Underwater Vehicle Manipulator Systems: Control Methodologies for Inspection and Maintenance Tasks

Enrico Simetti, Stefano Galeano, Giuseppe Casalino
Interuniversity Research Center on Integrated Systems for the Marine Environment
DIBRIS, University of Genova,
Via Opera Pia 13, 16145 Genova, Italy
Email: enrico.simetti@unige.it

***Abstract*—This paper presents the control framework under development within the DexROV Horizon 2020 project, for the execution of maintenance and inspection tasks by a semi-autonomous ROV. The work exploits a task priority based kinematic inversion developed by the authors, extending it to encompass also a force regulation task. A way to manage transitions between the different DexROV missions is also given. The paper presents some simulation results to support the proposed control architecture.**

I. Introduction

The Interuniversity Research Center on Integrated Systems for the Marine Environment (ISME, Italy) is working since more than a decade in marine robotics, with special focus on unmanned systems, from the theoretical, experimental and technological transfer point of views.

One of the strongest research trends regards the use of Underwater Vehicle Manipulator Systems (UVMS) for the execution of intervention tasks, i.e. tasks that require the manipulation of objects or interaction with the environment. Such a topic has been tackled within the successful TRIDENT Framework Programme 7 project [1], where ISME focused on the developing of a kinematic control strategy for the coordinated control of the vehicle and arm UVMS subsystems [2], [3], [4], [5]. ISME is now continuing the studies on this challenging topic within the Italian research project MARIS [6], where the previously developed methodology is extended to the control of dual arm systems [7] and to the problem of cooperative manipulation and transportation by two UVMSs [8], [9].

Nowadays, ISME is involved within the DexROV project [10], which is a recently started European project framed in the Horizon 2020 work programme. The main goal of the project is the development of a ROV (Remotely Operated Vehicle) system, endowed with two manipulators, capable to perform dexterous underwater operations, with focus on increased efficiency of subsea operations.

Toward that end, the project focuses on three main objectives:

1) allowing a far distance tele-operation, mainly in terms of supervised control rather than direct joint control to deal with the increased latencies;
2) linked to the previous point, providing semi-autonomous capabilities to the ROV;
3) providing dexterous capabilities to the ROV, such that operations that only divers are currently doing can be done by the ROV as well.

The motivations for the project are quite clear. Currently, maintenance and inspections operations at sea are mainly performed with ROVs operated by expert pilots under stress and heavy fatigue. Furthermore, ROV-based operations are very costly, due to the required offshore support vessels to operate the vehicle. With the possibility of controlling the ROV from a remote control center, the number of crew on board the ship could be reduced to just a few operators for deploying and recovering the ROV, reducing costs. Consequently, this would allow to use a smaller and cheaper support vessels. However, once the control center is moved onshore, the ROV would receive commands through a satellite link, hence one of the main challenge of the DexROV project is how to properly deal with the increased communication latencies.

Toward that end, the idea is to increase the autonomous capabilities of the UVMS. In this way, the remote operators can send higher-level commands performing a supervised control of the ROV-manipulator system, rather than executing a direct tele-operation of the system. These high level commands include, for example, the navigation toward a particular point, clamping to an underwater structure and turning a valve or plugging a connector.

Since all these maintenance and inspection tasks require interaction with underwater structures, ISME is now focusing on force control schemes [11], [12] and their integration in the developed kinematic task priority framework [7]. The control of the force at kinematic level is required since the DexROV setup cannot be directly controlled at torque level. In this work, a pipeline inspection task is used as case study, where an electromagnetic sensor must be maintained in contact with the pipe near its welding line, to check if any surface cracks are present.

The challenge and innovation of this work is that the force control is integrated into a well consolidated task priority control scheme, where many other kinematic tasks must be satisfied. No preliminary information about the environment is used; only the force and moments at the wrist sensor are used to accomplish the inspection task.

This paper presents the developed methodologies in Section II and their validation through simulation results in Section III, while some concluding remarks are given in Section IV.

