Copyright Information

BibTeX

```
@ARTICLE{Simetti2017manip,
  author={Simetti, Enrico and Casalino, Giuseppe},
  journal={IEEE Journal of Oceanic Engineering},
  title={Manipulation and Transportation With Cooperative Underwater
     Vehicle Manipulator Systems},
  year={2017},
  volume={42},
  number={4},
  pages={782-799},
  doi={10.1109/JOE.2016.2618182},
  ISSN={0364-9059},
  month={Oct}
}
```

Manipulation and Transportation with Cooperative Underwater Vehicle Manipulator Systems

Enrico Simetti, *Member, IEEE,* and Giuseppe Casalino, *Member, IEEE*

Abstract

Autonomous underwater manipulation is a topic of interest since the early nineties. In the past few years, several milestone projects such as SAUVIM and TRIDENT have demonstrated autonomy capabilities for a single underwater vehicle manipulator system (UVMS) in performing simple manipulation tasks, e.g. the recovery of an object from the seafloor. The Italian funded MARIS project aims to extend some of those results to multiple UVMSs performing a cooperative transportation task of a long object such as a pipe. This paper presents the results achieved in developing a unifying architecture for the control of both individually and cooperatively operating UVMSs that explicitly makes use of a limited amount of information exchange between the agents, which is needed due to the severe bandwidth limitations of the underwater acoustic communications. A complete execution of the reference transportation mission is presented to support the proposed distributed algorithm. Furthermore, hydrodynamic simulations of the cooperative transportation phase are presented and an analysis of the achievable performances as different communication schemes are employed is given.

Index Terms

Underwater manipulation, task priority control, underwater vehicle manipulator system, distributed dynamic control, cooperative control

E. Simetti and G. Casalino are with the Interuniversity Research Center on Integrated Systems for the Marine Environment, and with the Department of Computer Science, Bioengineering, Robotics, and System Engineering, University of Genova, Via Opera Pia 13, 16145 Genova, Italy e-mail: (see http://www.graal.dibris.unige.it and http://www.isme.unige.it).

Manuscript received -; revised -.

I. Introduction

Underwater manipulation tasks are nowadays performed either using manned submersibles or using work class ROVs (Remotely Operated Vehicles). The first option, from the manipulation point of view, has the clear advantage of having the operator in direct view of the object to be manipulated, but with the increased risk of having the human being in a very hostile environment and a reduced operational time. The second option is to employ ROVs, which are underwater vehicle manipulator systems (UVMS) remotely piloted from a surface vessel, with commands sent through an umbilical cable. Such ROVs usually require a vessel with a heavy crane to deploy them, an umbilical management system to handle their umbilical cable properly and dynamic positioning to keep the vessel stationary. In this case, the human operator is not anymore in the hostile environment, but the pilots operating the vehicle and the arms still go through heavy fatigue for coordinating themselves to carry out the manipulation task and managing the umbilical cable correctly. In both cases, the operational costs are very high due to the requirement of expensive support vessels, as well as highly qualified man power effort.

For the above reasons, the research has directed its efforts toward increasing the autonomy level in underwater manipulation, either by implementing some autonomous control features in work class ROVs, to reduce the operator's fatigue, or even by completely replacing ROVs with AUVs (Autonomous Underwater Vehicles). During the 90s, the Woods Hole Oceanographic Institute has carried out different works concerning the design and control of compliant underwater manipulators [1] and coordinated vehicle/arm control for tele-operation [2]. A pioneering project in this field is represented by the AMADEUS project [3] that between 1993 and 2000 developed grippers for underwater manipulation [4] and studied the problem of dual arm autonomous manipulation [5], demonstrating these features in water tank experiments. The UNION project [6] has instead represented the first project where the mechatronic assembly of an autonomous UVMS has taken place. An overall survey on the developed control architectures for underwater robots until late 90s can be found in [7].

In the early 2000s, two major milestones in autonomous underwater manipulation were achieved. The ALIVE project [8], [9] has demonstrated the capability of an underwater vehicle to dock autonomously with a ROV-friendly panel by using hydraulic grabs, and then performing some manipulation tasks such as opening and closing a valve. Within such a context, the manipulation task was basically a fixed-base one, and no particular interaction between the vehicle and the manipulator controllers was needed. The SAUVIM project [10], [11] has instead carried the first ever autonomous floating underwater intervention. Since the AUV was weighting 6 tons, and the arm only 65 kg, the dynamics of the two subsystem were practically decoupled, and indeed the two controllers were separate. The AUV was performing station keeping while the arm was instead performing the recovery of an object on the sea floor.

About ten years later, the Spanish funded RAUVI project [12] has taken a step further by developing the Girona 500 light AUV, which equipped with a small 4 d.o.f. (degrees of freedom) arm performed a hook-based recovery in a water tank. As in previous projects, the control of the vehicle and the arm was separated, i.e. the vehicle was performing station keeping and was not supporting the arm in a coordinated fashion, despite the masses of the AUV and the arm were now much more similar than in SAUVIM.

A true milestone has been achieved with the TRIDENT project [13], where for the first time a vehicle and an arm of comparable masses were controlled in a coordinated manner [14], [15] to perform autonomous underwater recovery of a blackbox mockup in a harbour environment [16]. In such works, the authors have demonstrated how the UVMS could be jointly controlled to effectively exploit all the available d.o.f. to perform autonomously the manipulation task while also taking care of a set of other control objectives such as, for example, keeping the object centered in the camera frame and respecting the arm joint limits.

Currently, the Spanish funded TRITON project [17] is dealing with the intervention on an underwater panel in a permanent observatory and free floating manipulation for camera dome defouling, the EU project PANDORA [18] is instead focusing on the issue of persistent autonomy, i.e. reducing the frequency of assistance requests, by recognising failures and responding to it, at all levels of abstraction, and the DexROV project [19] is dealing with the inspection and maintenance in presence of communication latencies [20], [21]. For a further comprehensive read on individually operating UVMSs, see the books [22]–[24].

Nowadays, the possibility of using autonomous dual arm UVMSs [25]–[28] and even cooperating UVMSs is slowly arising. The Italian funded MARIS project [29] is dealing with the development of control algorithms capable of integrating single, dual arm and cooperative UVMSs within a common control framework [30], [31]. Indeed, the contribution of this paper is the extension of the control framework developed within the TRIDENT project to the case of cooperating UVMSs, building on the authors' previous experience on cooperative mobile robotics [32], [33].

The particular underwater environment poses great constraints on the available communication bandwidth between the robots, thus major efforts have been spent in trying to devise a cooperative control algorithm with limited information exchange. The proposed algorithm requires the exchange of only six numbers between the agents, which considering floating point variables requires 192 bits. The achievable exchange rate depends on the available bandwidth. Field experiments reported in [34] achieved a data rate of 5 kbit/s at a distance of 6 km, while [35] achieved a bandwidth of up to 10 kbit/s. In such a case the exchange of velocities could be done at 25 and 50 Hz respectively.

In many cases such bandwidths can be achieved only after establishing a connection through a handshaking sequence, which introduces a considerable overhead as well as processing time to complete the connection. This is usually fine whenever a large amount of data must be sent. However, it is not suited for sending small amount of data repeatedly in time, as it is in our case.

For example, acoustic modems available within ISME (see for example the S2CR-18-34 Modem, capable of 13.9 kbit/s at short ranges [36] in the most favorable conditions) have only a bandwidth of 976 bit/s whenever small packets need to be sent, which would allow a maximum theoretical exchange rate of 5 Hz.

In addition to the above problems of bandwidth and latency, there is the problem of handling the medium access. Most of the commercial modems only allow half-duplex communications with time division policies that the user must implement. This further increases the difficulty of coordinating the UVMSs, as they now need to implement a time division medium access policy, further reducing the achievable exchange rate. A possible solution to have a full-duplex communication is to employ two acoustic modems characterized by different working frequencies, with the downside of increased cost.

An alternative to acoustic communications is represented by visible light communications (VLC). Experiments on this technology reported in [37], [38] show that the achievable full-duplex data rate is in the order of Mbit/s even at 100 m range. In this case the exchange rate could be carried out at the same frequency of the control loop (typically around 100 Hz for an UVMS). VLC is affected by problems such as turbidity of the water. It is however a very promising alternative, especially in this case as the UVMSs will operate quite close to each other and thus the attenuation loss will not be a key factor.

With respect to our previously published work on this subject [31], [39], [40], in this work we provide a comprehensive view of the kinematic task priority framework, including how to implement a seamless transition between different mission phases. The heavy mathematical details about the pseudo-inversion scheme are instead reported in another paper [41]. In addition, with respect to [16], [42] we are now using a two-step optimization procedure (a centralized law plus an arm control law parametrized by the vehicle velocity) based on the new pseudo-inversion scheme [41] as opposed to a dynamic programming based procedure, which required high damping values to avoid chattering phenomena. Furthermore, in this work a deeper investigation of the cooperative transportation phase is presented. In particular, the simulations include the hydrodynamic models of both UVMSs (as opposed to the preliminary kinematic simulations performed in the previous works) and different communication schemes are tested and compared (half vs full duplex channels and with or without latencies, whereas in previous works we assumed a full-duplex channel with no latency).

In this work we will focus on the kinematic control layer, and we will assume a dynamic control layer based on PI (proportional integrative) loops. This is due to the fact that most ROV and arm commercial system only provide a velocity interface to control the robot. This is the case of the DexROV and MARIS projects.

The work is structured as follows. We first report some basic definitions, such as the reference mission, its phases and control objectives in Section II that apply to either an UVMS acting alone or to each of the agents performing a cooperative transportation task. Then, the control of an individually operating UVMS is briefly recalled in Section III. The proposed cooperative algorithm is presented in Section IV, and the simulation results of a complete mission are reported in Section V, together with a deeper investigation of the cooperative phase through hydrodynamic simulations with communication constraints such as limited bandwidth and latency. Finally some conclusions are given in Section VI.

II. Definitions and Preliminaries

A. Notation

Vectors and matrices are expressed with a bold face character, such as $\boldsymbol{M}$, whereas scalar values are represented with a normal font such as γ. Given a matrix $\boldsymbol{M}$ and a vector $\boldsymbol{v}$:

- $M_{(i,j)}$ indicates the element of $\boldsymbol{M}$ at the i-th row and j-th column;
- $v_{(k)}$ refers to the k-th element of $\boldsymbol{v}$;

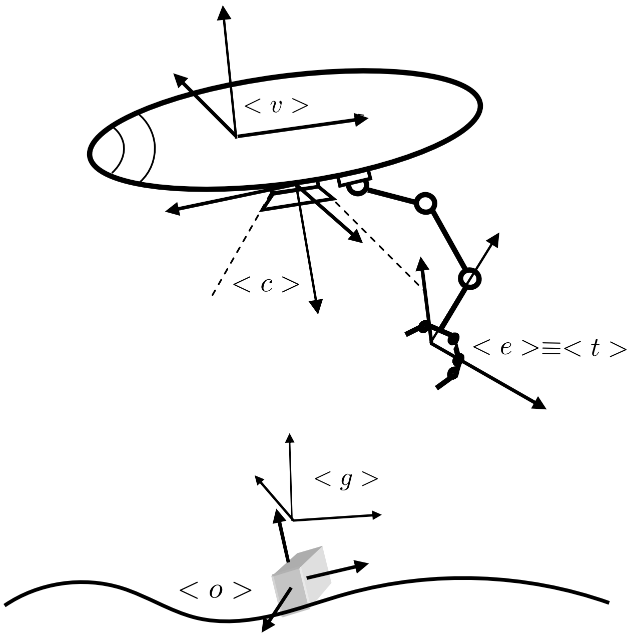


Fig. 1. The UVMS and its relevant frames

- $\boldsymbol{M}^{\#}$ is the exact generalized pseudo-inverse (see [43] for a review on pseudo-inverses and their properties), i.e. the pseudo inverse of $\boldsymbol{M}$ performed without any regularizations.

Further, less used, notation will be introduced as needed.

