikawa, "Analysis and control of robot manipulators with redundancy," in *Robotic Research: The First International Symposium*, M. Brady and R.Paul, Eds. MIT Press, 1984, pp. 735–747.

[45] G. Marani, J. Kim, J. Yuh, and W. K. Chung, "A real-time approach for singularity avoidance in resolved motion rate control of robotic manipulators," in *Proc. IEEE International Conference on Robotics and Automation ICRA '02*, vol. 2, 2002, pp. 1973–1978. doi: 10.1109/ROBOT.2002.1014830

[46] P. A. Di Lillo, E. Simetti, D. De Palma, E. Cataldi, G. Indiveri, G. Antonelli, and G. Casalino, "Advanced ROV autonomy for efficient remote control in the DexROV project," *Marine Technology Society Journal*, vol. 50, no. 4, pp. 67–80, jul 2016. doi: 10.4031/mtsj.50.4.8

[47] E. Simetti, F. Wanderlingh, G. Casalino, G. Indiveri, and G. Antonelli, "ROBUST project: Control framework for deep sea mining exploration," in *MTS/IEEE OCEANS 17*, Anchorage, US, 2017.

[48] G. Antonelli and E. Cataldi, "Virtual decomposition control for an underwater vehicle carrying a n-dof manipulator," in *MTS/IEEE OCEANS 2015*, Genoa, Italy, May 2015. doi: 10.1109/oceans-genova.2015.7271645

[49] D. De Palma, G. Indiveri, and G. Parlangeli, "Multi-vehicle relative localization based on single range measurements," in *IFAC Workshop on Multivehicle Systems (MVS 2015)*, vol. 48, no. 5. Elsevier, 2015, pp. 17–22.

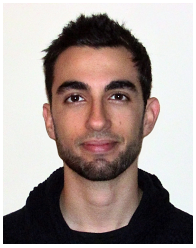
[50] G. Cossu, R. Corsini, A. Khalid, S. Balestrino, A. Coppelli, A. Caiti, and E. Ciaramella, "Experimental demonstration of high speed underwater visible light communications," in *Optical Wireless Communications (IWOW), 2013 2nd International Workshop on*. IEEE, 2013, pp. 11–15. doi: 10.1109/IWOW.2013.6777767

[51] E. Simetti and G. Casalino, "Whole body control of a dual arm underwater vehicle manipulator system," *Annual Reviews in Control*, vol. 40, pp. 191–200, 2015. doi: 10.1016/j.arcontrol.2015.09.011

[52] E. Simetti and G. Casalino, "Manipulation and transportation with cooperative underwater vehicle manipulator systems," *IEEE Journal of Oceanic Engineering*, pp. 1–18, 2016. doi: 10.1109/joe.2016.2618182

[53] G. Antonelli and E. Cataldi, "Basic interaction operations for an underwater vehicle-manipulator system," in *ICAR 2015 - 17th International Conference on Advanced Robotics*, Istanbul, Turkey, July 2015. doi: 10.1109/icar.2015.7251506

Enrico Simetti (M '10) received the Ph.D. degree from University of Genova in 2012, and since 2014 he is an Assistant Professor at DIBRIS, University of Genova. His research interests are in the area of autonomous systems and, in particular, on the development of control systems and coordination algorithms for robotic platforms, with special focus on marine robotics. In the last few years, he developed strong skills in real-time software development and in the development of embedded system software. He is the ISME responsible scientist in the H2020 ROBUST project.



Francesco Wanderlingh is a PhD candidate at University of Genova. His research interests are in the area of task priority based control schemes and their application to different robotics systems, ranging from underwater robots for intervention on subsea structures, to ground ones for factory of the future applications.

Sandro Torelli received the PhD from University of Genova in 2014, and since then he is a research fellow at DIBRIS, University of Genova. He is mainly interested in the development and control of robotic systems, with particular focus on underwater and ground robotics for intervention, and coordination of marine surface vehicles for surveillance and security applications in civilian harbour environments.

Marco Bibuli (M.Sc. 2005, Ph.D. 2010) is researcher at the CNR-ISSIA, focusing his research activity on the design and development of navigation, guidance and control algorithms for unmanned marine vehicles and software architectures for supervision and mission control. He is Principal Investigator for CNR in the EU FP7 project CADDY (20142016) and EU H2020 EXCELLABUST (2016-2018) and Responsible of Actions within the RITMARE Italian Flagship project; he has also been task leader in the EU FP7 projects MINOAS (20092012) and CART (20112013). He is the author of about 60 papers in international journals and conferences.

Angelo Odetti (M.Sc. 2010) is research fellow at the CNR-ISSIA, focusing his research activity on technologies and methodologies for the monitoring of wetlands and shallow water in coastal area by means of autonomous marine vehicles. This study will lead to a better definition of the characteristics of an air cushion platform and to the realization of a semi-autonomous surface vehicle. He will also work on the concept development of new marine autonomous vehicles. He is author of about 10 papers in international journals and conferences.

Gabriele Bruzzone (M.Sc. 1993) is the person in charge of the Robotics Lab and of the Computing Systems Lab of the Genoa Operating Unit of CNR-ISSIA. His main research activities focus on the design and the development of real-time and embedded hardware and software architectures for marine and maritime robotics, in particular unmanned marine vehicles, and for industrial automation. He participated to manifold Italian and European research projects, among the others: PRIN MARIS, Flagship initiative RITMARE, EU FP7 CART, EU FP7 MINOAS. He is author of more than 100 international publications with peer review.

Dario Lodi Rizzini received the PhD degree from the University of Parma in 2009, and is currently assistant professor at the Department at the University of Parma. He has been a visiting scholar at the University of Freiburg in 2007. His research interest are in the area of localization, mapping and navigation of mobile robots, and of 3D perception and object detection. During his career he focused on industrial and field robotic application.

Jacopo Aleotti is associate professor at the Department of Engineering and Architecture, University of Parma. He received his Laurea degree in Electronic Engineering and Ph.D. degree both from the University of Parma, in 2002 and 2006, respectively. In 2003, he was Marie Curie Fellow at the Learning System Laboratory, Orebro University, Sweden. His current research interests include robot learning from demonstration, robot manipulation, range sensing, human-robot interaction and virtual reality

Gianluca Palli received the Laurea and the Ph.D. degrees in automation engineering from the University of Bologna, Italy, in 2003 and 2007, respectively. Currently he is Assistant Professor at the University of Bologna. His research interests include design and control of robotic hands, modeling and control of robots with variable stiffness joints, design of compliant structures and actuation systems for robotics applications and development of real-time systems for automatic control applications.

Lorenzo Moriello received the Ph.D. in Automation and Operational research from the University of Bologna, Italy, in 2016. Currently he is research fellow at the University of Bologna. His research interests include design and control of robotic hands, development of optoelectronic sensors and design of highly integrated electronic system and the motion control of robotic manipulators.

Umberto Scarcia received his Master's Degree in Automation Engineering and the Ph.D. in Automatic Control and Operational Research from the University of Bologna, Italy, in 2011 and 2015, respectively. Currently he is Assistant Professor at the University of Bologna. His research interests include the design and control of innovative anthropomorphic robotic hands, dual arm/hand manipulation task planning and the design of innovative actuators and sensors for underwater applications.