II. DexROV Control Framework

A. Definitions

To explain the control architecture, we briefly recall some basic notations and definition:

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

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

  where $\boldsymbol{q} \in \mathbb{R}^l$ is the arm configuration vector and $\boldsymbol{\eta} \in \mathbb{R}^6$ is the vehicle *generalized coordinate position vector*, which is the stacked vector of the position vector $\boldsymbol{\eta}_1$, with components on the inertial frame $\langle w \rangle$, and the orientation vector $\boldsymbol{\eta}_2$, the latter expressed in terms of the three angles yaw, pitch and roll (applied in this sequence) [13]. 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 $\langle v \rangle$. We are assuming the vehicle fully actuated, hence in the following we will use the system velocity vector as our control vector. However note that if some d.o.f. are not actuated (typically roll and pitch), this can be easily taken into account if the angular rates can be measured;

The control objectives of the UVMS can be divided into two broad categories. Let us consider a configuration dependent scalar variable $x(\boldsymbol{c})$. We define an *equality control objective* when this variable is eventually required to satisfy $x(\boldsymbol{c}) = x_0$. Otherwise, we define an *inequality control objective* when it is required to satisfy $x(\boldsymbol{c}) \geq x_m$ or $x(\boldsymbol{c}) \leq x_M$ 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. 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

$$\dot{x} = \boldsymbol{J}(\boldsymbol{c})\dot{\boldsymbol{y}}, \tag{3}$$

where $\boldsymbol{J} \in \mathbb{R}^{1 \times n}$ is a row vector. Again, in the rest of the paper we will drop the dependency of $\boldsymbol{J}$ from $\boldsymbol{c}$.

With the above premises, we define as *task* the need of tracking at best a suitable reference rate $\dot{\bar{x}}$ 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 with higher priority, and tasks with the same priority should be achieved simultaneously, if possible.

B. DexROV Missions

In the scope of the DexROV project, we are considering the following reference missions:

1) navigation toward a particular point;
2) performing an operation on an underwater panel, clamping to it;
3) performing a free floating manipulation or inspection.

Each mission is characterized by a set of relevant control objectives. These objectives, or equivalently their associated tasks, are listed according to a suitably chosen priority list. In particular, the 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.

C. Control Objectives

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 a multiple solution exists.

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

For the considered DexROV scenario, the first category of tasks is composed only by the force regulation task. Let us define λ^* as the desired force that the end-effector must exert on the environment and λ as the actual force, then the objective is to have

$$\lambda = \lambda^*. \tag{4}$$

The second group of objectives is the one related with the safety of the system. Within this group we have the joint limits avoidance objective, which means having the following inequality control objectives fulfilled

$$\begin{cases} q_i \geq q_{i,m} \\ q_i \leq q_{i,M} \end{cases} \quad i = 1, \ldots, l, \tag{5}$$

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. A further safety objective can be the altitude control of the vehicle, which requires maintaining a pre-defined minimum distance from the seafloor:

$$h > h_m \tag{6}$$

where h is the altitude w.r.t. the seafloor and h_m the desired minimum distance.

Following the safety objectives, we have those that are prerequisite for the execution of the particular mission. In this category we find the need of maintaining a minimum arm dexterity. Indeed, approaching kinematic singularities of the arm has the unwanted effect of generating high joint velocities. For avoiding this effect, the arm must keep the manipulability measure μ [14] above a minimum threshold

$$\mu > \mu_m. \tag{7}$$

Another operational-enabling objective is maintaining the target of the operation within the visual cone of the camera system. This is important as the visual feedback is necessary to perform the operation. 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), is within a maximum threshold, i.e.

$$\|\boldsymbol{\xi}\| \leq \xi_M. \tag{8}$$

Another task belonging to this category is the alignment of the sensor with the surface's normal unit vector. Similarly to the previous objective, if we take the misalignment vector $\boldsymbol{\varphi}$ between the sensor axis that needs to be aligned with the surface's normal $\boldsymbol{n}$ we have

$$\|\boldsymbol{\varphi}\| \leq \varphi_M. \tag{9}$$

The successive category of tasks represents those defining the actual mission of the system. For the kind of tasks that the DexROV system must execute two main mission oriented task can be defined: the position control of the vehicle and the one for the end-effector of the manipulator. In the first case, considering the as $\boldsymbol{e}_v$ the position and orientation error between the vehicle frame $\langle v \rangle$ and a given goal frame $\langle g_v \rangle$ we need to reach the situation where

$$\|\boldsymbol{e}_v\| \leq e_{v,M}. \tag{10}$$

This requirement will make the vehicle reach approximately the required position, with a precision depending on the value of $e_{v,M}$. For the end-effector control, defining similarly the error $\boldsymbol{e}_e$ we can instead require its complete zeroing, i.e.

$$\boldsymbol{e}_e = \boldsymbol{0}. \tag{11}$$

Finally, the last group of objectives is related to finding out which among the solutions optimizes some additional criteria. Of course, this is only relevant if more than a solution exists. In any case, these possible optimization criteria include:

- minimize the vehicle velocity, which can be useful to use the vehicle the least possible, due to its much weaker performances compared to the arm;
- maintain the arm in a preferred shape, which allows to perform repetitive tasks minimizing the internal motions of the arm;
- any repetition of the safety or operational enabling tasks, defined as inequalities, with more stringent requirements.