B. Definitions

Let us consider a single free floating UVMS, possibly an agent of a team, such as the one depicted in Fig. 1, and let us first introduce some basic definitions, often used throughout the paper:

- 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 *generalised coordinate position vector*, which is the stacked vector of the position vector $\boldsymbol{\eta}_1$, with components on the inertial frame $\langle 0 \rangle$, and the orientation vector $\boldsymbol{\eta}_2$, the latter expressed in terms of the three angles yaw, pitch and roll (applied in this sequence) [44]. 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 velocity vector $\boldsymbol{v}_1$ and the vehicle angular velocity vector $\boldsymbol{v}_2$, both with components on the vehicle frame $\langle v \rangle$. For simplicity of discussion, we are assuming the vehicle fully actuated, hence in the following we will use the system velocity vector as our control vector. Details on how the proposed algorithm can be adapted to work with underactuated vehicles are given in the Appendix;
- a configuration dependant scalar variable $x(\boldsymbol{c})$ is said to correspond to an *equality control objective* when it is eventually required to satisfy
$$x(\boldsymbol{c}) = x_0, \tag{3}$$
or to an *inequality control objective* when it is required to satisfy
$$x(\boldsymbol{c}) \geq x_m \tag{4}$$
or
$$x(\boldsymbol{c}) \leq x_M, \tag{5}$$
where the subscripts m and M indicate a minimum and maximum value respectively. The case where a variable needs to stay within an interval can be represented by two separate objectives. Examples of such variables are the arm joints q_i, which are required to be within the joint limits, or the manipulability measure μ, which is required to be above a certain minimum threshold.
However, we can also consider as $x(\boldsymbol{c})$ the modulus of a certain vector $\boldsymbol{p}$. This allows to control its norm to a particular value (for instance if we want to nullify some error vector), to be below a given threshold (if we just need to keep a reasonable error bound), or to be above a threshold (for instance if such a vector is a distance between two objects to avoid collisions).
Furthermore, we can also consider $x(\boldsymbol{c})$ to be the $i-$th component of a vector $\boldsymbol{p} \in \mathbb{R}^m$. Then if we consider m different variables $x_i(\boldsymbol{c})$, it is possible to control the vector $\boldsymbol{p}$ to any desired value.

For the remainder of the paper, we will drop the dependency of x from $\boldsymbol{c}$ to ease the notation;

- for such variables, we also consider the existing Jacobian relationship between x and the system velocity vector $\dot{\boldsymbol{y}}$ as

$$\dot{x} = \boldsymbol{g}^T(\boldsymbol{c})\dot{\boldsymbol{y}}, \tag{6}$$

  where $\boldsymbol{g} \in \mathbb{R}^n$ is a vector. Again, in the rest of the paper we will drop the dependency of $\boldsymbol{g}$ from $\boldsymbol{c}$. All the details concerning how to compute the Jacobians corresponding to these variables can be found in the TRIDENT technical report [45];
- we define as *task* the need of tracking at best a suitable reference rate $\dot{\bar{x}}$ (see the remarks below), capable of driving the associated variable x toward the corresponding objective. Thus, for instance, a task is tracking at best a velocity reference rate generated to bring the arm's end-effector in the required Cartesian position;
- the control objectives may have different *priorities* and the same holds for their associated tasks. The achievement of a task with lower priority should not interfere with the achievement of an *active* task (see Section III-C) with higher priority, and tasks with the same priority should be achieved simultaneously, if possible. A set of tasks with different priorities is also called a *hierarchy of tasks*.

Remark 1: Let us consider a vector $\boldsymbol{p} \in \mathbb{R}^m$, the variable $x_1 = \|\boldsymbol{p}\|$ and the m variables $x_{2,i} = p_{(i)}$, $i = 1, \ldots, m$. Trivially, since the following equivalence between these objectives holds

$$x_1 = 0 \iff x_{2,i} = 0, \forall i, \, i = 1, \ldots, m, \tag{7}$$

if we are interested in zeroing the norm of a given vector $\boldsymbol{p}$ we can either consider a single equality control objective that requires its norm to be zero, or m objectives one for each of its components. The difference between the two possibilities is that on the one hand we have to employ only one d.o.f. in the Cartesian space and we impose a required behaviour to the derivative of the norm, on the other hand we employ m d.o.f. in the Cartesian space and we are separately controlling each of its components.

Remark 2: For equality control objectives, a suitable reference rate $\dot{\bar{x}}$ that drives x toward x_0 is

$$\dot{\bar{x}} \triangleq \gamma(x_0 - x), \, \gamma > 0, \tag{8}$$

where γ is a positive gain to control the convergence speed.

Remark 3: For inequality control objectives, a suitable reference rate $\dot{\bar{x}}$ 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 $x \leq x_M$ and consider a point x^* such that $x^* \leq x_M$, then a suitable reference rate that drives x toward its corresponding objectives is

$$\dot{\bar{x}} \triangleq \gamma(x^* - x),\, \gamma > 0. \tag{9}$$

We shall see in Section III-C how to disregard the reference rate whenever $x \leq x_M$ in order to avoid over-constraining the system.

C. Reference Mission and Relevant Phases

The typical manipulation and transportation reference mission carried out either by a single agent or by a team of UVMSs can be decomposed into the following sequential phases:

1) navigation: the vehicle(s) should (concurrently) get in close proximity with the target object to be manipulated;
2) grasping: the UVMS(s) must (concurrently) perform the grasping of the (shared) object;
3) transportation: the UVMS(s) must transport the (shared) object to the target area;
4) deployment: whenever in close proximity with the target area, the (shared) object must be deployed in the required position.

At the end of the above phases, the UVMS(s) can leave the target area and each of them can be assigned to a new mission.

D. Individual and Shared Control Objectives

Each agent generally has different individual control objectives to be possibly achieved during each phase of a mission; some of them can be traced back to the arm configuration $\boldsymbol{q}$ only, some to vehicle configuration $\boldsymbol{\eta}$ only, while others to the whole system configuration vector $\boldsymbol{c}$ or parts of it.

The objectives that are related only to the arm are the following ones:

1) Joint Limits Objective: the manipulator must operate within its joint limits, which means having the following inequality control objectives fulfilled

$$\begin{cases} q_i \geq q_{i,m} \\ q_i \leq q_{i,M} \end{cases} \quad i = 1, \ldots, l, \tag{10}$$

where q_i is the i-th joint variable, $q_{i,m}$ is the lower bound and $q_{i,M}$ is the higher one for the joint i, and where l is the total number of joints of the manipulator.

2) Dexterity Objective: the arm must maintain a good dexterity, to be able to perform the manipulation tasks without incurring into the problems related to the kinematic singularities, which means to keep the manipulability measure μ [46] above a minimum threshold, thus fulfilling the following inequality control objective

$$\mu > \mu_m. \tag{11}$$

Another possibility to maintain a good dexterity is to define a preferred shape of the manipulator, where it is known that the arm is characterized by a good manipulability, and then try to maintain a suitable distance between the current arm configuration and the preferred one.

3) Camera Occlusion Objective: a further inequality objective that might be required for the manipulator is to keep its links away from the camera system's cone of vision (assumed fixed to the vehicle), to avoid unnecessary occlusions of the target object frame $\langle o \rangle$. This task depends on the specific configuration of the arm: unavoidable occlusions might occur when the end-effector is in close proximity to the target, but in other cases other types of avoidable occlusions may happen, for instance when the elbow of the arm interferes with the camera well before the final grasping phases. In this case, a further inequality control objective can be introduced to force the elbow to stay outside the vision system cone.

The objectives that are related only to the vehicles are usually the following ones:

4) Camera Centering Objective: since in the underwater environment the availability of an absolute positioning system is not always guaranteed, the navigation toward a particular target and especially the grasping of an object is often carried out by means of the stereo vision of the target itself. Thus, it becomes important that the visual contact between the system and the target is maintained as the vehicle approaches it. This can be translated by requiring that the target is grossly centered within the vision system. Toward that end, the control must ensure that the norm of the misalignment vector $\boldsymbol{\xi}$ between the vector joining the object $\langle o \rangle$ and the camera frame $\langle c \rangle$ with the z axis of the camera frame itself (supposed going outwards the camera image plane, as in Fig. 1) is within a maximum threshold, i.e.

$$\|\boldsymbol{\xi}\| \leq \xi_M. \tag{12}$$

5) Horizontal Attitude Objective: the vehicle attitude should be maintained within reasonable bounds, to avoid situations where the vehicle is upside-down. However, especially for vehicles and arms with comparable masses, such thresholds should be big enough to avoid excessive energy consumption to keep the vehicle perfectly horizontal whenever the arm moves its joints away from the center of buoyancy and thus tilts the vehicle. The above outlined control objective requires the achievement of the following inequality

$$\|\boldsymbol{\varphi}\| \leq \varphi_M, \tag{13}$$

where $\boldsymbol{\varphi}$ represents the misalignment vector that the absolute world frame z axis forms with respect to the vehicle z axis one. Note that most of the ROVs or AUVs are passively stable in roll and pitch and these d.o.f. are not controllable. For this reason, this objective should be considered only for fully actuated vehicles.

6) Vehicle Position Control: another control objective is to have the vehicle frame $\langle v \rangle$ grossly aligned with a particular goal frame $\langle g_v \rangle$. This could be required in order to bring the vehicle close to the area where the manipulation activity needs to be carried out. This goal requires the achievement of the following inequality conditions

$$\|\boldsymbol{r}_v\| \leq r_{v,M} \;\; ; \;\; \|\boldsymbol{\vartheta}_v\| \leq \vartheta_{v,M}, \tag{14}$$

where $\boldsymbol{r}_v$ is the position error and $\boldsymbol{\vartheta}_v$ the orientation error.

Moreover, the individual objectives that are instead related with both the vehicle and arm are generally represented by the following ones:

7) End-effector/Tool-frame Position Control: this objective requires that the tool frame $\langle t \rangle$, rigidly attached to the end-effector space, converges to a given goal frame $\langle g \rangle$. In other words, the following two equality objectives must be eventually satisfied

$$r_{(i)} = 0, \, i = 1, 2, 3\,; \quad \vartheta_{(i)} = 0, \, i = 1, 2, 3\,; \tag{15}$$

where $\boldsymbol{r}$ is the position error and $\boldsymbol{\vartheta}$ the orientation error between the tool and goal frames. Note that we have chosen a component by component zeroing for achieving a straight convergence to the target (especially important for grasping tasks).

As it concerns objectives shared between the systems, we consider the following one:

8) Vehicles Distance: the vehicles must maintain a certain distance between themselves to avoid collision. This objective can be simply stated as maintaining the norm of the vector $\boldsymbol{d}$ joining two vehicle frames $\langle v_a \rangle$ and $\langle v_b \rangle$ above a certain threshold value, i.e.

$$\|\boldsymbol{d}\| \geq d_m. \tag{16}$$

Note that such a distance can be typically separately evaluated by the agents by means of their vision system or through acoustic distance evaluations directly embedded in their acoustic communication mechanism.

As already stated, all the details concerning how to compute the Jacobians of the tasks corresponding to these objectives can be found in the TRIDENT technical report [45]. As a final remark, note that the actual priority of the tasks corresponding to these objectives will be outlined in Section III-E.

III. Control of Individually Operating UVMS

The control of an individually operating UVMS is achieved by solving a sequence of optimization problems, following the assigned priority of each control objective. This mechanism descends from the original task priority framework [47] that within the TRIDENT project [16] has been extended to also encompass tasks corresponding to inequality control objectives, where each of them assigned with different priorities, while in the MARIS project it has been extended to include clusters of control objectives with equal priorities [27], [41], [48]. We briefly recall the basic steps behind the algorithmic structure of the task priority based kinematic control layer of the UVMS.

A. Phase Control Priorities

Each mission phase is characterized by a set of relevant control objectives. In particular, these objectives, or equivalently their associated tasks, are listed according to a suitably chosen priority list. This is exemplified in the following list, where the symbol of the reference rate associated to each task is used just for denoting its corresponding task:

$$\begin{aligned}&\dot{\bar{x}}_1,\\&\dot{\bar{x}}_2,\\&\begin{bmatrix}\dot{\bar{x}}_{3,1} & \dot{\bar{x}}_{3,2} & \dot{\bar{x}}_{3,3}\end{bmatrix}^T \triangleq \dot{\bar{\boldsymbol{x}}}_3,\\&\begin{bmatrix}\dot{\bar{x}}_{4,1} & \dot{\bar{x}}_{4,2}\end{bmatrix}^T \triangleq \dot{\bar{\boldsymbol{x}}}_4,\\&\vdots\\&\dot{\bar{x}}_N,\end{aligned}$$

where in the above the first index indicates the decreasing priority level, while the second one denotes the tasks allocated at the same priority level. In particular, we can note how the possible presence of tasks with the same priority naturally translates into the presence of what we call as multidimensional tasks.

In the following, when we shall refer to a list of tasks, for the sake of generality we shall therefore consider scalar tasks as a particular case of the multidimensional ones, and consequently we shall indicate a prioritized task list more simply as $\dot{\bar{\boldsymbol{x}}}_1, \cdots, \dot{\bar{\boldsymbol{x}}}_N$. Furthermore, we shall indicate the Jacobians relevant to the actual task velocities $\dot{\boldsymbol{x}}_1, \cdots, \dot{\boldsymbol{x}}_N$ as $\boldsymbol{J}_1, \cdots, \boldsymbol{J}_N$.

B. Unified Task List

As evidenced in the previous point, each phase is characterized by its task hierarchy. Some of these lists may have different tasks in common, even if with a different ordering within each list. For instance, consider the following two hypothetical lists of scalar tasks (now abstractly labelled with alphabetic letters) for two different phases, where $A \prec B$ denotes that A has higher priority than B:

$$\begin{aligned}\text{P1 :}&A \prec B, C, D\\\text{P2 :}&A \prec D \prec C, E\end{aligned}$$

where A, D, C are in common, but with D at a different priority ordering w.r.t. C within the two lists. It is always possible to find a minimal length larger list where each original task list can be extracted via trivial binary logic operations. For instance, for the above hypothetical two task lists, an associated merged sequence is the following one

$$\text{P1}, \text{P2} : A \prec D \prec B, C, D, E;$$

where some tasks may be duplicated as the example shows. The suitable insertion/deletion of some of its entries, while transitioning among the phases, always produces the task list of the current entering phase. Such a trivial logic mechanism for extracting the phase task sequences from the unified one is implemented through the use of the continuous activation functions that are presented in the next subsection.

C. Activation Functions

Let us consider a multidimensional task, and let us consider an activation function associated to each j-th of its components, called $a_{(j)}$, to be then organized in a diagonal activation matrix $\boldsymbol{A}$, whose meaning is the following:

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

In particular, we construct the overall activation function $a_{(j)}$ as the product of two functions

$$a_{(j)} \triangleq a^{p}_{(j)} a^{i}_{(j)}, \tag{17}$$

which have the following specific purposes:

- $a^{i}_{(j)}$ (where the superscript stands for inequality) is function of the current value of the actual j-th component $x_{(j)}$, which represents the state of the associated control objective, and is used to activate/deactivate a scalar task associated to an inequality type control objective;
- $a^{p}_{(j)}$ (where the superscript stands for phase) is a function of a suitably chosen vector variable $\boldsymbol{p}$, used for measuring the status of the current phase, and its output is exploited to activate/deactivate any task involved/not-involved in the new phase whenever there is a phase transition. An example is the activation of the end-effector position control only when the vehicle is sufficiently close to the object to be grasped. This allows a continuous transition from the first mission phase where the vehicle navigates to the object and the successive grasping phase.

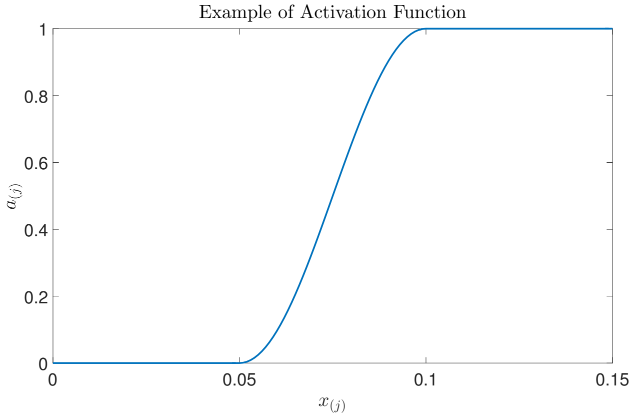


Fig. 2. Example of activation function corresponding to a control objective $x_{(j)} \leq 0.1$, with $\beta = 0.05$

For each inequality control objective, we consider as activation function $a^i_{(j)}$ the one defined as follows for objectives of the type $x_{(j)} \leq x_{(j),M}$ (a similar function can be constructed for objectives $x_{(j)} \geq x_{(j),m}$):

$$a^i_{(j)} \triangleq \begin{cases} 1, & x_{(j)} > x_{(j),M} \\ s_j(x), & x_{(j),M} - \beta_{(j)} \leq x_{(j)} \leq x_{(j),M} \\ 0, & x_{(j)} < x_{(j),M} - \beta_{(j)} \end{cases} \tag{18}$$

where $s_j(x)$ is any sigmoid function with a continuous behaviour from 0 to 1 when $x_{(j),M} - \beta_{(j)} \leq x_{(j)} \leq x_{(j),M}$. The $\beta_{(j)}$ value allows to create a buffer zone, where the inequality is already satisfied, but the activation value is still greater than zero. This is necessary to prevent any chattering problem around the inequality control objective threshold. An example of such a function is reported in Fig. 2. Note that for equality control objectives it clearly holds that $a^i_{(j)} = 1$.

The activation value $a^p_{(j)}$ is instead a value which depends on the status of the specific mission phase and possibly of the time elapsed within the phase itself, which allows to perform phase transitions. For example, as soon as the navigation phase is complete and the grasping one has to start, the $a^p_{(j)}$ of the vehicle position control goes to zero after some T seconds have elapsed, in order to deactivate the task; contemporarily, the $a^p_{(j)}$ of the end-effector task rises to one to activate the control of the end-effector to execute the grasp of the object.

Remark: We have implicitly considered objectives of the type $x_m < x < x_M$ as two separate ones. Note that if x_m and x_M are sufficiently spaced, i.e. $x_m + \beta < x_M - \beta$, then they can be considered together by using as activation function the sum of the two activation functions, and by choosing an arbitrary point inside the validity of both inequality to construct the common reference rate in (9). This is actually what is done for the joint limits task implementation, since the minimum and maximum limits satisfy the above conditions.

D. Additional Tasks

In addition to the tasks corresponding to the control objectives outlined in Section II-D, there can be tasks directly specified at a velocity level, which are useful for better controlling the behaviour of the overall system. Let us consider the following ones:

- Arm Motion Minimality: during the transportation phase it might be better if the arm minimizes its movements, leaving the bulk of the work to move the end-effectors to the vehicles. This allows to use the arm only to compensate the vehicle controller errors in tracking its reference velocity, as will be explained in Section III-G;
- Vehicle Motion Minimality: during the grasping and manipulation phases it might be preferred to have the vehicle roughly stationary, unless its movement is strictly needed. This is a consequence of the fact that, usually, the control performances of the vehicle are fairly worse than those of the arms.

Since these tasks require the minimization of their respective subsystem control variables, they should be placed at the bottom of the task hierarchy, as they consume any residual arbitrariness on their relevant control variables. Furthermore, they should be activated alternatively, in order to prioritize only one of the two subsystems.

E. Priority List

The control objectives of the UVMS can be divided in five broad categories:

- objectives related to physical constraints, i.e. tasks that deal with the interaction with the environment;
- objectives related to the safety of the system, e.g. avoiding joint limits or obstacles;
- objectives that are a prerequisite for the execution of the mission, e.g. maintaining the object to be manipulated in the camera vision system;
- mission oriented objectives, i.e. what the system really needs to execute to accomplish the user defined mission;

- optimization objectives, i.e. objectives that do not influence the mission, but allow to choose between multiple solutions, in case multiple solution exist.

These categories have been listed in their natural descending order of priority.

Given the above division, the considered unified prioritized list of tasks for individually operating UVMS is the following one:

1) Joint limits avoidance;
2) Minimum vehicle distance;
3) Dexterity (manipulability);
4) Camera object centering;
5) Horizontal attitude;
6) Vehicle position control;
7) Arm end-effector position control;
8) Minimization of arm movements;
9) Minimization of vehicle movements.

We must remark how the vehicle and end-effector position control, which are those most relevant for the execution of different mission phases are necessarily at a low priority. This is due to the fact that the higher priority tasks are all related to the safety of the system (joint limits, vehicle distances, horizontal attitude) or are needed to enable to system to actually execute the mission (manipulability, camera object centering).

F. Solution of the Task Hierarchy Problem

Given the definitions of the above sections, the problem of tracking with priorities the given reference velocities of each task can be found as the solution of a sequence of minimization problems:

$$S_k \triangleq \left\{ \arg \mathrm{R}\text{-} \min_{\dot{\boldsymbol{y}} \in S_{k-1}} \left\| \boldsymbol{A}_k (\dot{\bar{\boldsymbol{x}}}_k - \boldsymbol{J}_k \dot{\boldsymbol{y}}) \right\|^2 \right\}, \quad k = 1, 2, \ldots, N, \tag{19}$$

where S_{k-1} is the manifold of solutions of all the previous tasks in the hierarchy and the notation $\mathrm{R}\text{-}\min$ underlines the fact that the minimization process is performed in a special regularized manner, to avoid the discontinuity problems that actually arise in presence of activation function transitions [49], [50]. This methodology (named *iCAT task priority framework*) is duly reported in [41], [48] and will be omitted here. We simply recall that the solution of the $\mathrm{R}\text{-}\min$ problem exploits the definition of the special regularized pseudo inverse operator $\boldsymbol{X}^{\#,\boldsymbol{A},\boldsymbol{Q}}$ which is built as hereafter indicated, for given non-negative definite matrices $\boldsymbol{A}, \boldsymbol{Q}$, with dimensions equal to the rows and columns of $\boldsymbol{X}$ respectively:

$$\boldsymbol{X}^{\#,\boldsymbol{A},\boldsymbol{Q}} \triangleq \left(\boldsymbol{X}^T\boldsymbol{A}\boldsymbol{X} + (\boldsymbol{I}-\boldsymbol{Q})^T(\boldsymbol{I}-\boldsymbol{Q}) + \boldsymbol{V}^T\boldsymbol{H}\boldsymbol{V}\right)^{\#}\boldsymbol{X}^T\boldsymbol{A}\boldsymbol{A}, \tag{20}$$

where $\boldsymbol{V}$ is the right orthonormal matrix of the SVD decomposition of $\boldsymbol{X}^T\boldsymbol{A}\boldsymbol{X}+(\boldsymbol{I}-\boldsymbol{Q})^T(\boldsymbol{I}-\boldsymbol{Q})$ and $\boldsymbol{H}$ is a diagonal (singular value oriented regularization) matrix, whose elements $h_{(i,i)}$ are bell-shaped, finite support functions of the corresponding singular value of the same mentioned SVD decomposition. As a brief insight, note how the pseudo inverse operator depends explicitly on the activation matrix $\boldsymbol{A}$ defined in Section III-C, which allows to take properly into account the activation/deactivation of some rows of the multidimensional task and to eliminate discontinuity problems that might occur at the same priority level. In addition, the operator also depends on the projection matrix $\boldsymbol{Q}$, which allows to eliminate discontinuity problems that might arise between different priority levels.

We further hereafter report from [41], [48] the algorithmic translation of the above minimization sequence (19), which results as follows

$$\boldsymbol{\rho}_0 = \boldsymbol{0}, \quad \boldsymbol{Q}_0 = \boldsymbol{I}, \tag{21}$$

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

$$\begin{aligned} \boldsymbol{W}_k &= \boldsymbol{J}_k\boldsymbol{Q}_{k-1}(\boldsymbol{J}_k\boldsymbol{Q}_{k-1})^{\#,\boldsymbol{A}_k,\boldsymbol{Q}_{k-1}}, \\ \boldsymbol{Q}_k &= \boldsymbol{Q}_{k-1}(\boldsymbol{I} - (\boldsymbol{J}_k\boldsymbol{Q}_{k-1})^{\#,\boldsymbol{A}_k,\boldsymbol{I}}\boldsymbol{J}_k\boldsymbol{Q}_{k-1}), \\ \boldsymbol{\rho}_k &= \boldsymbol{\rho}_{k-1} + \boldsymbol{Q}_{k-1}(\boldsymbol{J}_k\boldsymbol{Q}_{k-1})^{\#,\boldsymbol{A}_k,\boldsymbol{I}}\boldsymbol{W}_k\left(\dot{\bar{\boldsymbol{x}}}_k - \boldsymbol{J}_k\boldsymbol{\rho}_{k-1}\right), \end{aligned} \tag{22}$$

where

- $\boldsymbol{\rho}_k$ is the control vector, which is computed in an iterative manner by descending the various priority levels;
- $\left(\dot{\bar{\boldsymbol{x}}}_k - \boldsymbol{J}_k\boldsymbol{\rho}_{k-1}\right)$ is the modified task reference that takes into account the contribution of the control vector $\boldsymbol{\rho}_{k-1}$ established at the previous iteration;
- $\boldsymbol{Q}_{k-1}$ is the projection matrix that is used to take into account the control direction (totally or partially) used by the higher priority tasks;
- $\boldsymbol{W}_k$ is a $m \times m$ matrix, where m is the row-dimension of the task at the current priority level, whose effect is to modify the task reference $\left(\dot{\bar{\boldsymbol{x}}}_k - \boldsymbol{J}_k\boldsymbol{\rho}_{k-1}\right)$ to avoid discontinuities between priority levels.

Algorithm (22) ends with the N-th iteration with the solution manifold $S_N = \{\dot{\boldsymbol{y}} = \boldsymbol{\rho}_N + \boldsymbol{Q}_N \dot{\boldsymbol{z}}_N \, ; \, \forall \dot{\boldsymbol{z}}_N\}$. We recall from [41], [48] that the algorithm requires a final task that minimizes the control vector $\dot{\boldsymbol{y}}$ in order to ensure the continuity (due to the general time variability of the activation functions and non-orthogonality of the matrices $\boldsymbol{Q}_k$). This leads to the following final velocity control vector

$$\dot{\boldsymbol{y}} = \arg \mathrm{R}\text{-} \min_{\dot{\boldsymbol{y}} \in S_N} \|\dot{\boldsymbol{y}}\|^2 = \boldsymbol{\rho}_{N+1}. \tag{23}$$

G. Vehicle Velocity Tracking Error Compensation Scheme

The above outlined task hierarchy resolution jointly considers both the vehicle velocity $\boldsymbol{v}$ and the arm velocity $\dot{\boldsymbol{q}}$ as optimization variables in the stacked vector $\dot{\boldsymbol{y}}$. However, it is well known that the dynamics of motors or hydraulic actuators are much quicker and cleaner than those of the vehicle thrusters. Indeed, thrusters have been well described dynamically [51]–[53] and their nonlinear properties are much more prominent than those of the arm joints, making it more difficult to obtain an accurate velocity control of the vehicle. Furthermore, the vehicle is also far more massive than the manipulator joints, and this difference is usually exaggerated by lower ratios of force/mass for the vehicle than for the manipulator d.o.f..