D. Managing Transitions

As evidenced in the previous points, each mission is characterized by its task hierarchy that may have different tasks in common, even if with a different priority. For instance, consider the following two hypothetical lists of scalar tasks (now labeled with alphabetic letters) for two different missions, where $A \prec B$ denotes that A has higher priority than B:

$$\begin{aligned} \mathrm{M1}: & A \prec B, C, D \\ \mathrm{M2}: & A \prec D \prec C, E \end{aligned}$$

where A, D, C are in common, but with D at a different priority ordering w.r.t. C within the two lists. A unified list, among the possible ones, is the following

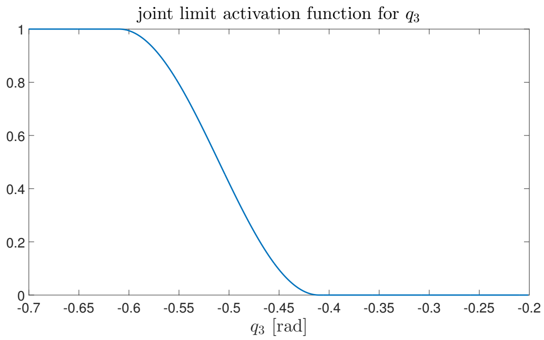
$$\mathrm{U}: A \prec D \prec B, C, D, E;$$

where some tasks may be duplicated as the example shows. The suitable insertion/deletion of some of its entries, while transitioning among the missions, produces the new mission task hierarchy. 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.

E. 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.



(a)

Fig. 1. Example of activation function for a joint limit task, where $q3 > 0.61$ and $\beta = 0.05$.

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

$$a_{(j)} \triangleq a^p_{(j)} a^i_{(j)}, \tag{12}$$

which have the following specific purposes:

- $a^i_{(j)}$ is function of the current value of the actual j-th component $x_{(j)}$ and it is used to activate/deactivate a scalar task associated to an inequality type control objective;
- $a^p_{(j)}$ is a function of the current mission phase and its output is exploited to activate/deactivate any task involved/not-involved in the new phase whenever there is a phase transition.

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 $x_{(j)} \geq x_{(j),m}$):

$$a^i_{(j)} \triangleq \begin{cases} 0, & x_{(j)} > x_{(j),m} + \beta_{(j)} \\ s_j(x), & x_{(j),m} \leq x_{(j)} \leq x_{(j),m} + \beta_{(j)} \\ 1, & x_{(j)} < x_{(j),m} \end{cases} \tag{13}$$

where $s_j(x)$ is any sigmoid function with a continuous behaviour from 0 to 1 when $x_{(j),M} - \beta_{(j)} \leq x_{(j)} \leq x_{(j),M}$ (see Fig. 1 for an example). 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. On the other hand, 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. 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.

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:

$$S_k \triangleq \left\{ \arg \mathrm{R}\text{-}\min_{\dot{\boldsymbol{y}} \in S_{k-1}} \left\| \boldsymbol{A}_k (\dot{\bar{\boldsymbol{x}}}_k - \boldsymbol{J}_k \dot{\bar{\boldsymbol{y}}}) \right\|^2 \right\}, \; k = 1, 2, \ldots, N, \tag{14}$$

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 functions [15], [16]. This methodology (named *iCAT task priority framework*) is duly reported in [7] and will be omitted here.

Let us only briefly focus on the force regulation task, since most of the other tasks are already well covered in the TRIDENT technical report [17]. Let us define $e(t) \triangleq \lambda^* - \lambda$ as the force error and let us consider a PI control law to regulate the force:

$$\dot{\bar{x}}_f \triangleq K_p e(t) + K_i \int_0^t e(\tau) d\tau \tag{15}$$

Let us suppose a frictionless contact, and let us define the contact Jacobian as

$$\boldsymbol{J}_f \triangleq \boldsymbol{n}^T \boldsymbol{J}_{el} \tag{16}$$

where $\boldsymbol{n} \in \mathbb{R}^{3 \times 1}$ is the surface unit vector where the force is exerted, and $\boldsymbol{J}_{el} \in \mathbb{R}^{3 \times n}$ is the linear end-effector Jacobian.