To reduce the effects of the inevitable vehicle velocity tracking errors, the idea is to use the above procedure just for setting the vehicle reference velocity, while adding, in parallel to it, an arm control law parametrized by the actual vehicle velocity. This allows the arm to always have as reference joint velocities the best ones tuned on the actual vehicle velocity. This can be done by considering the same task hierarchy but with just the arm velocities as optimization variables, and the vehicle velocity as a parameter [16]. The output of this procedure is an arm control law of the type:

$$\dot{\boldsymbol{q}} = \widetilde{\boldsymbol{\rho}}_{N+1} + \boldsymbol{P}_{N+1} \boldsymbol{v}, \tag{24}$$

where the last $N+1$ iteration is now the arm motion minimality (in analogy with (23)) and where $\boldsymbol{P}_N$ can be computed by using the following iterative formula, for $i = 1, \ldots, N+1$

$$\boldsymbol{P}_k = \left(\boldsymbol{I} - \boldsymbol{Q}_{k-1}(\boldsymbol{J}_k \boldsymbol{Q}_{k-1})^{\#,\boldsymbol{A},\boldsymbol{I}} \boldsymbol{W}_k\right) \boldsymbol{J}_k^a - \boldsymbol{Q}_{k-1}(\boldsymbol{J}_k \boldsymbol{Q}_{k-1})^{\#,\boldsymbol{A},\boldsymbol{I}} \boldsymbol{W}_k \boldsymbol{J}_k^v, \tag{25}$$

where each Jacobian $\boldsymbol{J}_k$ has now been considered as its two separate arm and vehicle contributions, i.e. $\boldsymbol{J}_k \triangleq \begin{bmatrix} \boldsymbol{J}_k^a & \boldsymbol{J}_k^v \end{bmatrix}$, and $\widetilde{\boldsymbol{\rho}}_N$ is the output of the algorithm (22) once only the arm-related part $\boldsymbol{J}_k^a$ of the Jacobians has been used.

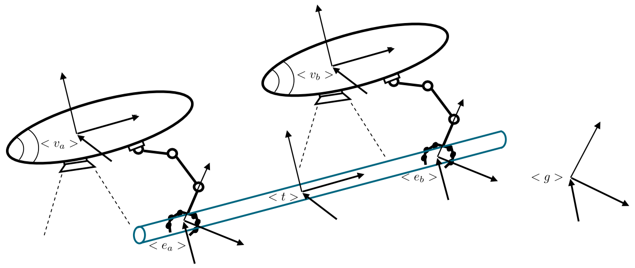


Fig. 3. The two cooperative UVMS and their relevant frames

As already noted, the resulting arm control law (24), since parametrized by the vehicle velocity $\boldsymbol{v}$, results to be the optimal one in correspondence of *any* vehicle velocity. Thus, if a good velocity feedback is available from the vehicle's sensors, then it can be substituted in (24) allowing the arm to behave in an optimal way despite the mismatches between the reference and actual velocity of the vehicle. However, note how such an optimal control law permits the arm to totally (otherwise only partially) compensate the vehicle velocity mismatches under three conditions: i) the arm tool-frame Jacobian is full-rank, i.e. its manipulability measure is currently above some minimum value, as required by its corresponding control objective, ii) the arm joints are not hitting a mechanical limit, as required by its corresponding control objective, iii) the vehicle velocity is actually measurable.

IV. Cooperative Control of Multiple UVMSs

In this section we shall tackle the extension of the above outlined control framework to the case of cooperating UVMSs, when transporting and deploying a grasped, shared object. For the sake of simplicity, the discussion is limited to two UVMSs, even though it can be easily extended to more than two agents.

In this case, by assuming a firm object grasping by part of both agents, we have that the tool frames $\langle t_a \rangle$, $\langle t_b \rangle$ can be respectively assigned by the agents to coincide with the shared object fixed frame $\langle o \rangle$, that is $\langle t_a \rangle \equiv \langle t_b \rangle \equiv \langle o \rangle \triangleq \langle t \rangle$, as exemplified in Fig. 3.

In these conditions, the firm grasp assumption also imposes

$$\dot{\boldsymbol{x}}_t = \boldsymbol{J}_{t,a} \dot{\boldsymbol{y}}_a = \boldsymbol{J}_{t,b} \dot{\boldsymbol{y}}_b, \tag{26}$$

with $\dot{\boldsymbol{x}}_t$ the object velocity with components on $\langle t \rangle$ and $\boldsymbol{J}_{t,a}$, $\boldsymbol{J}_{t,b}$ the system Jacobians with respect to $\langle t \rangle$ with output components on it. Then, by considering the second equation in the above we get

$$\begin{bmatrix} \boldsymbol{J}_{t,a} & -\boldsymbol{J}_{t,b} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{y}}_a \\ \dot{\boldsymbol{y}}_b \end{bmatrix} \triangleq \boldsymbol{G}\dot{\boldsymbol{y}} = \boldsymbol{0}, \iff \dot{\boldsymbol{y}} \in \ker(\boldsymbol{G}), \tag{27}$$

which represent the subspace where $\dot{\boldsymbol{y}}$ is constrained to lay as a consequence of the firm grasp assumption.

However, note how as a consequence of the full-row rankness of both $\boldsymbol{J}_{t,a}$, $\boldsymbol{J}_{t,b}$ (certainly guaranteed by the full actuation assumption of both vehicles), the space of the achievable object velocities remains $\mathbb{R}^6$, i.e. $\dot{\boldsymbol{x}}_t \in \mathbb{R}^6$, despite the presence of the above evidenced system velocities constraints.

Clearly, if no other objective other than the end-effector velocity tracking is taken into account, then the considered coordination problem results simplified, as for example it has been considered in [54]. However, as already stated, many other objectives must be considered for both the safety and the good operability of the systems.

At least in principle, a cooperative task hierarchy problem accounting for the merging of both the individual task hierarchies could be solved via the same procedure outlined in Section III, by just starting the overall optimization procedure from $\dot{\boldsymbol{y}} \in \ker(\boldsymbol{G})$ rather than from the unconstrained space $\mathbb{R}^{n_a+n_b}$. As in the individual case, this would consequently result into an optimal $\dot{\boldsymbol{y}}$ satisfying the constraints, whose corresponding object velocity $\dot{\boldsymbol{x}}_t$ would however not be the same, in general, of the desired one $\dot{\bar{\boldsymbol{x}}}_t$ because of the low priority assigned to the tool-frame position control task, for the same reasons outlined in Section III-E for individually operating agents.

In any case, since the centralized approach would require the exchange of all task reference rates and Jacobians, in the underwater domain it would be infeasible due to the heavy bandwidth restrictions. For this reason, we now present an alternative way of facing this coordination problem in a decentralized manner, with a very limited information exchange, even if generally leading to a suboptimal, best-effort, solution.

The developed control policy is based on three steps:

1) the first step is an independent optimization carried out by the two UVMSs, with the sole goal of finding out what would be their actual optimal tool velocity at the end of the task hierarchy optimization, whenever each one were solely operating;
2) in the second step the two systems exchange their computed tool velocities, and thanks to an a priori established fusion policy, they agree on common tool frame velocity;

3) the last step is again an independent optimization, but with a different task hierarchy for both, where now the two system impose, with highest priority, the agreed tool frame velocity, which just for this reason can be therefore exactly assigned.

A. First Independent Optimization

In this first step the two UVMSs perform an independent optimization, as if each one of them were the only one acting on the object. In this case, they both solve a task hierarchy consisting of the individual objectives presented in Section II-D plus the vehicle distance objective, thus deliberately neglecting the kinematic constraint.

In this context, at each time instant, the two UVMSs obtain, in a fully decentralized way, the couple of system control actions $\dot{\boldsymbol{y}}_a$, $\dot{\boldsymbol{y}}_b$, and the corresponding, separately evaluated, tool frame velocities

$$\begin{aligned} \dot{\boldsymbol{x}}_{t,a} &= \boldsymbol{J}_{t,a}\dot{\boldsymbol{y}}_a, \\ \dot{\boldsymbol{x}}_{t,b} &= \boldsymbol{J}_{t,b}\dot{\boldsymbol{y}}_b. \end{aligned} \tag{28}$$

In general, $\dot{\boldsymbol{x}}_{t,a} \neq \dot{\boldsymbol{x}}_{t,b}$ due to the different optimization conditions of the two systems. This means that $\dot{\boldsymbol{y}}_a$, $\dot{\boldsymbol{y}}_b$ do not generally fulfil the firm grasp kinematic constraints. Thus, in order to eliminate the drawback that could arise whenever $\dot{\boldsymbol{y}}_a$, $\dot{\boldsymbol{y}}_b$ were applied as system velocity references, possibly leading to unwanted object stresses, the execution of the second step of the procedure is proposed as hereafter described.

B. Exchange of Tool-frame Velocities

The second step of the proposed coordination architecture requires the exchange of the tool-frame velocities $\dot{\boldsymbol{x}}_{t,a}$, $\dot{\boldsymbol{x}}_{t,b}$ of the previous step. After the exchange of these velocities, the two systems can separately compute a common tool-frame velocity

$$\dot{\hat{\boldsymbol{x}}}_t = \boldsymbol{f}(\dot{\boldsymbol{x}}_{t,a}, \dot{\boldsymbol{x}}_{t,b}), \tag{29}$$

where $\boldsymbol{f}$ is an a priori agreed fusion policy function. As an example, a convex combination can be employed. Indeed, in our simulation we have used the mean as fusion function.

The reason for this choice is that we consider the independently evaluated tool frame velocities $\dot{\boldsymbol{x}}_{t,a}$ and $\dot{\boldsymbol{x}}_{t,b}$ as a suitable way to concisely represent the current states of the two UVMSs and their ability to track the desired original object velocity vector $\dot{\bar{\boldsymbol{x}}}_t$. For example, if both $\dot{\boldsymbol{x}}_{t,a}$, $\dot{\boldsymbol{x}}_{t,b}$ are equal to the desired tool frame velocity $\dot{\bar{\boldsymbol{x}}}_t$, it means that both UVMSs have enough deactivated higher priority tasks such that the desired object velocity $\dot{\bar{\boldsymbol{x}}}_t$ can be separately assigned by both of them.

Conversely, when one of the tool-frame velocities differs from the desired one $\dot{\bar{\boldsymbol{x}}}_t$, this means that the corresponding UVMS has still different higher priority active tasks which prevent the exact tracking of $\dot{\bar{\boldsymbol{x}}}_t$. Furthermore, the norms $\|\dot{\bar{\boldsymbol{x}}}_t - \dot{\boldsymbol{x}}_{t,a}\|$, $\|\dot{\bar{\boldsymbol{x}}}_t - \dot{\boldsymbol{x}}_{t,b}\|$ represent a sort of measure of the difficulty in tracking the desired object velocity and could be exploited in the fusion policy as a way to favour the agent with higher difficulty in tracking the object velocity reference.

The computation of the common $\dot{\hat{\boldsymbol{x}}}_t$ is a way to move the object with a velocity that implicitly considers the safety and operational-enabling tasks of both system, avoiding the direct employment of the kinematic constraint and that requires the minimal amount of information exchange. The resulting algorithm is a best-effort strategy, as it cannot guarantee the instantaneous tracking of all the safety tasks of both systems.

The above outlined fusion policy could be enhanced through the exchange of the spans of the end-effector tasks Jacobians at their given priority level, i.e. inclusive of the projection matrix which could effectively limit their spans. On the one hand, this enhancement would allow to guarantee that the resulting common velocity would be compatible with the internal situation of both systems. Indeed, by selecting an object reference velocity within the intersection of the spans of both end-effector tasks, we would be able to guarantee that despite the subsequent change of priority in the second optimization, the original higher level tasks would be still satisfied. Thus the resulting algorithm would not be anymore a simple best-effort strategy. On the other hand, it would require an increase in communication bandwidth (36 more numbers), with a consequent reduction of the exchange rate. This possibility will be investigated thoroughly as part of future works.

C. Second Independent Optimization

After the exchange of their optimal independent tool-frame velocities and the computation of the common one $\dot{\hat{\boldsymbol{x}}}_t$, the two UVMSs must now separately compute their system velocities, with the task of tracking of the $\dot{\hat{\boldsymbol{x}}}_t$ at the top of the hierarchy.

This approach allows to also subsequently further optimize the internal motions of the system for better achieving all the other tasks, for a given $\dot{\hat{\boldsymbol{x}}}_t$. Following this idea, the two UVMSs thus obtain a new couple of system velocity vectors $\dot{\hat{\boldsymbol{y}}}_a$ and $\dot{\hat{\boldsymbol{y}}}_b$ which guarantee

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}}_t &= \boldsymbol{J}_{t,a}\dot{\hat{\boldsymbol{y}}}_a, \\ \dot{\hat{\boldsymbol{x}}}_t &= \boldsymbol{J}_{t,b}\dot{\hat{\boldsymbol{y}}}_b, \end{aligned} \tag{30}$$

and thus satisfy the object kinematic constraint (26).

Remark: Since the end-effector task is at the top of the hierarchy, and the tracking of its velocity reference is guaranteed for the already stated reasons, one could ask why we have not used directly the original object reference rate $\dot{\bar{\boldsymbol{x}}}_t$. The difference lies in the fact that $\dot{\bar{\boldsymbol{x}}}_t$ only depends on the current tool frame $\langle t \rangle$ and goal frame $\langle g \rangle$ positions, while $\dot{\hat{\boldsymbol{x}}}_t$ depends, indirectly, also on all the task references of both UVMSs because it is the fusion (29) of the best Cartesian tool frame velocities (28) that each system can independently obtain considering all their other higher priority tasks.

D. Compensation of Vehicle Velocity Tracking Errors

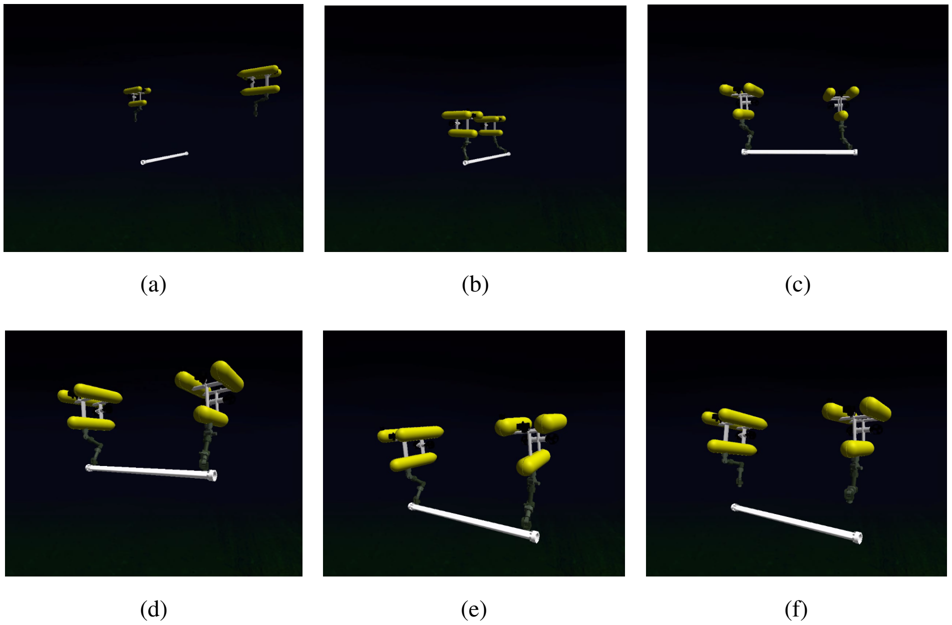
In the above sections we have presented the cooperative control scheme that the two UVMSs employ to execute a reference transportation mission requiring a limited information exchange. In the second optimization, the end-effector task is the highest priority one, in order to guarantee that the agreed tool frame velocity reference $\dot{\hat{\boldsymbol{x}}}_t$ is exactly followed. However, for the same considerations made in Section III-G about the vehicle thruster dynamics, it is clear that, in general, (30) is not satisfied.

To mitigate this problem, it is possible to apply the same technique presented in Section III-G, i.e. to run in parallel an optimization based only on the arm control variables, taking the vehicle velocity as a parameter. Of course, the same conditions outlined in Section III-G must hold.

V. SIMULATION RESULTS

In this section we present some results obtained in a simulation environment. In the scene we have a pipe object, whose reference frame $\langle o \rangle$ is placed at its center and is located in $\begin{bmatrix} 0.5 & 0.5 & 3 \end{bmatrix}$ w.r.t. a world inertial frame with the NED coordinates (thus z represents the depth and is positive going downwards). The two UVMSs are required to grasp the pipe at a distance of 1.5 m from its center, along the longitudinal axis.

Two kind of simulations are presented. In Section V-A we present a simplified simulation, where the UVMS is assumed kinematic. This allows to simulate a whole mission, where all



(a) (b) (c)

(d) (e) (f)

Fig. 4. Screenshots of the simulation environment as the transportation mission is executed: (a) initial position $t = 0$ (b) successful grasp of the objects by both UVMS $t = 13.71$ (c) transportation toward the goal position $t = 20$ (d) pre-deployment position reached $t = 39.3$ (e) deployment of the pipe $t = 45$ (f) deployment completed $t = 55$.

the phases (navigation, grasping, transportation and deployment) are present. The successive Section V-B instead focuses only on the problem of transportation, where the two UVMS are now coupled due to the firm grasp assumption. In that section, the kinematic assumption is relaxed, and an hydrodynamic model is used instead. Furthermore, the algorithm is validated considering different communication parameters, including half-duplex communication, limited bandwidth and latencies.

A. Kinematic Simulations of the Overall Mission

In this simulation, in order to focus our attention on the execution of the whole mission, we make the following assumptions:

- the UVMSs are assumed kinematic;
- the UVMSs know their position and the goal position for the object at each sampling time;
- the UVMSs know their relative position (for the distance task);
- the communication channel is assumed error free and able to deliver the exchanged velocities at each control interval.

Some of these assumptions will be relaxed in the next section.

The simulated mission requires that, after the successful grasp of the object, the two UVMSs should bring the pipe in a pre-deployment position, located at $\begin{bmatrix} 3 & 5 & 2 \end{bmatrix}$ with a rotation along the z axis of the world frame of $\pi/2$, before deploying the object at the seafloor depth of 3 m. Furthermore, the missions phases presented in Section II-C have been implemented. For all the phases, the list of prioritized tasks is, for the individual UVMS control and for the cooperative algorithm first independent optimization, the one already outlined in Section III-E. For the second independent optimization of the cooperative algorithm, the same priority list is used, with the addition of end-effector task at the top of the hierarchy.

The difference between the various phases is which of these tasks is active or not. The joint limits, minimum vehicle distance, manipulability and horizontal attitude tasks are always active; the camera centering task is only active during the grasping phase, while the vehicle position control is active only during the initial navigation phase. The end-effector position control is activated since the grasping phase and remains active for the rest of the mission. Finally, the tasks to minimize the arm or vehicle movements are alternatively activated: the arm movements are minimized during the navigation and transportation phases, while the vehicles movement are minimized during the grasping and deployment ones. Table I summarizes the phase-dependant part of the activation values. The transitions $0 \rightarrow 1$ and vice versa are implemented with sigmoid functions parametrized with the phase's time, i.e. they complete their transition within a couple of seconds since the start of their respective phase.

The above list represent which task are activated depending on the current phase of the system, i.e. the value of their corresponding $a^p_{(i)}$. Of course, for tasks that correspond to inequality control objectives, this does not immediately means that their corresponding activation function $a_{(i)}$ is equal to one, since it still depends on the actual value of their corresponding $a^i_{(i)}$. Indeed for such tasks we have that $a_{(i)} = a^i_{(i)}$. As an example, the camera centering *can* be active during the grasping phase, but if the error vector is within the pre-established thresholds, the task can have its activation value equal to zero.

In addition to the phases presented in Section II-C, we have added a small transition phase between the grasping and transportation phases, to introduce the switch between single UVMS control and the cooperative control policy. In particular, during this phase, the end-effector position control of the second independent optimization is activated at the top of the hierarchy, and smoothly deactivated at its original position. A similar, inverse, transition has been implemented at the end of the deployment of the object. The simulation also consider a final phase, after the

TABLE I
PHASE-DEPENDANT ACTIVATION VALUES

Task	Navigation	Grasping	Transition	Transportation	Deployment	Transition	Step-back
Joint Limits	1	1	1	1	1	1	1
Vehicle Distance	1	1	1	1	1	1	1
Manipulability	1	1	1	1	1	1	1
Camera Centering	0	$0 \rightarrow 1$	$1 \rightarrow 0$	0	0	0	0
Horizontal Attitude	1	1	1	1	1	1	1
Vehicle Pos. Control	1	$1 \rightarrow 0$	0	0	0	0	0
End-effector Pos. Control	0	$0 \rightarrow 1$	1	1	1	1	1
Min. Arm Velocity	1	$1 \rightarrow 0$	$0 \rightarrow 1$	1	$1 \rightarrow 0$	0	0
Min. Vehicle Velocity	0	$0 \rightarrow 1$	$1 \rightarrow 0$	0	$0 \rightarrow 1$	1	1

object has been correctly placed on the seafloor, where the two UVMSs' end-effectors rise up from the object of around 0.5 m, now again acting independently from each other.

Figure 4 shows some screenshots of the simulation environment (we have used UWSim [55] for the rendering of the scene), showing all the main phases of the mission. Figure 5 presents the generated system velocity references for both systems. It is important to remark how the generated velocities are free from discontinuities during each phase and especially from chattering phenomena, considering that the simulation implements a discrete control. The only discontinuities that can be seen are those in correspondence of the activation and deactivation of the firm grasp constraint, which is an intrinsically discontinuous event.

In the figures we have highlighted the times at which there is a change of a mission phase: $t = 5.95$ is the end of navigation phase and the start of the grasping phase, which ends at $t = 13.2$. Then a small transition window allows the two UVMSs to start the coordinated control ($t = 15.7$). The two UVMSs reach a pre-deployment position at $t = 39.3$, and complete the deployment of the pipe on the seafloor at $t = 45.04$. A small transition window is again present to allow the UVMSs to switch from the coordinated control policy to the single free floating control ($t = 47.5$). Finally the two UVMSs move their respective end-effectors away from the pipe and the simulation is ended at $t = 55$.

The successive Fig. 6 depicts the time history of the activation values of each task for both systems. In particular we note how many different task are activated and deactivated throughout the mission execution:

- the minimization of the sub-system velocities of the arm and the vehicle are alternatively activated and deactivated depending on the mission phase. This leads to the graphs of Fig. 5 where the arm is mostly used during the grasping and deploying tasks, while the vehicle is mostly used for accomplishing the transportation task;
- the camera centering task, which is an inequality control objective, is activated only during the grasping phase, however its value depends on the actual misalignment error and thus the activation value is not strictly one.

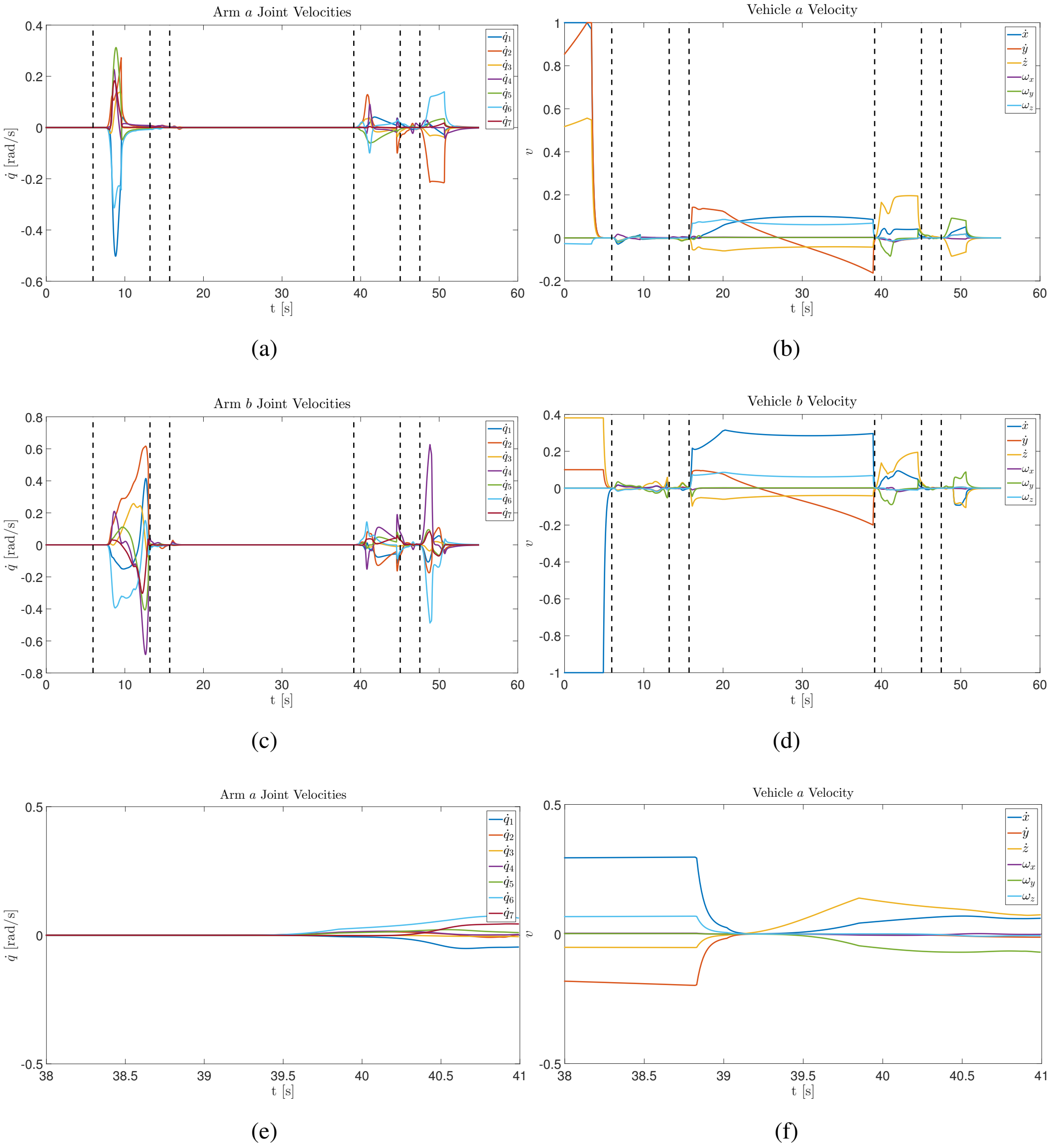
In the second simulation, the same virtual scene is used, but this time a sinusoidal velocity disturbance is added to the vehicles' velocities. The disturbance is directed as the inertial frame x axis, simulating an ocean time varying current. In this simulation, the disturbance compensation approach of Section IV-D is implemented. This means that the arm is using the real vehicle velocity (assumed measurable) as a parameter for the computation of its parametrized control law (24). Figure 7 shows the obtained system velocities in this case. It is noteworthy to highlight how in this particular case, during the transportation phase $22.99 < t < 47.85$ the arm velocity is not zero as in Fig. 5, due to the fact that the arm is actively compensating the mismatches between the required vehicle velocity and the actual one. Finally, Fig. 8 reports the time history of the activation values of both UVMSs.