Then, the force regulation task, which is also the first task of the hierarchy, can be written as

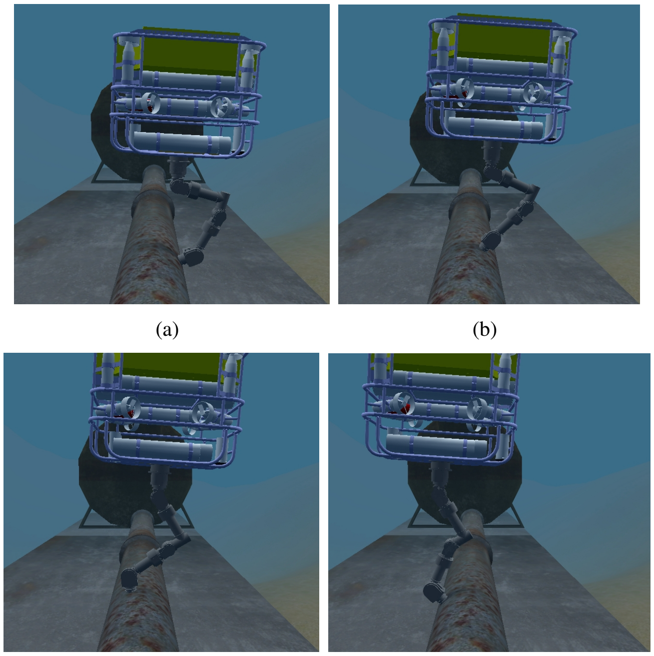
$$S_1 \triangleq \left\{ \arg \mathrm{R}\text{-}\min_{\dot{\bar{\boldsymbol{y}}}} \|\dot{\bar{x}}_f - \boldsymbol{n}^T \boldsymbol{J} \dot{\bar{\boldsymbol{y}}}\|^2 \right\} \tag{17}$$

whose minimum norm solution is $(\boldsymbol{n}^T \boldsymbol{J})^{\#} \dot{\bar{x}}_f$. Notice how its null space projector guarantees that all the lower priority task, including the end-effector position control tasks, are congruent with the physical constraint of the surface.

G. Vehicle Velocity Compensation

The above outline procedure considers the UVMS as a single system, obtaining the set of arm and vehicle reference velocities that satisfy the given task hierarchy at best. However, it is a fact that the dynamics of arm actuators and those of the vehicle's thrusters are very different. The former are much quicker and more precise than the latter [18], as also shown in the TRIDENT experimental trials [5]. For this reason, the task dynamics will be mostly affected by the vehicle velocity tracking errors. To compensate for this fact, it is possible to add in parallel to the above devised centralized procedure a similar one where only the arm joints are used as control variables, leaving the vehicle velocity as a parameter that affects the task references. The solution of this procedure is an arm control law of the type:

$$\dot{\boldsymbol{q}} = \boldsymbol{\rho} + \boldsymbol{P} \boldsymbol{v}, \tag{18}$$



(a) (b) (c) (d)

Fig. 2. Screenshots of the UVMS as it performs the pipe welding inspection task.

where $\boldsymbol{\rho}$ is the output of this new task hierarchy resolution, and $\boldsymbol{P} \in \mathbb{R}^{l \times 6}$ is a matrix that relates the vehicle velocity with the optimal arm joint velocity reference. Substituting $\boldsymbol{v}$ with the actual vehicle velocity allows to compensate for the vehicle velocity tracking errors. The performance of the compensation w.r.t. each task dynamics will depend on how precise is the feedback, as well as if the corresponding task Jacobians remain full-rank without the vehicle control variables.

III. Simulation Results

In this section we present some simulation results of the proposed kinematic control strategy. The reference mission is the inspection of a pipeline. A reference path is defined without an a priori knowledge of the pipe, and the expected results is that the end-effector follows the path projected on the pipe's surface. In all the simulations we have simulated a frictionless multi-point contact with the end-effector planar surface. All the simulated forces and moments are then transferred to a unique point on the end-effector's rigid body space, where we have assumed that a force/torque sensor has been placed. Some screenshots of the UVMS executing the inspection mission are shown in Figure 2. In both simulations the task control hierarchy that has been implement is the following one:

1) Force regulation;
2) Arm Joint limits;
3) Arm Manipulability;
4) End-effector alignment with the surface's normal;
5) Vehicle horizontal attitude;
6) End-effector linear position control;
7) End-effector angular position control;
8) Arm preferred shape;
9) Vehicle velocity minimization.