B. Hydrodynamic Simulations of the Cooperative Transportation Phase including Communication Constraints

In this section we focus on the transportation phase and we show the performances of the proposed coordination scheme under more realistic assumptions than those used in the previous section. In particular, in these simulations we have implemented an hydrodynamic model of the two UVMSs, relaxing the assumption of kinematic UVMS.

Different communication schemes have been tested:

- a full duplex communication at 100 Hz with no latency, which represents the case of optical modems;
- a full duplex communication at 1 Hz with 1 second latency, which corresponds to the case where two acoustic modem, operating at different frequency are employed;



(a)

(b)

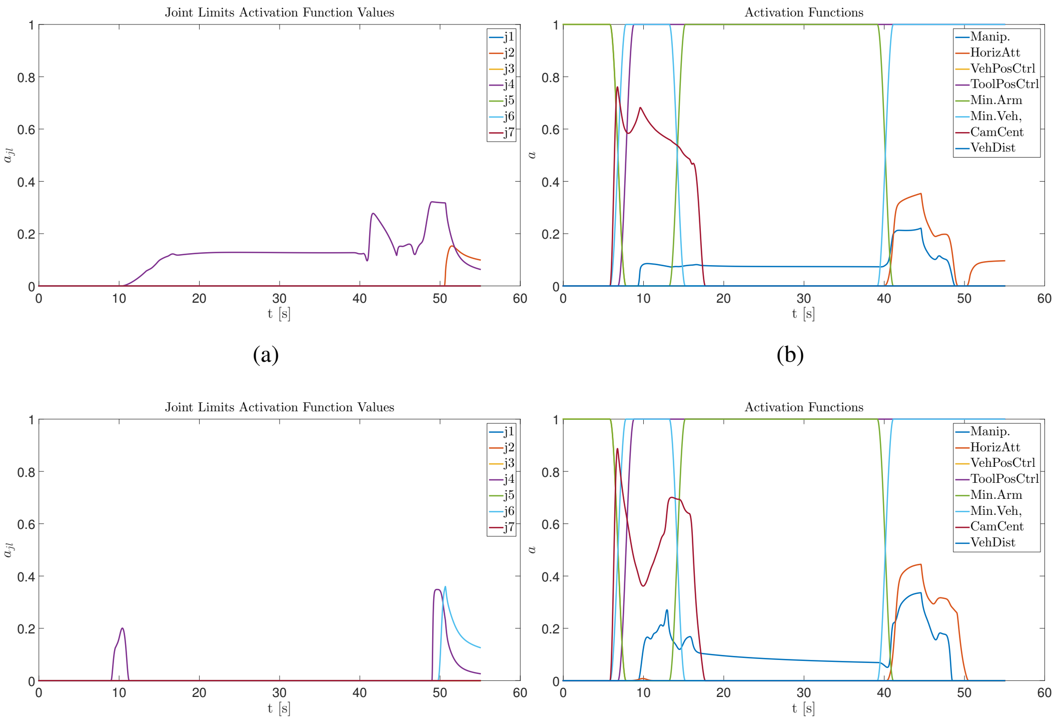
(c)

(d)

(e)

(f)

Fig. 5. The system velocities during the first simulation trial, showing that the system does not suffer from discontinuities or chattering phenomena, except at the activation or deactivation of the firm grasp constraint. The vertical dashed lines separate the different mission phases. Time history of (a) arm a joint velocities $\dot{\boldsymbol{q}}$ (b) vehicle b velocities $\boldsymbol{v}$, (c) arm b joint velocities $\dot{\boldsymbol{q}}$ (d) vehicle b velocities $\boldsymbol{v}$ (e) a zoom of the arm b joint velocities and (f) of the vehicle b velocities in the interval $38 < t < 41$ to show the continuity of the approach.



(a) (b)

(c) (d)

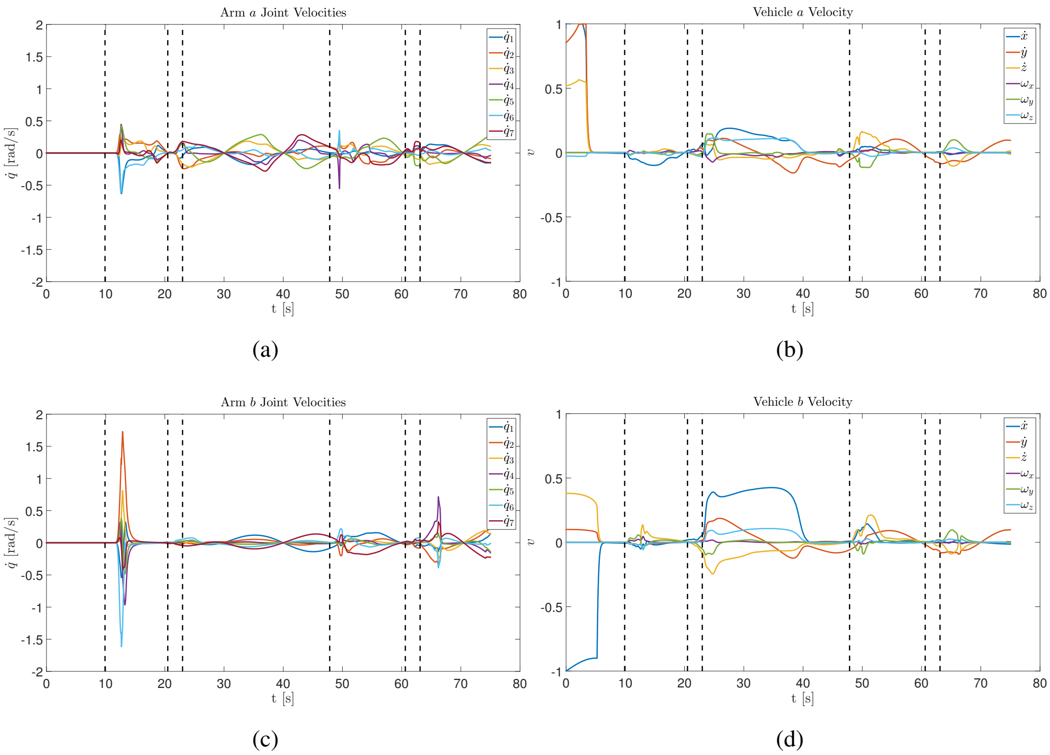
Fig. 6. The values of the activation functions during the first trial. Time history of (a) arm a joint limits activation values and (b) the activation values of all the other tasks for UVMS a (c) arm b joint limits activation values and (d) the activation values of all the other tasks for UVMS b.

- an half duplex communication at 1 Hz with 1 second delay, which corresponds to using a single acoustic modem.

The hypothesis that the communication channel can sustain the exchange of information at each control interval is thus relaxed, at least in some of the simulated cases. However, it should be noted that we have assumed the channels without errors.

Regarding the simulation parameters, they have been chosen to replicate those of the arm and vehicle employed within the MARIS project. In particular, vehicle parameters are a slightly scaled down version of those reported in [56], since the vehicle used in MARIS is an improved version of the original Romeo vehicle; arm parameters are instead those of the electric manipulator bought by University of Genova for the MARIS project, whose basic concept has been previously developed within the TRIDENT project [57]. In summary, the parameters are:

1) vehicle of length, width and height of 1 m;
2) vehicle mass of 300 Kg, arm link masses of $\begin{bmatrix} 8 & 6 & 4 & 3 & 3 & 3 & 2 \end{bmatrix}$ Kg and object mass



(a)

(b)

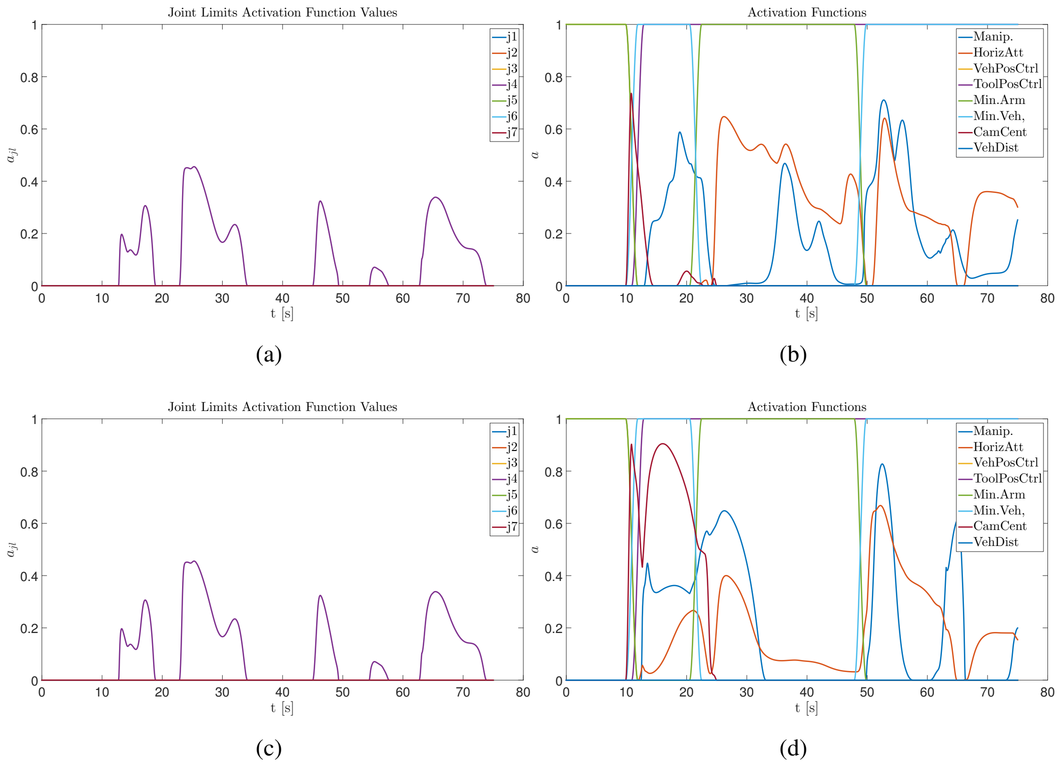
(c)

(d)

Fig. 7. The system velocities during the second simulation trial, showing that the system does not suffer from discontinuities or chattering phenomena, except at the activation or deactivation of the firm grasp constraint. The vertical dashed lines separate the different mission phases. Time history of (a) arm a joint velocities $\dot{\boldsymbol{q}}$ (b) vehicle b velocities $\boldsymbol{v}$, (c) arm b joint velocities $\dot{\boldsymbol{q}}$ and (d) vehicle b velocities $\boldsymbol{v}$.

of 28 Kg (all weights are reported in-air);

3) given the above two points, the vehicle's inertia tensor can be easily found. For the arm, we have modelled each link as a cuboid of length $\begin{bmatrix}0.108 & 0.105 & 0.3265 & 0.095 & 0.325 & 0.132 & 0.021\end{bmatrix}$ m where width and depth have been set to 0.1 m for all links, and we have found the inertia matrices along the inertia axes accordingly;
4) additional hydrodynamic parameters of the vehicle are reported in Table II, where for the added mass and drag terms we have assumed a simplified diagonal form;
5) buoyancy is set such that the different items are slightly positive;
6) dynamic control loops (PI loops, tuned around the nominal inertia matrix values) and simulation integration running at a frequency of 2 KHz;
7) kinematic control loop running at a frequency of 100 Hz;
8) vehicle velocity feedback received at the frequency reported in the Table III (100 or 10 Hz);



(a) (b)

(c) (d)

Fig. 8. The values of the activation functions during the second simulation trial. Time history of (a) arm a joint limits activation values and (b) the activation values of all the other tasks for UVMS a (c) arm b joint limits activation values and (d) the activation values of all the other tasks for UVMS b.