In the first simulation we have used a computed torque law to simulate the dynamic control layer (DCL) of the whole UVMS. Figure 3(a) shows the actual force measured during the trial, where it can be seen how the proposed approach allows to have a good regulation of the force to the desired value of 5 N. The regulation is accomplished despite different tasks such as joint limits and the manipulability one are being activated and deactivated during the trial (see Figures 3(d) and 3(c)). Finally, Fig. 3(b) shows the joint velocity of the arm during the whole simulation.

In the second simulation we have tested our approach when the underlying DCL implements a simpler proportional integrative (PI) control law. This case is closer to the actual implementation of DexROV, where the DCL of the vehicle and the arm are provided by the respective manufacturers and implementing a computed torque is not possible. Figure 4(a) shows the force exerted on the pipe, which again is very close to the desired value of 5 N. As expected, compared to the computed torque case the results are slightly worse, while still remaining quite good. As in the previous simulation, Fig. 4(d) and Fig. 4(c) present the activation values of some tasks that are being activated and deactivated during the mission.

IV. Conclusions

This paper has presented the control methodologies for inspection and maintenance of underwater structure that are being developed within the DexROV project. The task priority framework, originally developed within the TRIDENT project [5], later updated with the activation and deactivation of tasks within the MARIS project [7], has been used as the basis of this work, where a force regulation task has been added to perform the required maintenance operations. In particular, a use case where the UVMS must inspect the surface of a pipeline, in correspondence of a welding line, has been presented and has been used in a simulated environment to validate the proposed control scheme.

Acknowledgment

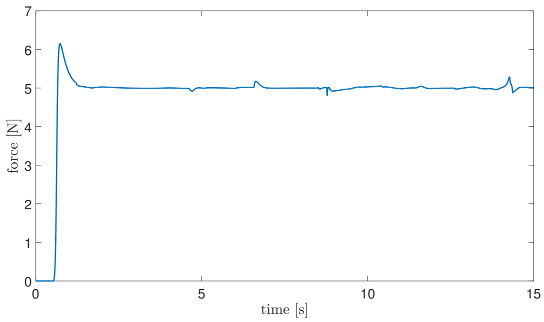
This work has been supported by the MIUR (Italian Ministry of Education, University and Research) through the MARIS prot. 2010FBLHRJ project and by the European Commission through H2020-BG-06-2014-635491 DexROV project.

References

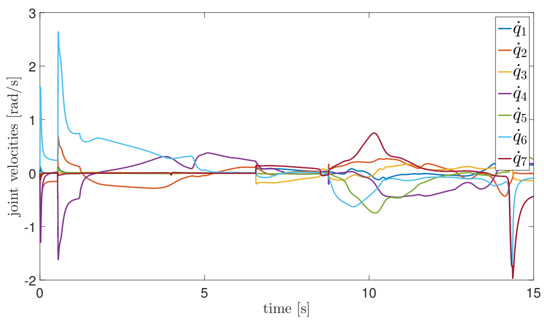
[1] J. P. Sanz, P. Ridao, G. Oliver, C. Melchiorri, C. Casalino, C. Silvestre, Y. Petillot, and A. Turetta, "Trident: A framework for autonomous underwater intervention missions with dexterous manipulation capabilities," in *IFAC Intelligent Autonomous Vehicles (IAV 2010)*, September 2010.

[2] 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.

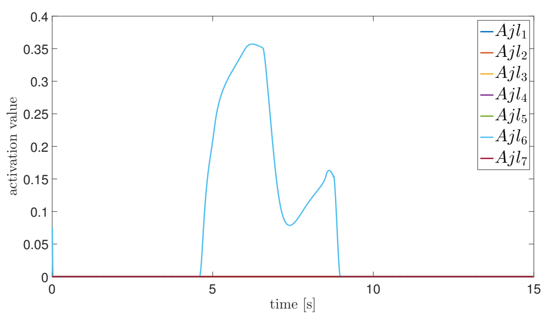
[3] ——, "A task priority and dynamic programming based approach to agile underwater floating manipulation," in *9th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC 2012)*, Arenzano, Italy, September 2012.