9) communication exchanges occurring as reported in the Table III;
10) for simplicity, thruster dynamics have not been modelled, and the control input for the vehicle is assumed to be a force/torque vector.

The last assumption requires some clarification. The proposed algorithms for single and cooperative UVMS control (Section III and Section IV) takes into account, in a global manner, the differences in the dynamic response of the two main subsystems (arm and vehicle). Indeed, the vehicle velocity tracking errors compensation scheme presented in Section III-G and Section IV-D allows the arm to behave in an optimal way despite the vehicle inaccuracy in tracking the required velocity, independently of the original *causes* (e.g. inertia or thruster dynamics). In the simulations that are shown hereafter, we have attributed these tracking errors only to the inertia of the vehicle and its dynamic control. More accurate simulations, taking into account a precise modelling of the thruster dynamics, would have not changed the structure of the proposed kinematic controller, since it already takes into account the resulting *effects*.

TABLE II
ADDITIONAL HYDRODYNAMIC SIMULATION PARAMETERS

Variable	Linear components	Angular components
Vehicle added mass	$\begin{bmatrix}158.4 & 137.1 & 171.3\end{bmatrix}$ Kg	$\begin{bmatrix}15 & 20 & 20\end{bmatrix}$ Kg m^2
Vehicle drag terms	$\begin{bmatrix}50 & 50 & 44\end{bmatrix}$ N s/m	$\begin{bmatrix}20 & 20 & 24\end{bmatrix}$ Nm s/rad
Vehicle quadratic drag terms	$\begin{bmatrix}320 & 320 & 430\end{bmatrix}$ N s^2/m^2	$\begin{bmatrix}40 & 40 & 31\end{bmatrix}$ Nm s^2/rad^2

Regarding the realism of the dynamic control loop and its gains, during the simulation it generates maximum forces (along a particular axis) roughly around 150-170 N, usually during the initial phase of the transportation where most of the acceleration is occurring, while mean forces are much lower (around 30-50 N). Such forces are within the range of commercial thruster systems. Finally, let us remark that experimental trials on single UVMS control using the approach proposed in Section III, with PI loops at dynamic level, have been performed and a video showing the behavior of the system during the approaching and grasping phases can be seen at the link: https://youtu.be/b3jJZUoeFTo.

The simulations start after successful grasp of the object by the two UVMSs. The mission requires them to bring the pipe in a pre-deployment position, located at $\begin{bmatrix}0.75 & 0.75 & 2.5\end{bmatrix}$ with a rotation along both the y and z axis of the world frame of $\pi/2$, before deploying the object at the seafloor depth of 3 m.

The most relevant results of these simulations are summarized in Table III, where the performance of the proposed coordination scheme are highlighted in terms of maximum, mean and standard deviation of the interaction forces and moments acting on the grasped object, as the different parameters are tested. However, let us highlight that the force and moment interaction values should be used to compare the different control and communication schemes between each other, rather than as absolute reference values. This is due to some of the simplifying assumptions that were made, but more importantly because the focus of the paper is on a general control methodology rather than a precise dimensioning of a particular system.

Let us now comment the results reported in Table III. First of all T1 and T2 show that if the two systems do not communicate and they do not implement a vehicle velocity tracking error compensation technique the simulation aborts due to too high forces generated and thus the two

TABLE III

HYDRODYNAMIC SIMULATION RESULTS

								Inter. force [N]			Inter. moment [Nm]		
Test	Vel. Comp.	Comms.	Latency	Filter	Force Mod.	Vel. Fbk.	Final error	Max	Mean	Std dev.	Max	Mean	Std dev.
T1	NO	NO	N/A	N/A	NO	100	N/A	N/A	N/A	N/A	N/A	N/A	N/A
T2	YES	NO	N/A	N/A	NO	100	0.0059	63.29	41.53	18.09	25.78	13.52	5.9
T3	YES	FD 100 Hz	NO	NO	NO	100	0.0039	41.08	26.52	12.81	16.09	10.86	4.7
T4	YES	FD 100 Hz	NO	YES	NO	100	0.004	9.9	1.78	1.72	5.13	0.99	0.75
T5	YES	FD 1 Hz	YES	YES	NO	100	0.0258	10.1	5.42	2.53	6.71	1.87	0.81
T6	YES	HD 1 Hz	YES	YES	NO	100	0.0305	78.5	29.03	17.29	14.55	5.11	3.34
T7	YES	NO	N/A	N/A	YES	100	0.0077	9.64	2.66	2.25	16.02	5.99	2.52
T8	YES	FD 100 Hz	NO	YES	YES	100	0.0039	5.03	0.89	0.9	5.7	0.82	0.92
T9	YES	FD 1 Hz	YES	YES	YES	100	0.0259	5.07	1.14	1.12	5.63	1.19	0.8
T10	YES	HD 1 Hz	YES	YES	YES	100	0.0109	13.42	3.19	2.75	8.17	2.16	1.36
T11	YES	HD 1 Hz	YES	YES	YES	10	0.0347	24.81	4.05	4.37	6.92	2.13	1.35

system cannot complete the desired transportation mission. This highlights the importance of the proposed vehicle velocity tracking compensation techniques, discussed in Section IV-D.

The mean value of the interaction force and moment of T2 is due to the integral terms that are present at the dynamic level. Indeed, every velocity error that is not coherent with the kinematic constraint is constantly integrated, without reducing the corresponding error. There are two causes for having errors that are not coherent with the constraint:

- the reference velocities were not coherent in the first place. This certainly may happen when the two systems do not communicate at all.
- the dynamics and the current configuration of the two system lead to instantaneously different Cartesian velocities at the end-effector.

T3 introduces a full-duplex communication between the two UVMS at every control loop, without latencies. The comparison with T2 shows a significant reduction in the maximum and mean forces. This is due to the fact that at every sampling interval the two system agree on a common velocity. This situation could be implemented through the use of optical modems, and represents the best situation from the communications point of view. However, despite the reference velocities are now coherent with the firm grasp constraint, the two system dynamics still lead to the problem of the integration of non coherent joint errors.

The introduction of a filter on the Cartesian velocities in T4 reduces the problem of the system dynamics, just because the requested motion is slower. Indeed, the reference velocity is now followed much better and the resulting interaction is far lower.

T5 introduces communication latency, i.e. the fact that a single message needs a certain processing and propagation time before it is received by the other system. We have used 1 second latency for our simulations. This also means that the two system communicate only *once* per second, as opposed to once per control loop (i.e. 100 times in one second). This situation reflects the possibility of using two acoustic modems at different frequency, in order to have them communicate at the same time.

T6 instead shows the performance of the cooperation when the communication is only half-duplex, i.e. when only one acoustic modem is used. In this case the two systems must alternate themselves in sending the computed velocity. This introduces the problem of having the optimal velocities of the two systems referred to two different time instants. In fact, the interaction results quite increased w.r.t. the full-duplex case, but still it is lower than the case of no communication between the two systems.

The successive four tests, T7, T8, T9 and T10 assume the possibility of measuring the interaction force and report the results with different communication parameters as done before. This information is used to modify the reference velocity for end-effector, adding a small component along its direction. The availability of this information reveals extremely useful for reducing the interaction on the object, especially it prevents the ”charging” of the integral terms outlined before. Naturally, the possibility of communication still allows to improve the performances. However, it must be noted that measuring the interaction force is not an easy task, as typically one only measures the force/torque at each of the arms’ wrists. Further investigation on how to compute and exploit the interaction force or the one measured at the wrist of each arms is currently undergoing.

Finally, T11 shows the reduction in performances if the vehicle feedback is available only at 10 Hz rather than 100 Hz. This reduction is evident mainly in terms of the maximum values of the interaction forces.

VI. CONCLUSIONS

This paper has presented a cooperative control policy for UVMSs employed for manipulation and transportation tasks. The algorithm is explicitly taking into account the heavy bandwidth restrictions that the underwater environment poses on the communication between the robots. For that reason, the algorithm makes use of a very limited amount of exchange of information between the UVMSs since only requiring the exchange of the tool-frame velocities that the two systems would have if they were the only one acting on the object to be transported. This exchange reveals necessary in order to allow each agent to globally understand what are the impacts of the current system configurations (i.e. possible joints near their limit, a bad manipulability value and so on) on the obtainable tool-frame Cartesian velocity.

A pre-defined fusion policy is used to compute a new common reference that is now tracked by the two systems with the highest priority, in order to avoid possible unwanted stresses on the object. With this approach the kinematic constraints are implicitly taken into account, and, coherently with them, system velocity reference signals are generated in a distributed manner. Thus, the proposed approach takes into account all the safety and operational constraints that each agent needs to fulfil, which is a clear distinction w.r.t other works on the subject such as [54], and also presents an efficient technique for the cooperative execution of the reference mission, with a very limited amount of information exchange required.

A complete mission simulation is presented to support the proposed approach, which contains all the phases of the reference manipulation and transportation mission. The vehicles first navigate to approximate object positions, then proceed to perform the grasp. After the grasp of the pipe-like object, the two system use the proposed cooperative policy to bring the object to a pre-deployment position, and then they execute the final deployment. Furthermore, the same mission is repeated with a velocity disturbance acting on the vehicles, to demonstrate the effectiveness of the proposed disturbance compensation technique.

Furthermore, different hydrodynamic simulations of the two UVMS have been done, narrowed to the cooperative phase in order to analyse the efficacy of the proposed distributed cooperative strategy. These simulations have been done by taking into account different communication schemes, showing reasonable values of interaction forces and moments even when half-duplex channels with latencies are used.

The MARIS project is now focusing on the experimental trials of a single underwater manipulator system [58]. A video of one of these trials, with a successful grasp of the pipe can be seen at the following link: https://youtu.be/b3jJZUoeFTo. Other relevant activities carried out within the project regard the improvement of the vision based techniques for object pose estimation [59], adaptive dynamic control [60], visible light communications [61] and studies on UVMS single range observability [62].

Currently, the possibility of also exchanging the spans of the end-effector matrices is being evaluated, mainly in terms of performance gain versus increased bandwidth requirements. Future works may also include the addition of velocity saturations in the prioritized control, in a similar manner as developed in [63].

Appendix

Vehicles not fully actuated

For simplicity of discussion, the algorithms have been presented under the assumption of fully actuated vehicle. In most of the cases, even work class ROVs might not be controllable in roll and even pitch. Those two d.o.f. are usually made passively stable, with a proper choice of the center of buoyancy w.r.t. the center of mass. Such cases can still be easily tackled with the proposed kinematic task-priority approach with just a slight modification. Rather than initializing the algorithm (22) as in (21), consider the following initial values:

$$\boldsymbol{\rho}_0 = \begin{bmatrix} \mathbf{0}_{l\times 1} \\ \mathbf{0}_{3\times 1} \\ \omega_x \\ \omega_y \\ 0 \end{bmatrix} \quad \boldsymbol{Q}_0 = \begin{bmatrix} \boldsymbol{I}_{l\times l} & \mathbf{0}_{l\times 3} & \mathbf{0}_{l\times 1} & \mathbf{0}_{l\times 1} & \mathbf{0}_{l\times 1} \\ \mathbf{0}_{3\times l} & \boldsymbol{I}_{3\times 3} & \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{1\times l} & \mathbf{0}_{1\times 3} & 0 & 0 & 0 \\ \mathbf{0}_{1\times l} & \mathbf{0}_{1\times 3} & 0 & 0 & 0 \\ \mathbf{0}_{1\times l} & \mathbf{0}_{1\times 3} & 0 & 0 & 1 \end{bmatrix}. \tag{31}$$

In practice, $\boldsymbol{\rho}$ is initialized with the actual angular velocities on the axes that are not actuated, and simultaneously the corresponding values of the initial projection matrix $\boldsymbol{Q}$ are set to zero. Such an initialization effectively inhibits the algorithm from using those d.o.f., thus it will naturally result that the final $\dot{\boldsymbol{y}}$ will have exactly the same value of $\boldsymbol{\rho}_0$ for the not actuated d.o.f.. However, exploiting this initialization of $\boldsymbol{\rho}_0$, the optimization procedure (22) will properly take into account these non-controllable velocities, just due to the term $\left(\dot{\bar{\boldsymbol{x}}}_k - \boldsymbol{J}_k\boldsymbol{\rho}_{k-1}\right)$.

ACKNOWLEDGMENT

This work has been supported by the MIUR (Italian Ministry of Education, University and Research) through the MARIS prot. 2010FBLHRJ project and by the European Commission through H2020-BG-06-2014-635491 DexROV project.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] D. Angeletti, G. Cannata, and G. Casalino, “The control architecture of the amadeus gripper,” *International journal of systems science*, vol. 29, no. 5, pp. 485–496, 1998.

[5] G. Casalino, D. Angeletti, T. Bozzo, and G. Marani, “Dexterous underwater object manipulation via multi-robot cooperating systems,” in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 4. IEEE, 2001, pp. 3220–3225.

[6] 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.

[7] J. Yuh, “Design and control of autonomous underwater robots: A survey,” *Autonomous Robots*, vol. 8, no. 1, pp. 7–24, 2000.

[8] J. Evans, P. Redmond, C. Plakas, K. Hamilton, and D. Lane, “Autonomous docking for Intervention-AUVs using sonar and video-based real-time 3D pose estimation,” in *Oceans 2003*, vol. 4, September 2003, pp. 2201–2210.

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

[10] 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.

[11] G. Marani, S. K. Choi, and J. Yuh, “Underwater autonomous manipulation for intervention missions AUVs,” *Ocean Engineering*, vol. 36, pp. 15–23, 2008.

[12] 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.

[13] 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.

[14] G. Casalino, E. Zereik, E. Simetti, S. Torelli, A. Sperindé, and A. Turetta, "A task and subsystem priority based control strategy for underwater floating manipulators," in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV 2012)*, Porto, Portugal, April 2012, pp. 170–177.

[15] ——, "Agility for underwater floating manipulation task and subsystem priority based control strategy," in *International Conference on Intelligent Robots and Systems (IROS 2012)*, Vilamoura, Portugal, September 2012, pp. 1772–1779.

[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.

[17] A. Peñalver, J. Pérez, J. J. Fernández, J. Sales, P. Sanz, J. García, D. Fornas, and R. Marin, "Autonomous intervention on an underwater panel mockup by using visually-guided manipulation techniques," in *19th world congress of the international federation of automatic control (IFAC)*, 2014, pp. 5151–5156.

[18] D. M. Lane, F. Maurelli, P. Kormushev, M. Carreras, M. Fox, and K. Kyriakopoulos, "Persistent autonomy: the challenges of the pandora project," in *Proceedings of IFAC MCMC*, 2012.

[19] J. Gancet, D. Urbina, P. Letier, M. Ilzokvitz, P. Weiss, F. Gauch, G. Antonelli, G. Indiveri, G. Casalino, A. Birk *et al.*, "Dexrov: Dexterous undersea inspection and maintenance in presence of communication latencies," in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV)*, 2015.

[20] E. Simetti, S. Galeano, and G. Casalino, "Underwater vehicle manipulator systems: Control methodologies for inspection and maintenance tasks," in *OCEANS 16*, Shanghai, China, 2016.

[21] 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, 2016.

[22] G. Antonelli, *Underwater Robots*, ser. Springer Tracts in Advanced Robotics. Springer, 2014, vol. 96.

[23] G. Marani and J. Yuh, *Introduction to Autonomous Manipulation: Case Study with an Underwater Robot, SAUVIM*. Springer-Verlag Berlin Heidelberg, 2014, vol. 102.

[24] P. J. From, J. T. Gravdahl, and K. Y. Pettersen, *Vehicle-Manipulator Systems: Modeling for Simulation, Analysis, and Control*. Springer-Verlag London, 2014.

[25] G. Casalino and A. Turetta, "Coordination and control of multiarm, non-holonomic mobile manipulators," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, vol. 3, Las Vegas, NV, USA, 2003, pp. 2203–2210.

[26] G. Casalino, A. Turetta, and A. Sorbara, "Computationally distributed control and coordination architectures for underwater reconfigurable free-flying multi-manipulator," in *Workshop on Underwater Robotics (IWUR 2005)*, Genova, Italy, November 2005.

[27] 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.

[28] H. Farivarnejad and S. A. A. Moosavian, "Multiple impedance control for object manipulation by a dual arm underwater vehicle–manipulator system," *Ocean Engineering*, vol. 89, pp. 82–98, 2014.

[29] G. Casalino, M. Caccia, A. Caiti, G. Antonelli, G. Indiveri, C. Melchiorri, and S. Caselli, "Maris: A national project on marine robotics for interventions," in *Control and Automation (MED), 2014 22nd Mediterranean Conference of.* IEEE, 2014, pp. 864–869.

[30] E. Simetti, G. Casalino, S. Torelli, A. Sperindé, and A. Turetta, "Underwater floating manipulation for robotic interventions," in *IFAC World Congress 2014*, August 2014, pp. 3358–3363.

[31] E. Simetti, G. Casalino, N. Manerikar, S. Torelli, A. Sperindé, and F. Wanderlingh, "Cooperation between autonomous

underwater vehicle manipulations systems with minimal information exchange," in *IEEE/MTS OCEANS 2015*, Genova, Italy, May 2015.

[32] E. Simetti, A. Turetta, and G. Casalino, "Distributed control and coordination of cooperative mobile manipulator systems," in *Distributed Autonomous Robotic Systems 8*, H. Asama, H. Kurokawa, J. Ota, and K. Sekiyama, Eds. Springer Berlin Heidelberg, 2009, pp. 315–324.

[33] E. Zereik, A. Sorbara, A. Merlo, E. Simetti, G. Casalino, and F. Didot, "Space robotics supporting exploration missions: Vision, force control and coordination strategy for crew assistants," *Intelligent Service Robotics*, vol. 4, no. 1, pp. 39–60, January 2011.

[34] S. Singh, S. E. Webster, L. Freitag, L. L. Whitcomb, K. Ball, J. Bailey, and C. Taylor, "Acoustic communication performance of the whoi micro-modem in sea trials of the nereus vehicle to 11,000 m depth," in *OCEANS 2009, MTS/IEEE Biloxi-Marine Technology for Our Future: Global and Local Challenges*. IEEE, 2009, pp. 1–6.

[35] B. Li, S. Zhou, M. Stojanovic, L. Freitag, and P. Willett, "Multicarrier communication over underwater acoustic channels with nonuniform doppler shifts," *Oceanic Engineering, IEEE Journal of*, vol. 33, no. 2, pp. 198–209, 2008.

[36] Evologics S2CR-18-34 acoustic modem. Evologics GmbH. [Online]. Available: https://www.evologics.de/files/DataSheets/EvoLogics_S2CR_1834_Product_Information.pdf

[37] C. Pontbriand, N. Farr, J. Ware, J. Preisig, and H. Popenoe, "Diffuse high-bandwidth optical communications," in *OCEANS 2008*. IEEE, 2008, pp. 1–4.

[38] C. Pontbriand, N. Farr, J. Hansen, J. C. Kinsey, L.-P. Pelletier, J. Ware, and D. Fourie, "Wireless data harvesting using the auv sentry and whoi optical modem," in *OCEANS'15 MTS/IEEE Washington*. IEEE, 2015, pp. 1–6.

[39] N. Manerikar, G. Casalino, E. Simetti, S. Torelli, and A. Sperindé, "On autonomous cooperative underwater floating manipulation systems," in *International Conference on Robotics and Automation (ICRA 15)*, Seattle, WA, May 2015, pp. 523–528.

[40] G. Casalino, E. Simetti, N. Manerikar, A. Sperindé, S. Torelli, and F. Wanderlingh, "Cooperative underwater manipulation systems control developments within the MARIS project," in *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles (NGCUV 2015)*, vol. 48, no. 2, April 2015, pp. 1–7.

[41] 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. (accepted), 2016.

[42] E. Simetti, G. Casalino, S. Torelli, A. Sperindé, and A. Turetta, "Experimental results on task priority and dynamic programming based approach to underwater floating manipulation," in *OCEANS 2013*, Bergen, Norway, June 2013.

[43] A. Ben-Israel and T. Greville, *Generalized inverses: theory and applications*. Springer Verlag, 2003, vol. 15.

[44] T. Perez and T. I. Fossen, "Kinematic models for manoeuvring and seakeeping of marine vessels," *Modeling, Identification and Control*, vol. 28, no. 1, pp. 19–30, 2007.

[45] G. Casalino, "Trident overall system modelling, including all variables needed for reactive coordination," ISME, Tech. Rep., 2011, available online at: http://www.graal.dist.unige.it/files/89. [Online]. Available: http://www.graal.dist.unige.it/files/89

[46] T. Yoshikawa, "Manipulability of robotic mechanisms," *Int. J. of Robotics Research*, vol. 4, no. 1, pp. 3–9, 1985.

[47] 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, 1991, pp. 1211–1216.

[48] E. Simetti and G. Casalino, "Technical note on the integration of inequality control objectives and task transitions in priority based control," GRAAL, DIBRIS, University of Genova, Tech. Rep., 2014. [Online]. Available: http://www.graal.dibris.unige.it/files/131

[49] K. L. Doty, C. Melchiorri, and C. Bonivento, "Theory of generalized inverses applied to robotics," *International Journal of Robotics Research*, vol. 12, no. 1, pp. 1–19, 1993.

[50] 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.

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

[52] ——, "Preliminary experiments in model-based thruster control for underwater vehicle positioning," *Oceanic Engineering, IEEE Journal of*, vol. 24, no. 4, pp. 495–506, 1999.

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

[54] R. Conti, E. Meli, A. Ridolfi, and B. Allotta, "An innovative decentralized strategy for i-auvs cooperative manipulation tasks," *Robotics and Autonomous Systems*, vol. 72, pp. 261–276, 2015.

[55] M. Prats, J. Pérez, J. J. Fernández, and P. J. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 2577–2582.

[56] M. Caccia, G. Indiveri, and G. Veruggio, "Modeling and identification of open-frame variable configuration unmanned underwater vehicles," *IEEE journal of Oceanic Engineering*, vol. 25, no. 2, pp. 227–240, 2000.

[57] D. Ribas, P. Ridao, A. Turetta, C. Melchiorri, G. Palli, J. Fernandez, and P. Sanz, "I-AUV Mechatronics Integration for the TRIDENT FP7 Project," *Mechatronics, IEEE/ASME Transactions on*, vol. PP, no. 99, pp. 1–10, 2015.

[58] G. Casalino, M. Caccia, S. Caselli, C. Melchiorri, G. Antonelli, A. Caiti, G. Indiveri, G. Cannata, E. Simetti, S. Torelli, A. Sperind, 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, 2016.

[59] D. L. Rizzini, F. Kallasi, F. Oleari, and S. Caselli, "Investigation of vision-based underwater object detection with multiple datasets," *Int J Adv Robot Syst*, vol. 12, no. 77, pp. 1–13, 2015.

[60] G. Antonelli and E. Cataldi, "Recursive adaptive control for an underwater vehicle carrying a manipulator," in *Control and Automation (MED), 2014 22nd Mediterranean Conference of*. IEEE, 2014, pp. 847–852.

[61] 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.

[62] G. Parlangeli and G. Indiveri, "Single range observability for cooperative underactuated underwater vehicles." in *Proceedings of the 19th IFAC World Congress*, vol. 19, 2014, pp. 5127–5138.

[63] 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.

Enrico Simetti (M'10) received the received the Ph.D. degree from University of Genova in 2012, and since 2014 he is an Assistant Professor at DIBRIS, teaching "Automation Systems" in the Mechanical Engineering course and "Cooperative Robotics" in the Robotics Engineering course. His main interests of research are focused on the development of control systems and coordination algorithms for robotic platforms. In particular, he is interested in marine robotics, including surface and underwater vehicles, and autonomous manipulation. He is also involved in the development of a surface vehicle prototype for security and surveillance applications. In the last few years, he developed strong skills in real time and embedded system software development. He is currently co-author and supervisor of most of the software developed in the Research Unit he is part of. He is ISME responsible scientist in the H2020 ROBUST project on deep mining exploration with AUVs.

Giuseppe Casalino (M'01) is currently full professor at DIBRIS, University of Genova, holding the chair of "Industrial Robotics" and also teaching the course of "Automatic Control". His research activities are in the field of Robotics and Automation, with special interests devoted to all the aspects involving planning, motion and interaction control problems within sensorized multirobot structures. Particular applicative interests are directed toward the fields of Underwater and Space Robotics.

Within DIBRIS, he is currently the head of the Laboratory of Robotics and Automation. He is also the Director of the Interuniversity Research Center on Integrated Systems for the Marine Environment (ISME), and also Director of the research laboratories of the Ligurian Technological District on Integrated Intelligent Systems (SIIT). He is, and has been, the responsible scientist of many EEC funded collaborative research projects and many Italian funded research projects, all in the field of robotics and automation. He is the author of more then hundred papers on the subject, published on international journal and conferences.