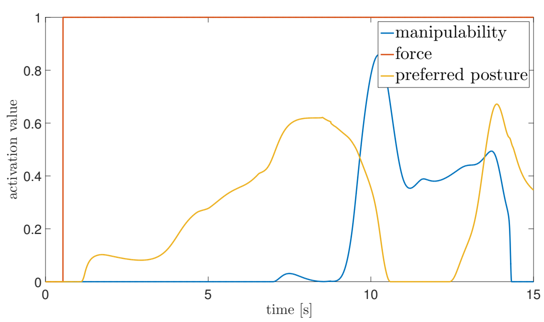
(a)



(b)

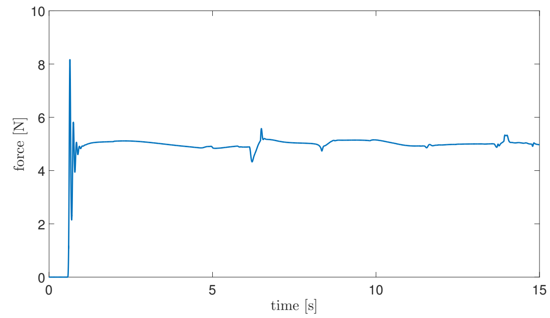


(c)

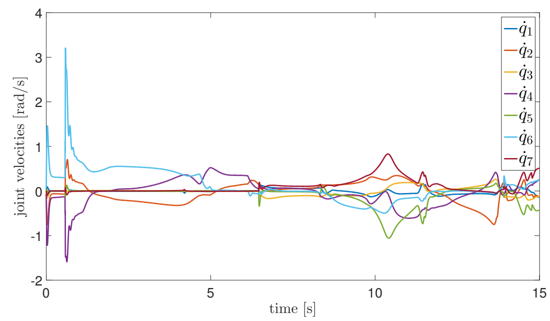


(d)

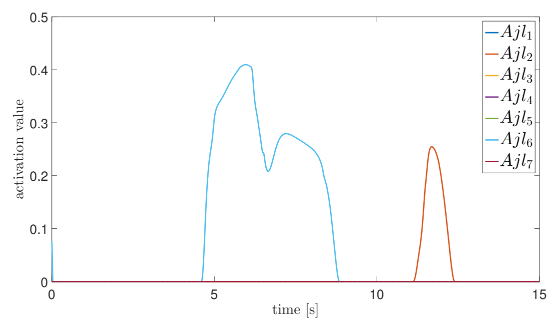
Fig. 3. Simulation results with a computed torque DCL (a) the time behaviour of the norm of the force (reference value is 5 N), (b) the arm joint velocities (c) the activation values of the joint limit tasks, and (d) the activation values of the other tasks.



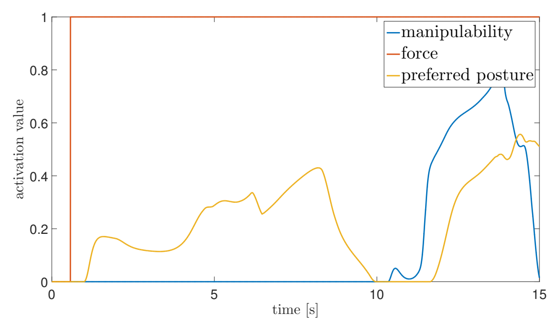
(a)



(b)



(c)



(d)

Fig. 4. Simulation results with a PI DCL (a) the time behaviour of the norm of the force (reference value is 5 N), (b) the arm joint velocities (c) the activation values of the joint limit tasks, and (d) the activation values of the other tasks.

[4] 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.

[5] ——, "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.

[6] 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.

[7] 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.

[8] 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.
[9] 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.
[10] 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.
[11] G. Antonelli and E. Cataldi, "Basic interaction operations for an underwater vehicle-manipulator system," in *ICAR 2015 - 17th International Conference on Advanced Robotics*, Istanbul, Turkey, July 2015.
[12] ——, "Virtual decomposition control for an underwater vehicle carrying a n-dof manipulator," in *MTS/IEEE OCEANS 2015*, Genoa, Italy, May 2015.
[13] 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.
[14] T. Yoshikawa, "Manipulability of robotic mechanisms," *Int. J. of Robotics Research*, vol. 4, no. 1, pp. 3–9, 1985.
[15] 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.
[16] 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.
[17] 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
[18] 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.